

# Unified System Level Reliability Evaluation Methodology for Multiprocessor Systems-on-Chip

Alexandre Yasuo Yamamoto and Cristinel Ababei  
Department of Electrical and Computer Engineering  
North Dakota State University, Fargo ND, USA  
Email: {alexandre.yamamoto, cristinel.ababei}@ndsu.edu

**Abstract**—Reliability is a growing fundamental challenge in the design of multiprocessor Systems-on-Chip (MPSoCs). This trend is accelerated by the increasingly adverse process variations and wearout mechanisms that result in an increased number of errors. Previously proposed fault-tolerant techniques are ad-hoc and target processors or Networks-on-Chip (NoC) separately. Because each of these two units may become a reliability bottleneck for NoC based multiprocessor SoCs, it is imperative that the reliability of SoCs be evaluated and addressed in a unified manner, as a combination of communication and computational units. Using this holistic approach, in this paper, we propose a new architecture level unified reliability evaluation methodology for MPSoCs. At the core of the reliability estimation engine lies a Monte Carlo algorithm which works with failure times for time-dependent dielectric breakdown (TDDB) and negative bias temperature instability (NBTI) modeled as Weibull distributions. To demonstrate its usefulness, we utilize the proposed methodology to explore the impact of NoC router layout on the failure time of the system running the same set of benchmarks. In addition, we investigate the failure time of the system when the NoC as the communication unit of the MPSoC is taken or not – as in previous work – into consideration. Our simulation framework can be very helpful to architecture designers, who could use it to identify architectural characteristics and to develop design techniques meant to improve system’s lifetime.

**Keywords**—multiprocessor system-on-chip; network-on-chip; reliability; lifetime;

## I. INTRODUCTION

Due to continuous downscaling of CMOS technologies, several trends exacerbate the traditional design challenges in deep submicron domains. First, increased aging mechanisms cause performance degradation and eventual device and system failure [1]. Second, process variations increase the uncertainty of signal delays and result in variability of circuit performance and power [2]. Third, increased device densities increase the circuit vulnerability to soft errors. Fourth, workload variations and dynamic power management techniques contribute to varying on-chip temperatures. Finally, due to smaller supply voltages, the leakage power consumption increases and voltage noise margins decrease, hence affecting adversely reliability [3], [4]. These increasingly adverse factors lead to an increased number of transient, intermittent, and permanent errors.

To address such errors, designers have used guardbanding. For example, supply voltages are selected high enough in order to guarantee correct functionality despite variation in threshold voltage or in temperature and supply noise. In this way energy

gained from downscaling is sacrificed to combat potential reliability problems [5]. If this sacrifice becomes too large, downscaling may become detrimental [6]. Hence, with power requirements already limiting chip performance, continuing to demand perfect, upset-free transistors while attempting to reduce the energy per operation can no longer be maintained.

Fault tolerance techniques have also been introduced to address errors. However, previously proposed fault-tolerant techniques are ad-hoc and target processors or Networks-on-Chip (NoCs) [7], [8] separately. For a given set of resilience techniques, its effectiveness in achieving the desired system-level reliability must be evaluated, and its associated costs such as system-level energy and performance costs must be quantified. Currently, we are not aware of any attempt to evaluate system reliability in a unified manner, as a combination of both communication and computation units. Therefore, in this paper, we propose a new architecture level unified reliability evaluation methodology for MPSoCs. This methodology provides multiprocessor architecture designers with a framework that enables them to explore multiprocessor architecture characteristics and their impact on the mean time to failure (MTTF) as a measure of system’s reliability.

## II. PREVIOUS WORK AND CONTRIBUTION

Significant work has been carried out to estimate the reliability of either single- and multi-processors [9]–[16] or of computer networks [17], [18]. Reliability of NoCs has only recently been studied [19]–[21]. Evaluation of the reliability of NoC based multiprocessor SoCs is a challenging task because reliability is affected by numerous factors including wearout mechanisms (e.g., time-dependent dielectric breakdown (TDDB) [22], negative bias temperature instability (NBTI) [23], etc.), process variations, dynamic power and thermal management, workload conditions, and system architecture and configuration.

High-level metrics for reliable systems (e.g., reliability, availability, data integrity, mean time to failure (MTTF) [9], mean time to repair (MTTR), architectural vulnerability factor (AVF), failures in time (FIT), FIT for reference circuit (FORC) [12], etc.) have been used for quantifying the benefits of reliable systems. One popular metric, reliability function  $R(t)$  denotes the probability that the system will operate correctly at time  $t$ . The expected value of the reliability function is the MTTF. Increasing MTTF well beyond the expected useful life

of a product is an important design objective.

Reliability estimation can be done by simulation or analytical methods. Many proposed lifetime reliability models assume a uniform device density on the chip and an identical vulnerability of devices to failure mechanisms [9]. The RAMP approach [10] models the MTTF of a processor microarchitecture as a function of temperature related failure rates of individual structures on chip. Because the lifetime distributions of failure mechanisms are assumed to be exponential [10], the system-level reliability is calculated by applying the sum-of-failure-rates (SOFR) model. This approach is not realistic because failure rates of units increase with time due to aging. To address this issue, more general lifetime distributions (e.g., Weibull or lognormal) may be utilized. However, in this case the prediction of the system-level reliability becomes more difficult and Monte Carlo simulations must be employed [11].

Despite the significant work on modeling the lifetime reliability of computer networks and single- and multi-processors, there is no comprehensive methodology for assessing the reliability of NoC based multiprocessor SoCs. Designers should be able to answer questions about which units have the largest impact on system reliability and to validate that certain combinations of resilience techniques offer the optimal reliability for an application. The ability to perform such design activities depends on the availability of accurate and efficient metrics and tools. This is the main motivation for this paper. To this end, our main contribution is as follows:

- We propose a new architecture level unified reliability evaluation methodology for MPSoCs. At the core of the proposed reliability estimation engine lies a Monte Carlo algorithm which works with failure times for time-dependent dielectric breakdown and negative bias temperature instability modeled as Weibull distributions.
- We integrate existing simulation tools to develop a full-system simulation framework and implement the proposed MC based reliability evaluation algorithm. We refer to the proposed reliability estimation tool as REST.
- We utilize the proposed methodology to explore the impact of NoC router layout on the system's MTTF. We also investigate the system's MTTF when the NoC as the communication unit of the MPSoC is taken or not into consideration.

### III. LIFETIME FAILURE MODELS

Most of the previously proposed reliability models assume the lifetime distributions of failure mechanisms to be exponential [9]–[13]. As discussed in the previous section, this allows system level reliability to be calculated by applying the sum-of-failure-rates (SOFR) model. However, this approach is not realistic because failure rates of units increase with time due to aging. To address this issue and to develop an accurate reliability model, more general lifetime distributions (e.g., Weibull and lognormal) must be utilized. On the other hand, when using Weibull or lognormal distributions the analytical prediction of reliability becomes hard and therefore Monte Carlo simulations must be employed. In this paper, we adopt

the Weibull distribution modeling for two of the most critical wearout mechanisms, time dependent dielectric breakdown (TDDB) and negative bias temperature instability (NBTI), because these distributions have been found to best fit the corresponding wearout mechanisms.

#### A. Time Dependent Dielectric Breakdown (TDDB)

Time dependent dielectric breakdown, or gate oxide breakdown, is a well studied failure mechanism in CMOS semiconductor devices. It is caused by the gradual wearout of gate dielectrics, which can lead to transistor degradation and eventually failure due to the formation of a conducting path between gate and substrate [22]. The model for  $MTTF_{TDDB}$  at a temperature  $T$  and voltage  $V$ , is described by the following expression [24]:

$$MTTF_{TDDB} \propto \left(\frac{1}{V}\right)^{a-bT} \times e^{\frac{X+Y}{kT}+ZT} \quad (1)$$

where  $k$  is the Boltzmann's constant and  $a$ ,  $b$ ,  $X$ ,  $Y$ , and  $Z$  are model fitting parameters and are determined from experimental data. In our implementation discussed later on, we use the same values as in [24]  $a = 78$ ,  $b = -0.081$ ,  $X = 0.759eV$ ,  $Y = -66.8eVK$ , and  $Z = -8.37e^{-4}eV/K$  based on the data from [25].

#### B. Negative Bias Temperature Instability (NBTI)

Negative bias temperature instability is an electro-chemical reaction that takes place in PFETs, when they are stressed at large negative gate voltages with respect to the source and drain [23]. It manifests as an increase in the threshold voltage and consequent decrease in drain current and transconductance. Higher chip temperatures exacerbate this phenomenon. The model for  $MTTF_{NBTI}$  at a temperature  $T$  is described by the following expression [24]:

$$MTTF_{NBTI} \propto \left[ \ln\left(\frac{A}{1+2e^{\frac{B}{kT}}}\right) - \ln\left(\frac{A}{1+2e^{\frac{B}{kT}}} - C\right) \right] \times \frac{T}{e^{-\frac{D}{kT}}} \right]^{\frac{1}{\beta}} \quad (2)$$

where  $A$ ,  $B$ ,  $C$ ,  $D$ , and  $\beta$  are model fitting parameters. We use the same values as in [24]  $A = 1.6328$ ,  $B = 0.07377$ ,  $C = 0.01$ ,  $D = -0.06852$ , and  $\beta = 0.3$  based on the data from [26].

## IV. PROPOSED ARCHITECTURE LEVEL RELIABILITY EVALUATION METHODOLOGY

### A. Motivation

The key idea of the proposed time to failure evaluation methodology is to treat the MPSoC in a unified manner as a combination of communication and computation units. The motivation for this new approach is as follows. First, the area occupied by the NoC can represent up to 20% of the total chip area [27]–[29]. This is a significant portion of each tile (see Fig.1) and can drastically impact power and temperature estimations. Second, the power consumption of the NoC can be as much as 25%-40% of the overall chip power consumption [30], [31]. The dissipation of this power can

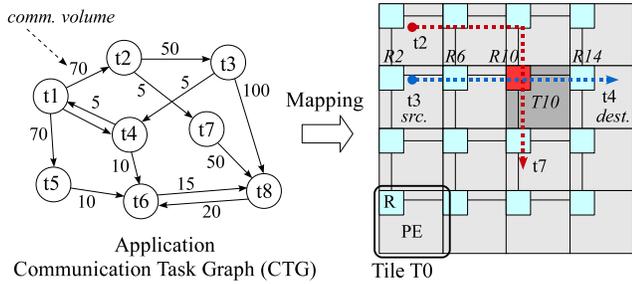


Fig. 1. Simplified diagram of an NoC based multiprocessor SoC. Each tile is composed of a core or processing element (PE) and a router (R).

introduce hotspots that will affect the neighboring processing elements (PE) or cores and introduce errors in their temperature estimations. This problem is exacerbated when the PE of a tile is inactive (e.g., it is not processing any task), while its router is highly active due to the traffic between other source-destination communication pairs. For example, in Fig.1, the processing element of tile  $T_{10}$  is affected by the traffic of  $(t_2, t_7)$  and  $(t_3, t_4)$  communication pairs, which contribute to the power consumption of the router  $R_{10}$ . Inspired by the RAMP approach [10], [11], which focused on a processor alone, the proposed unified model accounts for the behavior of the executing application and it will therefore capture the impact of workload variations on reliability.

### B. Full System Simulation Framework

In order to implement and evaluate the proposed reliability evaluation methodology we construct a full-system simulation framework. The block diagram of the our simulation framework illustrates the main steps of the proposed reliability evaluation methodology and is shown in Fig.2. Its key components are as follows:

- First and foremost, we need a multicore processor cycle-accurate simulator. For that, we utilize the gem5 [32]

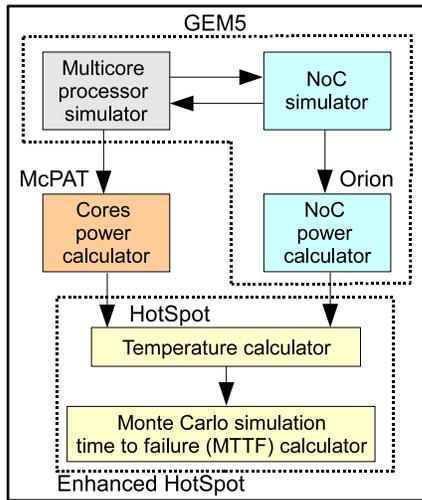


Fig. 2. Block diagram of REST: full-system simulation framework with integrated power consumption, temperature, and MTTF estimators.

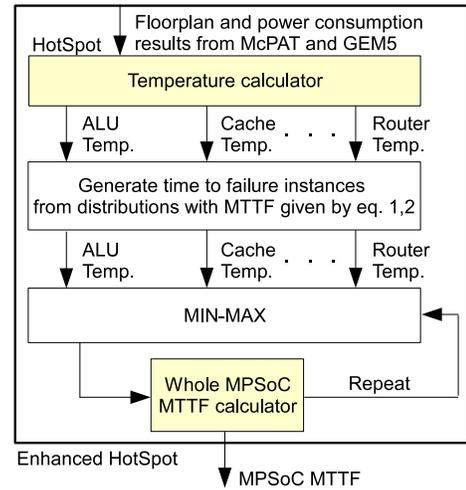


Fig. 3. Block diagram of the Monte Carlo simulation based MPSoC time to failure evaluation methodology.

full-system simulator, which is a combination of M5 full-system simulator [33] and GEMS [34] (essentially Ruby with support for cache coherence protocols and interconnect models via Garnet [35]). gem5 provides detailed timing and performance data and also integrates capabilities to estimate NoC router and link power consumptions. Therefore, simulation of a given benchmark is accurate as it accounts for the operating system as well.

- Performance data of each of the cores are then used as input to the power estimator McPAT [36]<sup>1</sup>. The output of the McPAT power estimator is a list with power consumptions of each subblock of each core.
- Processors power consumptions provided by McPAT and the power consumption of individual routers of the NoC (provided by gem5) are fed then to HotSpot [38], [39]. HotSpot is an accurate and fast thermal model based on an equivalent circuit of thermal resistances and capacitances that correspond to microarchitecture blocks. The output of the HotSpot simulation is a list with temperatures of all NoC routers and of each subblock of all cores of the MPSoC.
- These temperatures are utilized together with the system level architecture floorplan by the Monte Carlo simulation engine to estimate the time to failure of the whole system. Details of this engine are presented in the next subsection.

### C. Monte Carlo Simulation Based Time to Failure Estimation

At the core of the proposed architecture level reliability evaluation methodology we employ a Monte Carlo (MC) simulation algorithm, which we implemented inside the source code of HotSpot. The flow chart of the MC algorithm is shown in Fig.3.

<sup>1</sup>Because the output format of gem5 is not compatible with the input format of McPAT, we utilize a modified version of [37] to automatically do the necessary data formatting.

Algorithm: Monte Carlo algorithm	
1:	<b>In:</b> MPSoC floorplan and power consumption of all subblocks
2:	<b>Out:</b> Estimate of MTTF of whole MPSoC
3:	<b>for</b> $l \leftarrow 1$ to $F$ <b>do</b> // $F$ : number of failure types
4:	Calculate $MTTF_l$ using equations from Section III
5:	<b>for</b> $j \leftarrow 1$ to $N$ <b>do</b> // $N = 10^5$ Monte Carlo iterations
6:	$t_{min}^j \leftarrow INF$ // Initialize
7:	<b>for</b> $k \leftarrow 1$ to $S$ <b>do</b> // $S$ : number of subblocks
8:	$t_{f_k} \leftarrow generate\_instance(MTTF_l)$
9:	<b>if</b> $t_{f_k} < t_{min}^j$ <b>then</b> // Generalization: $MIN\_MAX$
10:	$t_{min}^j = t_{f_k}$
11:	<b>end if</b>
12:	<b>end for</b>
13:	<b>end for</b>
14:	$t_{f_l} = \frac{\sum_j^N t_{min}^j}{N}$
15:	<b>end for</b>
16:	return $t_f = MIN\{t_{f_l}\}$ // Estimate of MTTF of whole MPSoC

Fig. 4. Algorithm pseudocode of the Monte Carlo simulation.

The input to the HotSpot temperature calculator is the floorplan of the MPSoC and power consumption of all subblocks: NoC routers and components of each processor core (e.g., ALU unit, L1 cache, etc.). We assume a regular tiled floorplan for the MPSoC and a regular 2D mesh NoC. The output of HotSpot is a list with temperatures for all routers and subblocks of each processor core. Note that these temperatures depend on the individual utilization of all cores and routers as exercised by the application and its traffic. Similar to the RAMP approach [10], [11] these temperatures are plugged into equations (1) and (2) from Section III. These equations model the mean time to failures of the probability distributions associated with each router and core subblock, from which we draw samples (or instances) during the Monte Carlo iterations.

The MC algorithm (see also Fig.4) proceeds with the following main steps (1) for each failure mechanism run  $N = 10^5$  simulations: (a) for each subblock, generate failure time instances from the corresponding distribution and (b) use MIN-MAX analysis of these times according to the system's configuration to calculate the time to failure  $t_{min}^j$  during simulation iteration  $j = 1, \dots, N$ . (2) calculate the time to failure for the current failure mechanism as  $t_{f_l} = (\sum_{j=1}^N t_{min}^j) / N$ . (3) calculate the value of the overall MTTF or time to failure of the MPSoC as the minimum among the failure times due to each failure mechanism.

#### D. Generation of Samples from a Weibull Distribution

During each MC simulation iteration, we need to generate random *instances* of failure times for each subblock. This is realized by the *generate\_instance(MTTF<sub>l</sub>)* procedure called in line number 8 in Fig.4, which draws samples from Weibull distributions whose means are given by equations (1) and (2). Because the Weibull cumulative distribution function is given by:

$$F(x) = 1 - e^{-\left(\frac{x}{\alpha}\right)^\beta} \quad (3)$$

one can generate samples via the expression:

$$x_{sample} = \alpha \cdot [-\ln(1 - u)]^{\frac{1}{\beta}} \quad (4)$$

where  $u = rand(0, 1)$  is a random number generated uniformly from the interval  $[0, 1]$ .  $\alpha$  and  $\beta$  are the scale and the shape factors characterizing the Weibull distribution. In our implementation of *generate\_instance(MTTF<sub>l</sub>)*, we utilize a value of  $\beta = 1.64$  as in [40] while  $\alpha$  is derived from the expression of the mean of a Weibull distribution:

$$\alpha = \frac{MTTF_l}{\Gamma\left(1 + \frac{1}{\beta}\right)} \quad (5)$$

where  $\Gamma(\cdot)$  is the Gamma function.

## V. SIMULATION RESULTS

To demonstrate the proposed reliability evaluation methodology, we utilize it in two different sets of experiments. In all our simulations, we utilize a set of four Parsec benchmarks [41]. The default architectural configuration parameters utilized in our simulations, unless otherwise specified, are shown in Table I.

TABLE I  
ARCHITECTURAL CONFIGURATION PARAMETERS.

Parameter	Value
Technology node	180nm
Core (Frequency, VDD)	Alpha EV6 21264 (1GHz, 2V)
Branch predictor	2 bit counter
Reorder buffer	80-entries
L1 ICache	32KB
L1 DCache	64KB
L2	2MB
Network	2D regular mesh, 1 router per core
Link bandwidth	32 bits
Routing algorithm	XY
Number of virtual channels (VCs)	2

#### A. Router Location Within the Tile

In this set of experiments, we investigate the impact of NoC router location within the floorplan of a single tile on the MTTF of the overall MPSoC. We consider two simple tile layouts as shown in Fig.5. While the area occupied by a router depends primarily on the buffers size and the number of ports, based on the discussions and designs in [27]–[29], we assume a router whose area is 20% of the area occupied by the processor core within a tile.

The comparison between the MTTFs achieved in these two different cases for each of the simulated benchmarks on multicore architectures with 4, 16, and 64 cores is shown in Fig.6. We observe that when the router is located in the upper part of the tile, as shown in Fig.5.a, the system's MTTF is slightly shorter but with no more than 3%. Because in the case shown in Fig.5.b the router is further away from the actual Alpha core, the thermal profile of the overall system is better. However, the difference is rather small; we suspect as the main reason the benchmarks, which do not create a lot of traffic through the network. Nevertheless, when the router is located in the upper part of the tile closer to the Alpha core,

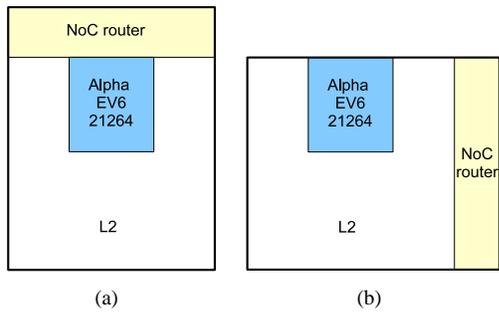


Fig. 5. Tile layouts with different locations for the NoC router: (a) top router, (b) side router.

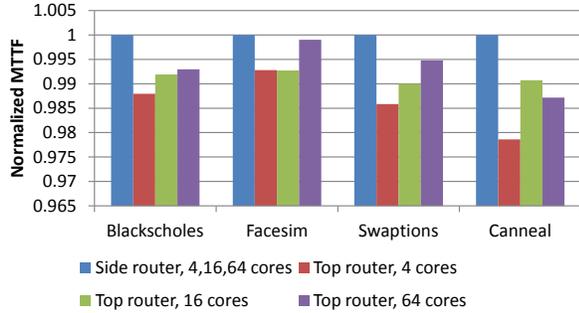


Fig. 6. Relative comparison of the MPSoC's MTTF achieved for two different locations of the NoC routers within a tile. The two different NoC routers are shown in Fig.5.

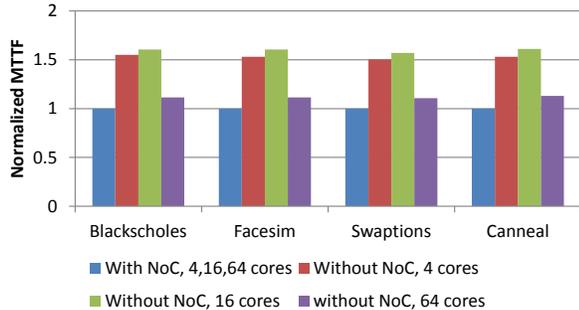


Fig. 7. Relative comparison of the MPSoC's MTTF achieved when the network is taken or not into consideration during the reliability evaluation process. Results are obtained for the side router layout from Fig.5.b.

it still represents a poorer heat sink (due to its own higher temperature) for the heat diffused from the core.

### B. Network Impact

Here, we investigate the impact of taking into consideration the NoC (as the communication unit of the MPSoC) on the MTTF of the overall MPSoC. In other words, we want to see with how much is the MTTF optimistically estimated by previous reliability models, which did not consider the network. The comparison between the MTTFs achieved in these two different cases for each of the simulated benchmarks on multicore architectures with 4, 16, and 64 cores is shown in Fig.7. In both cases we utilize the tile layout from Fig.5.b.

We observe that when the NoC is not taken into account during the lifetime evaluation process, the MTTF of the overall

MPSoC is with up to 60% longer than when the network is included. This is not surprising, as previous work found that networks and processors alone can reach peak temperatures of  $68.6^{\circ}C$  and  $77.9^{\circ}C$ , respectively, while when networks and processors are jointly considered, chip peak temperature can reach  $104.7^{\circ}C$  [42]. We also note that when the MPSoC architecture is composed of 64 cores the difference is about 12% only. This is because the execution of a given benchmark is split among a larger number of cores and the traffic per router is less compared to architectures with fewer cores. This was confirmed by the activity statistics reported by Ruby, which showed that the average activity per router was less than half the activity recorded when the same benchmark was run on the 16 core architecture.

### C. Discussion and Future Work

While the main goal of this paper is to introduce the proposed unified reliability evaluation methodology, here we present preliminary architecture level design exploration scenarios. We are currently extending our investigation to all Parsec benchmarks as well as other benchmarks including Spec 2006 and Splash. In addition, while we have considered only TDDDB and NBTI wearout mechanisms, the proposed framework can be easily extended to accommodate other wearout mechanisms such as electromigration, thermal cycling, and stress migration.

The computational runtime of our simulation framework is dominated by the gem5 full-system simulator, which may take several hours or longer depending on the benchmark size. The computational runtime of the Monte Carlo algorithm (implemented in C++) is in the order of several minutes on a Linux machine running on a 2.8 GHz Intel Quad processor with 4 GB memory. While gem5 suffers from long computational runtimes, it is a sophisticated and capable simulation platform, which can simulate the operating system as well as a variety of core types. Hence, this reliability evaluation framework could be utilized to explore a large variety of design tradeoffs and techniques spanning multiple layers. For example, designers could investigate dynamic voltage and frequency scaling, activity throttling, workload migration/scheduling among cores, and network traffic migration via adaptive routing as mechanisms or knobs to control and regulate the power/thermal profiles of the overall chip or to budget lifetime of individual cores. Such investigations are left to future work.

## VI. CONCLUSION

We proposed a new architecture level unified reliability evaluation methodology for MPSoCs. This approach is motivated by the fact that each of the communication and computational units of multicore processors may become a reliability bottleneck. At the core of the reliability estimation engine lies a Monte Carlo algorithm which works with failure times for TDDDB and NBTI modeled as Weibull distributions. We utilized the proposed methodology to explore the impact of NoC router layout on the system's lifetime. We also investigated how system's lifetime changes when the NoC

as the communication unit of the MPSoC is considered or not during the reliability evaluation process and found that differences can be as high as 60%.

#### ACKNOWLEDGMENT

This work was supported by the National Science Foundation (NSF) under grant CCF-1116022. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and do not necessarily reflect the views of the NSF.

#### REFERENCES

- [1] M. Agarwal, B.C. Paul, M. Zhang, and S. Mitra, "Circuit failure prediction and its application to transistor aging," *IEEE VLSI Test Symp.*, pp. 277-286, 2007.
- [2] S.R. Nassif, N. Mehta, and Y. Cao, "A resilience roadmap," *ACM/IEEE Design Automation and Test in Europe Conf. (DATE)*, pp. 1011-1016, 2010.
- [3] D. Zhu, R. Melhem, and D. Mosse, "The effects of energy management on reliability in real-time embedded systems," *ACM/IEEE Int. Conf. on Computer-Aided Design (ICCAD)*, pp. 35-40, 2004.
- [4] Z. Baoxian, H. Aydin, and D. Zhu, "Reliability-aware dynamic voltage scaling for energy-constrained real-time embedded systems," *IEEE Conf. Computer Design (ICCD)*, pp. 633-639, 2008.
- [5] A. DeHon, H.M. Quinn, and N.P. Carter, "Vision for cross-layer optimization to address the dual challenges of energy and reliability," *ACM/IEEE Design Automation and Test in Europe Conf. (DATE)*, pp. 1017-1022, 2010.
- [6] D. Bol, R. Ambroise, D. Flandre, and J.-D. Legat, "Interests and limitations of technology scaling for subthreshold logic," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI)*, vol. 17, no. 10, pp. 1508-1519, 2009.
- [7] W.J. Dally and B.P. Towles, *Principles and Practices of Interconnection Networks*, Morgan Kaufmann, 2004.
- [8] G. De Micheli and L. Benini, *Networks on Chip*, Morgan Kaufmann, 2006.
- [9] J. Srinivasan, S.V. Adve, P. Bose, J. Rivers, and C.K. Hu, "RAMP: a model for reliability aware microprocessor design," *IBM Research Report*, RC23048, 2003.
- [10] J. Srinivasan, S.V. Adve, P. Bose, and J.A. Rivers, "The case for lifetime reliability-aware microprocessors," *ACM/IEEE Int. Symp. on Computer Architecture (ISCA)*, pp. 276-287, 2004.
- [11] J. Srinivasan, S. Adve, P. Bose, and J. Rivers, "Exploiting structural duplication for lifetime reliability enhancement," *ACM/IEEE Int. Symp. on Computer Architecture (ISCA)*, pp. 520-531, 2005.
- [12] J. Shin, V. Zyuban, Z. Hu, J. Rivers, and P. Bose, "A framework for architecture-level lifetime reliability modeling," *IEEE/IFIP Int. Conf. on Dependable Systems and Networks (DSN)*, pp. 534-553, 2007.
- [13] A.K. Coskun, T.S. Rosing, K. Mihic, G.D. Micheli, and Y. Leblebici, "Analysis and optimization of MPSoC reliability," *Journal of Low Power Electronics*, vol. 2, no. 1, pp. 56-69, Apr. 2006.
- [14] T.S. Rosing, K. Mihic, and G. De Micheli, "Power and reliability management of SoCs," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI)*, vol. 15, no. 4, pp. 391-403, 2007.
- [15] Z. Gu, C. Zhu, L. Shang, and R.P. Dick, "Application-specific MPSoC reliability optimization," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI)*, vol. 16, no.5, pp. 603-608, May 2008.
- [16] E. Karl, D. Blaauw, D. Sylvester, and T. Mudge, "Multi-mechanism reliability modeling and management in dynamic systems," *IEEE Trans. on Very Large Scale Integration Systems (TVLSI)*, vol. 16, no. 4, pp. 476-487, Apr. 2008.
- [17] Q. Gan and B.E. Helvik, "Dependability modelling and analysis of networks as taking routing and traffic into account," *IEEE Conf. on Next Generation Internet Design and Engineering (NGI)*, pp. 8-32, 2006.
- [18] J.P.G. Sterbenz, D. Hutchison, E. Cetinkaya, A. Jabbar, J.P. Rohrer, M. Scholler, and P. Smith, "Resilience and survivability in communication networks: strategies, principles, and survey of disciplines," *Computer Networks*, pp. 1245-1265, 2010.
- [19] A. Dalirani, M. Hosseinabady, and Z. Navabi, "An analytical model for reliability evaluation of NoC architectures," *IEEE Int. On-Line Testing Symposium (IOLTS)*, pp. 49-56, 2007.
- [20] T. Lehtonen, P. Liljeberg, and J. Plosila, "Fault tolerance analysis of NoC architectures," *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, pp. 361-364, 2007.
- [21] H. Elmiligi, A.A. Morgan, M.W. El-Kharashi, and F. Gebali, "A reliability-aware design methodology for Networks-on-Chip applications," *IEEE Int. Conf. on Design and Technology of Integrated Systems in Nanoscale Era*, pp. 107-112, 2009.
- [22] J.H. Stathis, "Reliability limits for the gate insulator in CMOS technology," *IBM Journal of Research and Development*, vol. 46, pp. 265-286, 2002.
- [23] D.K. Schroder and J.A. Babcock, "Negative bias temperature instability: road to cross in deep submicron silicon semiconductor manufacturing," *Journal of Applied Physics*, vol. 94, no. 1, pp. 1-18, July 2003.
- [24] Jayanth Srinivasan, *Lifetime reliability aware microprocessors*, Ph.D. Thesis, University of Illinois at Urbana-Champaign, 2006.
- [25] E. Wu, J. Sune, W. Lai, E. Nowak, J. McKenna, A. Vayshenker, and D. Harmon, "Interplay of voltage and temperature acceleration of oxide breakdown for ultra-thin gate oxides," *Solid-State Electronics*, vol. 46, no. 11, pp. 1787-1798, May 2002.
- [26] S. Zafar, B. Lee, J. Stathis, A. Callegar, and T. Ning, "A model for negative bias temperature instability (NBTI) in oxide and high k pFETs," *Int. Symposium on VLSI Technology*, pp. 208-209, 2004.
- [27] S.R. Vangal et al., "An 80-tile sub-100-W TeraFLOPS processor in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 3, no.1, pp. 29-41, Jan. 2007.
- [28] S. Bell et al., "TILE64 - processor: a 64-Core SoC with mesh interconnect," *IEEE SSSC*, pp. 588-598, 2008.
- [29] E. Salminen, A. Kulmala, and T.D. Hamalainen, "Survey of Network-on-Chip proposals," *White Paper*, OCP-IP, 2008.
- [30] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar, "A 5-GHz mesh interconnect for a teraflops processor," *IEEE Micro*, vol.27, no.5, pp. 51-61, Sep.-Oct. 2007.
- [31] B. Li, L.-S. Peh, and P. Patra, "Impact of process and temperature variations on Network-on-Chip design exploration," *IEEE Int. Symposium on Networks-on-Chip (NOCS)*, pp. 117-126, 2008.
- [32] N. Binkert, B. Beckmann, G. Black, S.K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D.R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewall, M. Shoaib, N. Vaish, M. D. Hill, and D.A. Wood, "The gem5 simulator," *ACM SIGARCH Computer Architecture News Archive*, vol. 39, no. 2, May 2011.
- [33] N.L. Binkert, R.G. Dreslinski, L.R. Hsu, K.T. Lim, A.G. Saidi, and S.K. Reinhardt, "The M5 simulator: modeling networked systems," *IEEE Micro*, pp. 52-60, July/August 2006.
- [34] M.M.K. Martin, D.J. Sorin, B.M. Beckmann, M.R. Marty, M. Xu, A.R. Alameldeen, K.E. Moore, M.D. Hill, and D.A. Wood, "Multifacet's general execution-driven multiprocessor simulator (GEMS) toolset," *Computer Architecture News (CAN)*, Sep. 2005.
- [35] N. Agarwal, L.-S. Peh, and N. Jha, "GARNET: a detailed interconnection network model inside a full-system simulation framework," *CE-PO8-001*, Princeton University, Feb. 2008.
- [36] S. Li, J.H. Ahn, R.D. Strong, J.B. Brockman, D.M. Tullsen, and N.P. Jouppi, "McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures," *IEEE/ACM Int. Symposium on Microarchitecture (MICRO)*, pp. 469-480, 2009.
- [37] R.D. Strong, <https://bitbucket.org/rickshin/m5-mcpat-parser/src/b7aec9f64b82/m5-mcpat-parse-se.py>.
- [38] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron and M.R. Stan, "HotSpot: a compact thermal modeling method for CMOS VLSI systems," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 5, pp. 501-513, May 2006.
- [39] HotSpot source code, 2012, <http://lava.cs.virginia.edu/HotSpot>
- [40] X. Li, J. Qin, and J.B. Bernstein, "Compact modeling of MOSFET wearout mechanisms for circuit-reliability simulation," *IEEE Trans. on Device and Materials Reliability*, vol. 8, no. 1, pp. 98-121, March 2008.
- [41] M. Gebhart, J. Hestness, E. Fatehi, P. Gratz, and S.W. Keckler, "Running PARSEC 2.1 on M5," The University of Texas at Austin, *Technical Report TR-09-32*, Oct. 2009.
- [42] L. Shang, L.-S. Peh, A. Kumar, and N.K. Jha, "Thermal modeling, characterization and management of on-chip networks," *Int. Symp. Microarchitecture*, Dec. 2004.