Bayesian models for early cross-layer reliability analysis and design space exploration

(Article begins on next page)

20 April 2024

Bayesian models for early cross-layer reliability analysis and design space exploration

Alessandro Vallero, Alessandro Savino, Alberto Carelli and Stefano Di Carlo

Dipartimento di Automatica ed Informatica

Politecnico di Torino

Torino, Italy

Contact: stefano.dicarlo@polito.it

Abstract—Designing soft-errors resilient systems is a complex engineering task, which nowadays follows a cross-layer approach. It requires a careful planning for different fault-tolerance mechanisms at different system's layers: starting from the technology up to the software domain. While these design decisions have a positive effect on the reliability of the system, they usually have a detrimental effect on its size, power consumption, performance and cost. Design space exploration for cross-layer reliability is therefore a multi-objective search problem in which reliability must be traded-off with other design dimensions.

Assessing the reliability of a complex system and performing design space exploration in the early phases of the design cycle is a complex task and designers are increasing looking at stochastic models able to provide fast results to quickly drive early design decisions. This paper summarizes some of the results achieved by the authors in more than five years of research in this domain.

Index Terms—cross-layer reliability, soft errors, design space exploration, radiations

I. INTRODUCTION

Today's computing is a true continuum that ranges from smartphones to mission-critical data center machines, and from desktops to automobiles, with a total market of more than two billion devices per year [1]. We are facing a radical change compared to past business and technical development models. Well defined computing segments (e.g., embedded systems or High Performance Computing) that were driven by separated players and exploited different technologies see now the same solutions and providers acting across all computing segments [1], [2].

Although several design parameters (e.g., performance, power, etc.) have benefited from this continuum of technologies, reliability remains a main issue. Reliability requirements significantly vary across markets and designs, thus creating tensions [3]. Cost-effective techniques to meet varying requirements with the same or derived designs are critical and design reuse across market segments becomes complex. Moreover, the market is not the only factor to consider. The perception of reliability is another key factor. It is not just about the reliability requirements (high, medium, or low), but about consequences of failures and how users perceive the consequences. This creates a context that impacts the design choices.

Several techniques to handle reliability at different abstraction layers have been proposed over the years. At the process layer, transistor architecture/geometry [4], doping details [5]

and FinFET fin height [6] have been extensively explored. At the circuit level, radiation resistant circuits [7], Razor latches [8], [9], tunable replica circuits [10] and LEAP-DICE designs [11] are just a few examples of proposed solutions. At the architecture/micro architecture level, solutions such as parity, Error Correcting Codes (ECC) [12], Triple Modular Redundancy (TMR) [13], lockstep execution [14], and watchdogs [15], [16] are extensively used in commercial products. Eventually, software solutions such as Error Detection by Duplicated Instructions (EDDI) [17], [18], Control Flow Checking [19], [20] and Algorithm Based Fault Tolerance [21] first introduced several years ago are experiencing a new wave of popularity given their simple implementation effort.

However, reliability does not stand alone. There are high costs associated with unreliable products (i.e., field returns, reputation) but also high costs associated with over-designing to provide high reliability (e.g., performance, power, area, etc.) that overall represent the so called reliability tax.

Finding the sweet spot is hard. The questions for the designer today are: *how can we help alleviating the reliability "tax"? Do we really need to protect everything? Do we really need to protect everything all the time? Can reliability mechanisms be re-purposed when not needed?*

In the last years, cross-layer resilience has been referred to as the path to optimal reliability solutions. In a cross-layer resilient system, error management (i.e., detection, diagnosis, reconfiguration, recovery and adaptation) is performed by a combination of hardware and software protection techniques implemented at different layers of the system stack [22]–[26]. Cross-Layer resiliency has the potential to address multiple fault types. It can minimize the reliability "tax" by amortizing it across the system stack. However, despite lot of promises and work done at the academic level, there is still limited impact on the market. Several problems need to be considered to move cross-layer resilience from the research domain to the industrial domain:

- hardware and software vendors have completely different business models and pushing them into a competition could be risky;
- the vendors must own influence to the entire system stack;
- HW/SW vendors must introduce reliability solutions that can be tuned or fully disabled for markets where not needed.

Overall, a cross-layer holistic design approach has several advantages compared to traditional single layer techniques, but it increases the complexity of the design process since a larger design space must be explored. This translates into an increasing demand for system-level reliability analysis frameworks able to evaluate different combinations of cross-layer error protection techniques early in the design cycle and to perform design space exploration (DSE) efficiently [27], [28]. Unfortunately, such tools still lack maturity, especially compared to those available to optimize other design parameters such as power and performance.

In the following sections, the results of more than five years of research by the authors both in the reliability analysis and the design space exploration domains.

II. STOCHASTIC TECHNIQUES FOR EARLY CROSS LAYER RELIABILITY ANALYSIS AND DESIGN SPACE EXPLORATION

Figure 1 depicts our cross-layer reliability design framework. It provides two main functionalities: a fast and accurate model to evaluate the reliability of complex systems early in the design time and an efficient multi-objective DSE system supporting the designers during the initial design choices. The framework focuses on Radiation Induced Failures (RIF) leading to soft errors in memory structures.

The core of the framework is a stochastic Bayesian reliability model. We strongly believe that the complexity of next generation computing systems requires stochastic approaches able to handle the complexity of the modeling task.

A. Reliability Analyzer

Creating frameworks for cross-layer reliability analysis is difficult. They require to integrate data generated by different design teams.

Register Transfer Level (RTL) or gate level fault injection is among the most accurate tools to perform accurate reliability analysis [36], [37]. Nevertheless, even using statistical fault injection, the complexity of the required simulations (especially when considering large memory arrays) is too high to allow the analysis of several cross-layer combinations of error mitigation mechanisms. This is a critical issue in the early design phases when fast evaluations are required to take informed design decisions. Moreover, critical elements of the systems such as the operating system, drivers and filesystems are hard to model in an RTL simulation environment.

To overcome these limitation, in our research we developed a full framework named SyRA (System Reliability Analyzer) able to analyze the impact of radiation induced soft errors in the memory arrays of a complex computing cores (i.e., microprocessor, GPUs) [25], [34]. SyRA has been created to support designers in the early phases of the design, considering all layers of a system from the hardware up to the application software (including the operating system). SyRA exploits a multi-level hybrid Bayesian model to describe the target system and to estimate different reliability metrics. The construction of the system is based on simulations at the different abstraction levels. This allows us to speed up the

analysis and therefore to cope with the complexity of the simulation of the full software stack. SyRA can compute several reliability metrics including Architecture Vulnerability Factor (AVF), Failures In Time (FIT) rate, and Executions Per Failure (EPF). The last metric enables the designer to trade-off reliability and performance in a single measure providing a valuable tool to optimize a computing system.

The complete tool-chain developed to build the model is described in [25], [34] and an example of the accuracy of the analysis performed by SyRA is reported in Figure 2. The proposed framework scales efficiently with the complexity of the system. On average it is 68% faster than full micro-architecture level fault injection and two orders of magnitude faster than RTL fault injection while maintaining a comparable accuracy [38].

B. Design Space Exploration

Bayesian inference supported by SyRA enables speculation on the effects that different protection mechanisms have on the system. This feature has been used to build ReDO (Reliability Design Optimizer), a DSE framework to build soft error resilient computing systems [35]. ReDO is designed to support the early phase of the design of a computing system. It evaluates the application of selected classes of cross-layer soft error protection techniques taking into account multiple design objectives. Exploiting this framework during the design phase, reliability can be traded-off with other design constraints, i.e., hardware area, software size, performance and power consumption.

ReDO internally models the target system resorting to our proposed reliability Bayesian model [25], [34]. This model provides a very compact component based representation of the system stack (from the fabrication technology up to the software layer).

On top of the reliability model of the system, ReDO builds a new exploration heuristic, based on the extremal optimization (EO) theory [39]. The heuristic is designed to efficiently explore the design space composed of different combinations of cross-layer protection mechanisms applied to the components of the system. The goal of the EO is to optimize a global variable by improving local variables that involve co-evolutionary avalanches. This is important in a cross-layer reliability scenario in which we want to evaluate how the application of different combinations of local protection mechanisms in selected components of the system (improvements of local variables) affect the global characteristics of the system in terms of reliability combined with power, area and performance (global variable).

The combination of the proposed reliability model with the DSE heuristic supports the analysis of a complex system in a limited computation time. This makes ReDO an interesting option to support designers in the early phases of the design cycle, when strategic decisions must be taken to design highly optimized systems.

The full framework is described in [35]. Figure 3 shows an example of the AVF improvement that ReDO can achieve
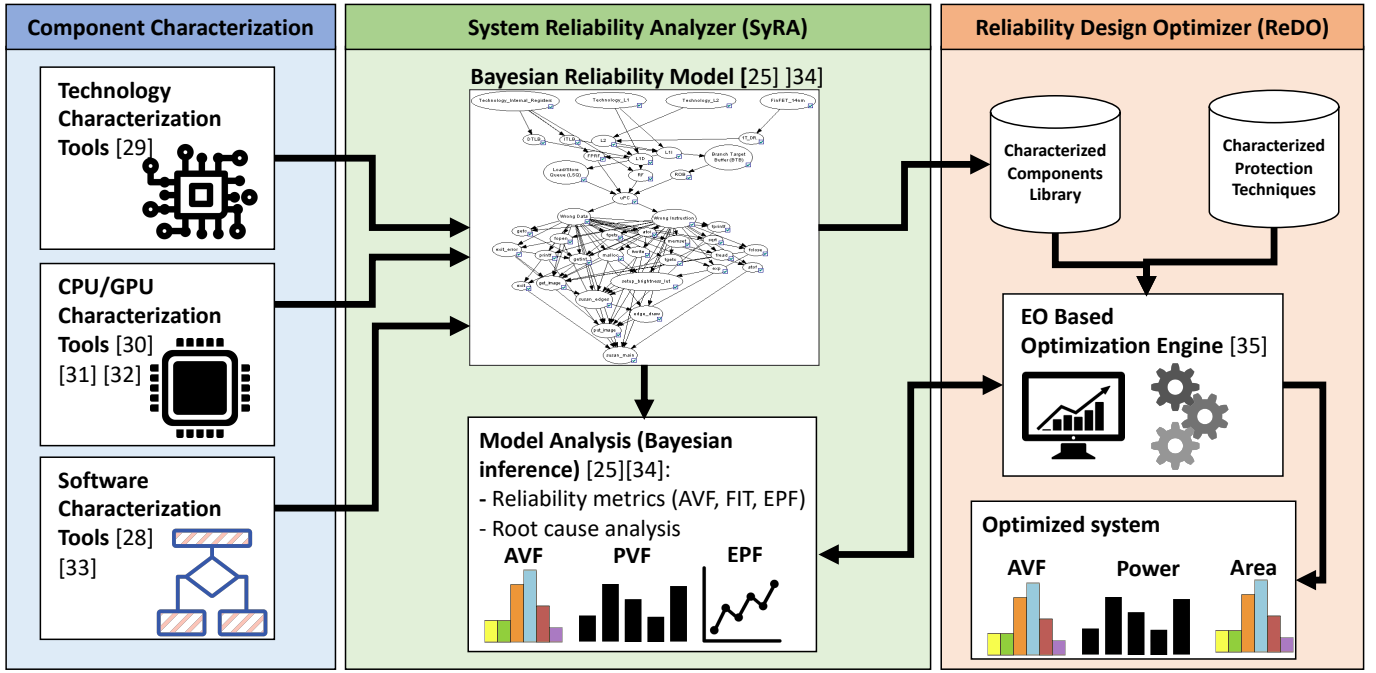
Fig. 1. Overview of our complete cross-layer reliability framework. The component characterization toolset integrates a set of characterization tools for technologies [29], CPUs [30], GPUs [31] [32] and software routines [28] [33]. The tools are used to build the Bayesian Reliability model that is at the core of our System Reliability Analyzer (SyRA) [25], [34]. Eventually, the reliability model is exploited by our Reliability Design Optimizer (ReDO) [35] to evaluate several combinations of cross-layer mechanism and to trade-off them with other design parameters such as Power and Area.
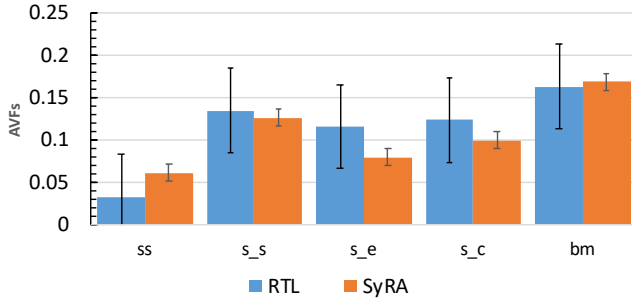

Fig. 2. Adapted from [34]. Results obtained using SyRA to estimate the AVF of five different applications executed an ARM Cortex A9. The figure compares estimation provided by SyRA with those obtained using precise RTL fault injection. The full experimental setup is described in [34].
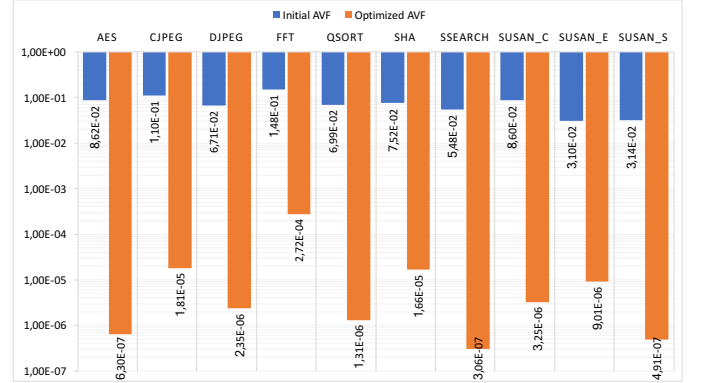

Fig. 3. Results obtained using ReDO to optimize different microprocessor based systems. The full experimental setup is described in [35].

for different software benchmarks when given the freedom to play with different microprocessor architectures and different HW/SW fault tolerance mechanisms.

III. CONCLUSIONS

Reliability is a critical vector for the whole compute continuum. Understand the requirements of the different markets and find cost-effective solutions to address reliability in an ever-challenging space is a key goal to drive down costs, increase innovation and accelerate time to market.

One of the primary need to reach this goal is to support designers with dedicated frameworks able to easy the task of reducing the reliability "tax" driving the available resources toward the implementation of highly optimized systems.

The framework presented in this paper is an example going in this direction that may work as a stimulus for future researches in the field.

REFERENCES

[1] D. Buchholz and I. J. Dunlop, "The future of enterprise computing: Preparing for the compute continuum," *IT@ Intel White Paper, Intel IT*, 2011.

[2] S. Di Carlo, A. Vallero, D. Gizopoulos, G. Di Natale, A. Gonzalez, R. Canal, R. Mariani, M. Pipponzi, A. Grasset, P. Bonnot, F. Reichenbach, G. Rafiq, and T. Loekstad, "Cross-layer early reliability evaluation:

Challenges and promises," in *2014 IEEE 20th International On-Line Testing Symposium (IOLTS)*, July 2014, pp. 228–233.

[3] A. Biswas, "Cost-effective reliability trade-offs and challenges," [Online] http://www.selse.org/wp-content/uploads/2015/09/SELSE_2018\ _Keynote_abiswas.pptx, April 2018.

[4] S. Ramey, A. Ashutosh, C. Auth, J. Clifford, M. Hattendorf, J. Hicks, R. James, A. Rahman, V. Sharma, A. St Amour, and C. Wiegand, "Intrinsic transistor reliability improvements from 22nm tri-gate technology," in *2013 IEEE International Reliability Physics Symposium (IRPS)*, April 2013, pp. 4C.5.1–4C.5.5.

[5] N. Rezzak, M. L. Alles, R. D. Schrimpf, S. Kalemeris, L. W. Massengill, J. Sochacki, and H. J. Barnaby, "The sensitivity of radiation-induced leakage to sti topology and sidewall doping," *Microelectronics Reliability*, vol. 51, no. 5, pp. 889–894, 2011.

[6] C.-Y. Su, M. Armstrong, L. Jiang, S. Kumar, C. Landon, S. Liu, I. Meric, K. Park, L. Paulson, K. Phoa *et al.*, "Transistor reliability characterization and modeling of the 22ffl finfet technology," in *2018 IEEE International Reliability Physics Symposium (IRPS)*. IEEE, 2018, pp. 6F–8.

[7] V. Sharma and A. Rajawat, "Review of approaches for radiation hardened combinational logic in cmos silicon technology," *IETE Technical Review*, vol. 35, no. 6, pp. 562–573, 2018.

[8] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner *et al.*, "Razor: A low-power pipeline based on circuit-level timing speculation," in *Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture*. IEEE Computer Society, 2003, p. 7.

[9] M. Hosseinabady, P. Lotfi-Kamran, G. Di Natale, S. Di Carlo, A. Benso, and P. Prinetto, "Single-event upset analysis and protection in high speed circuits," in *Eleventh IEEE European Test Symposium (ETS'06)*, May 2006, pp. 29–34.

[10] J. 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 *2009 Symposium on VLSI Circuits*, June 2009, pp. 112–113.

[11] L. H.-H. Kelin, L. Klas, B. Mounaim, R. Prasanthi, I. R. Linscott, U. S. Inan, and M. Subhasish, "Leap: Layout design through error-aware transistor positioning for soft-error resilient sequential cell design," in *2010 IEEE International Reliability Physics Symposium*. IEEE, 2010, pp. 203–212.

[12] M. Fabiano, M. Indaco, S. Di Carlo, and P. Prinetto, "Design and optimization of adaptable bch codecs for nand flash memories," *Microprocessors and Microsystems*, vol. 37, no. 4-5, pp. 407–419, 2013.

[13] S. Hudson, R. S. Sundar, and S. Koppu, "Fault control using triple modular redundancy (tmr)," in *Progress in Computing, Analytics and Networking*. Springer, 2018, pp. 471–480.

[14] Á. B. d. Oliveira, "Applying dual core lockstep in embedded processors to mitigate radiation induced soft errors," 2017.

[15] A. Benso, S. Di Carlo, G. Di Natale, and P. Prinetto, "A watchdog processor to detect data and control flow errors," in *9th IEEE On-Line Testing Symposium, 2003. IOLTS 2003.*, July 2003, pp. 144–148.

[16] S. Di Carlo, G. Di Natale, and R. Mariani, "On-line instruction-checking in pipelined microprocessors," in *2008 17th Asian Test Symposium*. IEEE, 2008, pp. 377–382.

[17] N. Oh, P. P. Shirvani, and E. J. McCluskey, "Error detection by duplicated instructions in super-scalar processors," *IEEE Transactions on Reliability*, vol. 51, no. 1, pp. 63–75, 2002.

[18] A. Benso, S. Di Carlo, G. Di Natale, P. Prinetto, and L. Tagliaferri, "Data criticality estimation in software applications," in *International test conference*, 2003, pp. 802–810.

[19] N. Oh, P. P. Shirvani, and E. J. McCluskey, "Control-flow checking by software signatures," *IEEE transactions on Reliability*, vol. 51, no. 1, pp. 111–122, 2002.

[20] A. Benso, S. Di Carlo, G. Di Natale, P. Prinetto, and L. Tagliaferri, "Control-flow checking via regular expressions," in *Proceedings 10th Asian Test Symposium*. IEEE, 2001, pp. 299–303.

[21] P. Banerjee, J. T. Rahmeh, C. Stunkel, V. Nair, K. Roy, V. Balasubramanian, and J. A. Abraham, "Algorithm-based fault tolerance on a hypercube multiprocessor," *IEEE Transactions on Computers*, vol. 39, no. 9, pp. 1132–1145, 1990.

[22] A. DeHon, N. Carter, and H. Quinn, "Final report for ccc cross-layer reliability visioning study," [Online] http://www.relxlayer.org/FinalReport?action=AttachFile\&do=get\&target=final_report.pdf, March 2011.

[23] J. Henkel, L. Bauer, H. Zhang, S. Rehman, and M. Shafique, "Multi-layer dependability: From microarchitecture to application level," in *2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2014, pp. 1–6.

[24] E. Cheng, S. Mirkhani, L. G. Szafaryn, C. Y. Cher, H. Cho, K. Skadron, M. R. Stan, K. Lilja, J. A. Abraham, P. Bose, and S. Mitra, "Clear: Cross-layer exploration for architecting resilience: Combining hardware and software techniques to tolerate soft errors in processor cores," in *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2016, pp. 1–6.

[25] A. Vallero, A. Savino, G. Politano, S. Di Carlo, A. Chatzidimitriou, S. Tselonis, M. Kaliorakis, D. Gizopoulos, M. Riera, R. Canal, A. Gonzalez, M. Kooli, A. Bosio, and G. Di Natale, "Cross-layer system reliability assessment framework for hardware faults," in *2016 IEEE International Test Conference (ITC)*, Nov 2016, pp. 1–10.

[26] A. Vallero, A. Savino, S. Tselonis, N. Foutris, M. Kaliorakis, G. Politano, D. Gizopoulos, and S. D. Carlo, "A bayesian model for system level reliability estimation," in *Test Symposium (ETS), 2015 20th IEEE European*, May 2015, pp. 1–2.

[27] D. W. Coit, T. Jin, and N. Wattanapongsakorn, "System optimization with component reliability estimation uncertainty: a multi-criteria approach," *IEEE transactions on reliability*, vol. 53, no. 3, pp. 369–380, 2004.

[28] A. Vallero, S. Tselonis, N. Foutris, M. Kaliorakis, M. Kooli, A. Savino, G. Politano, A. Bosio, G. Di Natale, D. Gizopoulos *et al.*, "Cross-layer reliability evaluation, moving from the hardware architecture to the system level: A clereco eu project overview," *Microprocessors and Microsystems*, vol. 39, no. 8, pp. 1204–1214, 2015.

[29] M. Riera, R. Canal, J. Abella, and A. Gonzalez, "A detailed methodology to compute soft error rates in advanced technologies," in *Proceedings of the 2016 Conference on Design, Automation & Test in Europe*. EDA Consortium, 2016, pp. 217–222.

[30] M. Kaliorakis, S. Tselonis, A. Chatzidimitriou, N. Foutris, and D. Gizopoulos, "Differential fault injection on microarchitectural simulators," in *2015 IEEE International Symposium on Workload Characterization*. IEEE, 2015, pp. 172–182.

[31] A. Vallero, D. Gizopoulos, and S. Di Carlo, "Sifi: Amd southern islands gpu microarchitectural level fault injector," in *2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, July 2017, pp. 138–144.

[32] S. Tselonis and D. Gizopoulos, "Gufi: A framework for gpus reliability assessment," in *2016 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, April 2016, pp. 90–100.

[33] M. Kooli, G. Di Natale, and A. Bosio, "Cache-aware reliability evaluation through llvm-based analysis and fault injection," in *2016 IEEE 22nd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, July 2016, pp. 19–22.

[34] A. Vallero, A. Savino, A. Chatzidimitriou, M. Kaliorakis, M. Kooli, M. Riera, M. Anglada, G. Di Natale, A. Bosio, R. Canal, A. Gonzalez, D. Gizopoulos, R. Mariani, and S. Di Carlo, "Syra: Early system reliability analysis for cross-layer soft errors resilience in memory arrays of microprocessor systems," *IEEE Transactions on Computers*, vol. 68, no. 5, pp. 765–783, May 2019.

[35] A. Savino, A. Vallero, and S. Di Carlo, "Redo: Cross-layer multi-objective design-exploration framework for efficient soft error resilient systems," *IEEE Transactions on Computers*, vol. 67, no. 10, pp. 1462–1477, Oct 2018.

[36] L. Entrena, M. Garcia-Valderas, R. Fernandez-Cardenal, A. Lindoso, M. Portela, and C. Lopez-Ongil, "Soft error sensitivity evaluation of microprocessors by multilevel emulation-based fault injection," *IEEE Transactions on Computers*, vol. 61, no. 3, pp. 313–322, 2012.

[37] A. Benso, A. Bosio, S. Di Carlo, and R. Mariani, "A functional verification based fault injection environment," in *22nd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT 2007)*. IEEE, 2007, pp. 114–122.

[38] A. Chatzidimitriou, M. Kaliorakis, D. Gizopoulos, M. Iacaruso, M. Pipponzi, R. Mariani, and S. D. Carlo, "Rt level vs. microarchitecture-level reliability assessment: Case study on arm(r) cortex(r)-a9 cpu," in *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*, June 2017, pp. 117–120.

[39] S. Boettcher, "Extremal optimization: heuristics via coevolutionary avalanches," *Computing in Science Engineering*, vol. 2, no. 6, pp. 75–82, Nov 2000.