Special session: How approximate computing impacts verification, test and reliability

Special Session: How Approximate Computing impacts Verification, Test and Reliability

L. Sekanina[1], Z. Vasicek[1], A. Bosio[2], M. Traiola[2], P. Rech[3], D. Oliveria[3], F. Fernandes[3], S. Di Carlo[4]
[1]IT4I, Brno University of Technology, Czechia; [2]LIRMM - France; [3]UFRGS, Brazil; [4]Politecnico di Torino - Italy

I. Introduction

Two AxC techniques have been successfully applied to hardware components. The first one is the *functional approximation* [1]that modifies the circuit structure replacing the original function F with the function G. G implementation leads to area/energy reduction at the cost of reduced accuracy, meaning that some errors can be observed at the outputs of G. The observed errors are a variation between the output values of F (precise) and G (approximate). The variation is the accuracy loss measured by means of quality metric(s) [1]. The second AxC technique is the *over-scaling based approximation*. Basically, the HW component is forced to work outside its specified operating conditions [1]. The classical example is the reduction of the supply voltage under the minimum value.

In the functional approximation, the circuit netlist is modified to introduce the approximation. Thus we have to guarantee that the approximate circuit does not introduce an error greater than the acceptable one. It is therefore required to adapt the "verification" phase to the AxC design flow. We present an approach in which the approximation problem is formulated as a complex multi-objective design and optimization problem and solved using advanced search methods. In order to **exactly** determine the quality (the average error, the worst-case error etc.) of candidate solutions, symbolic methods based on BDD analysis and SAT problem solving are employed. Several case studies in the area of arithmetic circuit approximation will be presented to demonstrate the efficacy and scalability of the search-based approximation combined with formal error analysis [2]. Once the verification phase is done, the AxC design enters in the manufacturing process, where physical defects may impact the produced AxC. This means that the manufactured device may have a different error w.r.t. the verified one. In this context, the role of "testing" is to ensure that the amount of error (due to manufacturing defects) is not greater than the acceptable error threshold. We propose our **AxC aware ATPG** for generating test vectors targeting only the "critical faults" (i.e., faults leading to an unacceptable error) [3]. The benefits are a lower test set (i.e., reduced test time) and a yield improvements. Experimental results are carried out on a public benchmark suite to prove the efficiency of the proposed approach.

AxC exploits the index property of some applications to tolerate the presence of inaccuracy. This inaccuracy is quantified as an error w.r.t to the precise application and quantified by means of quality metrics. Actually the applications are inherently able to tolerate the presence of a certain amount of error because they intrinsically embed redundancy: a higher number of loop iterations or data precision larger than the required. AxC cut this embedded redundancy and on one hand it gains in terms of overhead reduction, but on the other hand it introduces a reduction of the capability to tolerate errors (because of the accuracy reduction). This means that if a resilient application is able to tolerate a certain amount of errors due to harsh environment or aging problems, the approximated application can tolerate only a reduced fraction of errors. For safety critical applications, it is thus very important to identify a trade off between approximation and reliability. Moreover, the use of the over-scaling technique may exacerbate this problem.

We experimentally evaluate the benefits and challenges in terms of reliability introduced by approximate computing. Faults that only slightly impact the code output could be tolerated if an approximate solution is accepted as correct. We have seen through accelerated neutron beam experiments (i.e., the harsh environment) that a 0.1% approximation of the output value allows the application to tolerate up to 90% of radiation-induced transient errors [4]. It is then fundamental to analyze the application and distinguish between tolerable and critical errors. Unfortunately, approximate hardware is likely to increase the device error rate. Errors in the least significant digits of a 32 bit number are much less severe than those in a 16 bit number. Moreover, a fault in a 32 bit hardware used to execute two 16 bit operations is likely to corrupt both half-precision outputs, thus reducing the application reliability. As a result, while the benefits of approximate hardware are unquestionable it is necessary to carefully evaluate the impact of such an hardware in the system reliability.

This work has been partially supported by the IT4Innovations excellence in science project LQ1602.

References

[1] S. Mittal, "A survey of techniques for approximate computing," *ACM Comput. Surv.*, vol. 48, no. 4, pp. 62:1–62:33, Mar. 2016. [Online]. Available: http://doi.acm.org/10.1145/2893356

[2] M. Ceska, J. Matyas, V. Mrazek, L. Sekanina, Z. Vasicek, and T. Vojnar, "Approximating complex arithmetic circuits with formal error guarantees: 32-bit multipliers accomplished," in *Proc. of 36th IEEE/ACM Int. Conf. On Computer Aided Design*. IEEE, 2017, pp. 416–423.

[3] I. Wali, M. Traiola, A. Virazel, P. Girard, M. Barbareschi, and A. Bosio, "Towards approximation during test of integrated circuits," in *2017 IEEE 20th International Symposium on Design and Diagnostics of Electronic Circuits Systems (DDECS)*, April 2017, pp. 28–33.

[4] D. Oliveira, L. Pilla, N. DeBardeleben, S. Blanchard, H. Quinn, I. Koren, P. Navaux, and P. Rech, "Experimental and analytical study of xeon phi reliability," in *Proc. of the Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, ser. SC '17, 2017, pp. 28:1–28:12.