Learning to Approximate Computing at Run-time

Paulo Garcia, Mehryar Emambakhsh, Andrew Wallace

Abstract—**Intelligent sensor/signal processing systems are increasingly constrained by tight power budgets, especially when deployed in mobile/remote environments. Approximate computing is the process of adaptively compromising over the accuracy of a systems output in order to obtain higher performance for other metrics, such as power consumption or memory usage, for applications resilient to inaccurate computations. It is, however, usually statically implemented, based on heuristics and testing loops, which prevents switching between different approximations at run-time. This limits approximation versatility and results in under- or over-approximated systems for the specific input data, causing excessive power usage and insufficient accuracy, respectively. To avoid these issues, this paper proposes a new approximate computing approach by introducing a supervisor block embedding prior knowledge about runtime data. The target system (i.e., signal processing pipeline) is implemented with configurable levels and types of approximations [1]. Data processed by the target system is analysed by the supervisor and the approximation is updated dynamically, by using prior knowledge to establish a confidence measure on the accuracy of the computed results. Moreover, by iteratively evaluating the output, the supervisor block can learn and subsequently update tunable parameters, in order to improve the quality of the results. Our approach also envisions switching between multiple approximation and learning engines at run-time. We detail and evaluate this approach for tracking problem in computer vision. Results show our approach yields promising trade-offs between accuracy and power consumption.**

Index Terms—**field programmable gate array (FPGA), optimisations, power, image processing, dataflow**

I. Introduction

Power/performance trade offs are well established compromises in the design of all embedded systems. In both hardware and software domains, there is a great deal of formal and empirical knowledge which guides system architects towards optimal design time decisions, and myriad runtime operation modes (i.e., power saving modes) controlled locally or remotely. Approximate computing promises unprecedented power savings by introducing a trade off between power and another dimension: accuracy. For applications resiliant to innacurate computations, or where there isn't a single golden result, approximate computing methods can improve traditional design strategies for power reduction: essential in the dark silicon era.

Despite its promise, approximate computing is still an immature technology: a formal model of the impact of approximations on other design metrics does not yet exist. Hence, most approximate computing applications require two premises to be implemented successfully: (a) adequate test data are available, to correctly model the accuracy impact

P. Garcia, M. Emambakhsh and A. Wallace are with the School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh EH14 4AS, U.K. E-mail: {p.garcia, a.m.wallace}@hw.ac.uk.

Manuscript received XXX XX, XXXX; revised XXX XX, XXXX.

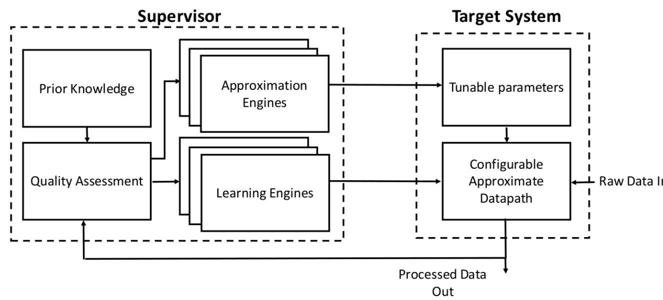

Fig. 1: Block diagram

of approximations; and, (b) approximations are performed iteratively at design time, in order to meet the required power/accuracy goals, and remain static throughoput deployment.

This is a stark contrast to performance/power trade offs, where well established benchmark suites offer near total coverage of application scenarios: in approximate computing, test data that allows adequate modeling of accuracy is often unavailable. In performance/power trade offs, systems can self-tune their operation based on load and run time parameters to dynamically adjust metrics. In approximate computing, approximations are static: mainly because there is no trusted method to determine if accuracy suffices, without access to ground truth. In this paper, we tackle this problem: adjusting the level of approximations at run time, for signal processing applications. Our hypothesis states that prior knowledge about processed data can guide built-in approximation engines, dynamically modifying the level of approximations whilst ensuring that accuracy suffices for the required task.

Specifically, this paper offers the following contributions:

- We introduce the concept of prior knowledge-guided approximations. This represents a statistical measure of approximation impact, unlike test data-based empirical measures prevalent in the state of the art.
- We introduce a model of run time approximations, which use prior knowledge to ensure that accuracy suffices, without access to ground truth, unlike iterative comparisons to ground truth prevalent in the state of the art.
- We describe and evaluate a proof of concept of our approach, using an Extended Kalman Filter for motion tracking, where we have prior knowledge about the target's motion.

II. Background and related work

A. Approximate computing

B. Prior knowledge in sensing

III. Prior knowledge for runtime approximations

IV. Proof of concept: extended Kalman filter

V. Experimental results

VI. Conclusions and future work

Acknowledgment

We acknowledge the support of the Engineering and Physical Research Council, grant references EP/K009931/1 (Programmable embedded platforms for remote and compute intensive image processing applications) and EP/K014277/1 (MOD University Defence Research Collaboration in Signal Processing).

References

[1] V. Vassiliadis, K. Parasyris, C. Chalios, C. D. Antonopoulos, S. Lalis, N. Bellas, H. Vandierendonck, and D. S. Nikolopoulos, "A programming model and runtime system for significance-aware energy-efficient computing," *SIGPLAN Not.*, vol. 50, no. 8, pp. 275–276, Jan. 2015. [Online]. Available: http://doi.acm.org/10.1145/2858788.2688546