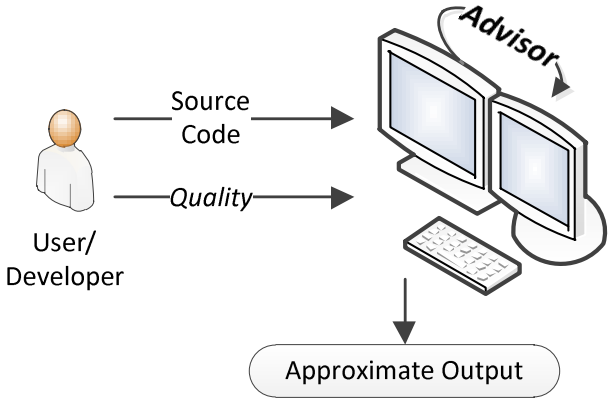
CES Chair for Embedded Systems

Prof. Dr. J. Henkel

Bachelor-/Master Theses

Approximation Advisor Tool for Error Resilient Applications



Computer architecture is evolving with the workloads. Previous shifts in our workloads have led to introduction of alternative computing platforms such as GPUs and accelerators to address the changing characteristics. Modern applications have different characteristics than before: Error resilience.

Many modern application domains including decision making, computer vision, recognition, artificial intelligence (e.g. *deep neural networks*), data mining and synthesis show an intrinsic error tolerance in their computation. This tolerance can be attributed to characteristics such as attenuation of error through statistical methods or iterations. Approximate computing leverages application resilience to significantly improve energy and/or performance.

Existing research works have proposed relaxing synchronizations at task level, skipping non-critical computations at software level and reducing circuit precision/complexity or lowering the operation voltage at hardware level. These techniques are becoming an enabling technology for computing heavy workloads with limited resources (e.g. machine learning on smart phones). Commercial examples of reduced precision computing already started to appear (NVIDIA Tegra 1X & Volta, AMD Polaris, Google tensor processing unit, etc.).

Approximate computing has opened many questions to maximize its benefits. We need to answer when, where to apply and also which technique(s) to choose.

In this thesis we will develop a software tool using **LLVM** compiler to search design space with several approximations techniques and find the best accuracy vs. benefit trade-off. This tool will be useful to users/developers in taking advantage of approximations in an automated manner.

Tasks:

- Development of a software advisor tool for approximate computing with support for a set of approximation techniques.
- Search the design space during compilation in an automated manner.
- Demonstration using existing benchmarks and their analysis.

Skills acquired with the Thesis:

- Compiler development and integration for hardware/software co-design
- Work in a research environment
- Technical writing
- Prior knowledge on Approximate Computing is **not** required

Required Knowledge:

- Compiler background, ideally LLVM
- Programming skills

Helpful skills (not required but helpful):

- Scripting skills (tcl, python, makefile)
- Experience with EDA flow

Start Date:
Immediately or within a couple of months

Supervision:
M.Sc. Tanfer Alan
alan@kit.edu
ces.itec.kit.edu/~alan

M.Sc. Jorge Castro-Godínez,
jorge.castro-godinez@kit.edu
ces.itec.kit.edu/~godinez

Questions? → Feel free to contact us.