

Abstract

Hassan Shokry El-Dib

An SMT Based Method for Optimizing Arithmetic Computations in Embedded Software Code

We present a new method for optimizing the C/C++ code of embedded control software with the objective of minimizing implementation errors in the linear fixed-point arithmetic computations caused by overflow, underflow, and truncation. Our method relies on the use of an SMT solver to search for alternative implementations that are mathematically equivalent but require a smaller bit-width, implementations that use the same bit-width but have a larger error-free dynamic range. Our systematic search of the bounded implementation space is based on an inductive synthesis procedure, which guarantees to find a solution as long as such solution exists. Furthermore, the synthesis procedure is applied incrementally to small code regions - one at a time - as opposed to the entire program, which is crucial for scaling the method to programs of realistic size and complexity. We have implemented our method in a software tool based on the Clang/LLVM compiler and the Yices SMT solver. Our experiments, conducted on a set of representative benchmarks from embedded control and DSP applications, show that the method is both effective and efficient in optimizing fixed-point arithmetic computations in embedded software code.