

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 source 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 the satisfiability modulo theory (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 a new inductive synthesis procedure, which is guaranteed to find a valid solution as long as such solution exists. Furthermore, we propose an incremental optimization procedure, which applies the synthesis procedure only to small code regions-one at a time-as opposed to the entire program, which is crucial for scaling the method up to programs of realistic size and complexity. We have implemented our new method in a software tool based on the Clang/LLVM compiler frontend and the Yices SMT solver. Our experiments, conducted on a set of representative benchmarks from embedded control and digital signal processing applications, show that the method is both effective and efficient in optimizing arithmetic computations in embedded software code.