I. Motivation

"... proving a software system correct requires much more effort, knowledge, training, and ingenuity than writing the software in trial-and-error style."

- E. M. Clarke et al., Handbook of Model Checking 2018.

ESBMC is an open source, permissively licensed (apache 2), bounded model checker (BMC) for C programs. It is written primarily in portable C++ and, using Autotools, builds on multiple platforms.

The tool was developed for bounded model checking of both sequential and concurrent programs using a variety of SMT solvers, and has a proven track record of bug finding in real world applications.

ESBMC also implements a k-induction algorithm to provide proofs of correctness for some unbounded programs.

II. Components & Features

ESBMC now uses clang, a state-of-the-art compiler suite for C/C++/ObjectiveC/Objective++ widely used in industry, as its front-end.

Control-Flow Graph Generator

It takes the program AST and transforms it into an equivalent GOTO program: a simplified representation that consists only of assignments, conditional and unconditional branches, assumptions, and assertions.

Symbolic Execution Engine

ESBMC symbolically executes the GOTO program. It unrolls loops k times, generates the SSA form of the unrolled program, and derives all the safety properties to be checked by the SMT solver.

SMT Back-end

ESBMC’s SMT back-end supports five solvers: Boolector (default), Z3, MathSAT, CVC4 and Yices.

Python API

ESBMC now includes a Python API that reduces the difficulty of prototyping new features and makes the tool internals accessible to a wider audience.

k-Induction

\[ \text{kind}(P, k) = \begin{cases} 1 & \text{if } B(k) \text{ is SAT} \\ 2 & \text{if } B(k) \land [F(k) \lor (k)] \text{ is UNSAT} \\ 0 & \text{otherwise} \end{cases} \]

Floating-point Encoding

ESBMC encodes floating-point arithmetic using:

• bitvectors, which extends the floating-point arithmetic support to all solvers that are currently integrated.

• the SMT theory of floating-points, available only in Z3 and MathSAT.

III. SV-COMP 2018

The ESBMC’s k-induction version achieved a score of 5476 and third place overall.

The k-induction algorithm reported 4301 correct results, with 92% of witnesses being correctly validated.

None of the wrong results were related to the k-induction algorithm.

IV. Future Work

We are extending the k-induction algorithm to reuse information from the inductive step, to make bug finding more efficient.