Bounded Model Checking of C++ Programs
Based on the Qt Framework
Felipe R. M. Sousa, Lucas C. Cordeiro, and Eddie B. de Lima Filho
Electronic and Information Research Centre
Federal University of Amazonas

I. Introduction

- The present work identifies the main Qt features used in real applications and, based on that, creates an operational model, which provides a way to analyse and check properties related to those features.

Bounded Model Checking

ESBMC++

- ESBMC++ is a bounded model checker based on SMT solvers, which is used for ANSI-C/C++ single- and multi-threaded programs. Properties checked:
  - arithmetic under- and overflow, division by zero, out-of-bounds index, pointer safety, deadlocks, and data races, and assertions defined by user.

Pre-conditions

- A condition that must be fulfilled before a function or method can be executed.
  - Checked through assertions in the operational model;
  - When the a method/function is called, ESBMC++ interprets its behaviour as implemented in the operational model.

Post-conditions

- An assertion that characterizes the state of the program immediately after execution of a certain function or method.

III. Experimental Evaluation

- Currently, esbmc-qt test suite contains 52 benchmarks, which take about 48 seconds to be verified.

- CONCLUSIONS

  - This paper proposes an approach to verify C++/Qt programs using an operational model.
  - The experimental results show the efficiency of this approach for verifying Qt programs and present, for the developed test suite, a success rate of 94.45%.
  - As future work, more classes and libraries will be integrated into the operational model, in order to increase Qt framework coverage and validate its properties.