Dietrich Geisler

Zvonimir Rakamarić

FORMAL VERIFICATION OF THE IEEE FLOATING-POINT TYPE IN BOOGIE AND SMACK

Dietrich Geisler (Zvonimir Rakamarić)

School of Computing; College of Engineering

Throughout academia and industry, formal verification techniques have become essential for asserting code correctness and identifying potential bugs. Despite the growth of this field, however, there exists relatively few verifiers that can reason about the floating-point type. Since many areas of study involve code that relies on the correctness of the floating-point type, the lack of verifiers which can reason about the floating-point type is a cause for concern.

We present the introduction of the floating-point data type to the Boogie[1] and SMACK[2] tools. In particular, we have introduced the floating-point type to the Boogie language and implement support for evaluating floating-points using the Boogie and SMACK tools.

The addition of the floating-point type to Boogie allows mathematical reasoning about the floating-point type in the context of the Boogie type system and prover, including analysis of floating-points of arbitrary exponent and significand size. The subsequent addition of the floating-point type to SMACK allows the reasoning of floating-points in the C and C++ languages, including reasoning about memory safety and parallelization through the SMACK evaluation framework.

SMACK was recently entered into the 2017 Software Verification Competition (SVCOMP) in a variety of categories including the floating-point verification category.[3] Overall, SMACK performed very well, placing first in the primary category of Reach Safety and second overall. In the floating-point category, a subset of the Reach Safety category, SMACK placed 7th with a score of 153/314, scoring higher than other major verifiers such as UAutomizer and CPAChecker.

[1] https://github.com/boogie-org/boogie
[2] https://github.com/smackers/smack
[3] https://sv-comp.sosy-lab.org/2017/results/results-verified/