Divider and Unique Inverse Benchmarks Submitted to the SAT Competition 2018

Armin Biere
Institute for Formal Models and Verification
Johannes Kepler University Linz

Our benchmark submission for the SAT 2018 Competition consist of two sets of word-level properties originally formulated as SMT problems in the quantifier-free theory of bit-vectors in BTOR [1] or SMTLIB [2] format. We then use our SMT solver Boolector [3] to synthesize AIGs [4], which in turn were translated to DIMACS format.

Division

The first set specifies word-level (modulo 2^n) division using multiplication for various bit-widths n in BTOR format [1].

We consider both *unsigned* and *signed* dividers. For unsigned division we check validity over unsigned n-bit bit-vectors ("$/_u$" denotes unsigned division):

$$y \neq 0 \quad \Rightarrow \quad (x - (x \;/_u\; y) \cdot y) \;<_u\; y$$

As common in bit-vector logics arithmetic operators take two n-bit bit-vectors as input and produce one n-bit bit-vector as output, with the effect, that there is no difference between signed and unsigned versions of multiplication nor subtraction.

For signed division it is more complicated and we have to take signs into account (now "$/_s$" denotes signed division):

$$y \neq 0 \quad \Rightarrow \quad |x - (x \;/_s\; y) \cdot y| \;<_u\; |y|$$

where "| |" is actually implemented with an if-then-else operator testing the argument to be smaller than zero (using signed "$<_s$" comparison) and if so negating it (two-complement). These signed benchmarks are as a consequence much harder.

Inversion

The second set of benchmarks checks that bit-vector multiplication modulo 2^n has unique inverses for odd numbers, which translates to the following SMT benchmark for $n = 32$ in SMTLIB format [2]:

```
(set-logic QF_BV)
(declare-fun x () (_ BitVec 32))
(declare-fun y () (_ BitVec 32))
(declare-fun z () (_ BitVec 32))
(assert (= (bvmul x y) (bvmul x z)))
(assert ((_ extract 0 0) x))
(assert (distinct y z))
(check-sat)
(exit)
```

References

[1] R. Brummayer, A. Biere, and F. Lonsing, "BTOR: Bit-precise modelling of word-level problems for model checking," in *Proceedings of the Joint Workshops of the 6th International Workshop on Satisfiability Modulo Theories and 1st International Workshop on Bit-Precise Reasoning*, ser. SMT '08/BPR '08. New York, NY, USA: ACM, 2008, pp. 33–38.

[2] C. Barrett, P. Fontaine, and C. Tinelli, "The SMT-LIB Standard: Version 2.6," Department of Computer Science, The University of Iowa, Tech. Rep., 2017, available at www.SMT-LIB.org.

[3] A. Niemetz, M. Preiner, C. Wolf, and A. Biere, "Btor2, BtorMC and Boolector 3.0," in *Computer Aided Verification - 30th International Conference, CAV 2018*, ser. Lecture Notes in Computer Science. Springer, 2018, to appear.

[4] A. Biere, "The AIGER And-Inverter Graph (AIG) format version 20071012," FMV Reports Series, Institute for Formal Models and Verification, Johannes Kepler University, Altenbergerstr. 69, 4040 Linz, Austria, Tech. Rep., 2007.

Supported by FWF, NFN Grant S11408-N23 (RiSE).