Boolector at the SMT Competition 2017

Aina Niemetz, Mathias Preiner, and Armin Biere

Institute for Formal Models and Verification
Johannes Kepler University, Linz, Austria

***Abstract*—This paper serves as solver description for our SMT solver Boolector as entered into the SMT Competition 2017. We only list important differences to the version of Boolector that entered the SMT Competition 2016 [5]. For further and more detailed information, we refer to [3, 4, 7] or source code.**

Overview

For the SMT competition 2017, we improved several key components and features of our SMT solver Boolector [4]. In 2016, we added experimental support for quantified bit-vectors and at the SMT competition 2016, for the first time Boolector entered the BV and UFBV divisions of the main track. Since then, we considerably improved Boolector's quantified bit-vector engine [7, 8], which is now stable but currently does not support uninterpreted functions (UF). It enters the BV division of the main track of SMT-COMP'17 and will be included in the upcoming (and all future) official release(s) of Boolector.

In 2016, we also introduced an additional engine for quantifier-free bit-vectors, which implements a novel propagation-based local search approach [6] without bit-blasting. A combination of this approach with bit-blasting within a sequential portfolio setting as described in [3, 6] entered SMT-COMP'16 as configuration *Boolector preprop*. For the SMT competition 2017, this combination is our default approach for the QF_BV division of the main track.

At the SMT competition 2017, additionally to the default configuration *Boolector*, Boolector enters the QF_BV main track in an experimental configuration *Boolector+CaDiCaL*, which uses version sc17 of our novel SAT solver CaDiCaL [2] as back-end. The default configuration *Boolector*, which also enters the division QF_ABV, QF_UFBV, QF_AUFBV and BV, uses an internal version (version bbe) of our SAT solver Lingeling [1], which is identical to the version submitted to the SAT competition 2017, as back-end.

Configurations

At the SMT competition 2017, we submitted two configurations of our SMT solver Boolector: *Boolector* and *Boolector+CaDiCaL*. For both configurations, we enabled *unconstrained optimization*.

Boolector

This configuration of Boolector enters the QF_BV, QF_UFBV, QF_ABV, QF_AUFBV, and BV divisions of the main track and uses our SAT solver Lingeling [1] (version bbe) as back-end.

Boolector+CaDiCaL

This configuration of Boolector enters only the QF_BV division of the main track and uses our novel SAT solver CaDiCaL [2] (version sc17) as back-end.

Copyright

Boolector has been originally developed by Armin Biere and Robert Brummayer at the FMV institute of the Johannes Kepler University Linz. Since 2009 it was maintained and extended by Armin Biere. Since 2012 it is maintained and extended by Armin Biere, Aina Niemetz, and Mathias Preiner.

License

For the competition version of Boolector we use the same license scheme as introduced in 2013 for our SAT solver Lingeling [1]. It allows the use of the software for academic, research and evaluation purposes. It further prohibits the use of the software in other competitions or similar events without explicit written permission. Please refer to the actual license, which comes with the source code, for more details.

References

[1] A. Biere. Lingeling, Plingeling and Treengeling entering the SAT Competition 2013. In A. Belov, M. Heule, and M. Järvisalo, editors, *Proc. of SAT Competition 2013*, volume B-2013-1 of *Department of Computer Science Series of Publications B, University of Helsinki*, pages 51–52, 2013.

[2] Armin Biere. CaDiCaL. `http://fmv.jku.at/cadical`, 2017.

[3] Aina Niemetz. *Bit-Precise Reasoning Beyond Bit-Blasting*. PhD thesis, Informatik, Johannes Kepler University Linz, 2017.

[4] Aina Niemetz, Mathias Preiner, and Armin Biere. Boolector 2.0. *JSAT*, 9:53–58, 2015.

[5] Aina Niemetz, Mathias Preiner, and Armin Biere. Boolector at the SMT competition 2016. Technical report, FMV Reports Series, Institute for Formal Models and Verification, Johannes Kepler University, Altenbergerstr. 69, 4040 Linz, Austria, 2016.

[6] Aina Niemetz, Mathias Preiner, and Armin Biere. Precise and complete propagation based local search for satisfiability modulo theories. In Swarat Chaudhuri and Azadeh Farzan, editors, *Computer Aided Verification - 28th International Conference, CAV 2016, Toronto, ON, Canada, July 17-23, 2016, Proceedings, Part I*, volume 9779 of *Lecture Notes in Computer Science*, pages 199–217. Springer, 2016.

[7] Mathias Preiner. *Lambdas, Arrays and Quantifiers*. PhD thesis, Informatik, Johannes Kepler University Linz, 2017.

[8] Mathias Preiner, Aina Niemetz, and Armin Biere. Counterexample-guided model synthesis. In *TACAS (1)*, volume 10205 of *Lecture Notes in Computer Science*, pages 264–280, 2017.

https://doi.org/10.35011/fmvtr.2017-1
Technical Report 17/1, June 2017, FMV Reports Series
Institute for Formal Models and Verification, Johannes Kepler University
Altenbergerstr. 69, 4040 Linz, Austria