SMT-COMP Entry: the Cooperating Validity Checker (CVC)

Aaron Stump
Washington University
St. Louis, Missouri, USA

Clark Barrett
New York University
New York, New York, USA

David Dill
Stanford University
Stanford, California, USA

The Cooperating Validity Checker (CVC) [4] combines the Chaff SAT solver [2] with decision procedures (DPs) for arrays [5], reals, uninterpreted functions, and datatypes (although the datatype DP has known problems and should not be used at present). The combination is in the Nelson-Oppen style [3], and the code is written in C++. Although CVC has been superceded by CVC Lite [1], whose codebase is smaller and more comprehensible, CVC remains of interest due to its apparently better performance on some large benchmarks.

The version of CVC submitted to SMT-COMP is different from the one previously publicly available (version 1.0a). The new version (version 1.0b) incorporates new ideas on mining equality proofs to reduce the size of the conflict clauses produced by the decision procedures [6]. The basic idea is to use algebraic properties of proofs built from proof rules of reflexivity, symmetry, and transitivity to excise unnecessary subproofs. Excising unnecessary subproofs may lead to the elimination of assumptions from the proof, hence reducing the size of the conflict clause. The cited work describes the techniques in full detail, and presents empirical results where as much as a 60% improvement in wall-clock time is observed for large benchmarks (which are incorporated in SMT-COMP in the QF_AUFLIA division). Version 1.0b sources and linux executable may be found in the Software section of `http://cl.cse.wustl.edu/`.

For SMT-COMP, the proof-mining version of CVC was augmented with an SMT-LIB parser. Other minor modifications include testing unsatisfiability instead of validity, and returning `unknown` for integer benchmarks that the tool, using its linear real arithmetic DP, determines are satisfiable. That is, since CVC only supports real arithmetic, it is incomplete for unsatisfiability in the integers: it may have failed to perform some integer arithmetic which shows that an apparently satisfying partial assignment is inconsistent.

References

[1] Clark Barrett and Sergey Berezin. CVC Lite: A new implementation of the cooperating validity checker. In R. Alur, editor, *Proceedings of the 16th International Conference on Computer Aided Verification*, 2004.

[2] M. Moskewicz, C. Madigan, Y. Zhaod, L. Zhang, and S. Malik. Chaff: Engineering an Efficient SAT Solver. In *39th Design Automation Conference*, 2001.

[3] G. Nelson and D. Oppen. Simplification by cooperating decision procedures. *ACM Transactions on Programming Languages and Systems*, 1(2):245–57, 1979.

[4] A. Stump, C. Barrett, and D. Dill. CVC: a Cooperating Validity Checker. In *14th International Conference on Computer-Aided Verification*, 2002.

[5] A. Stump, C. Barrett, D. Dill, and J. Levitt. A Decision Procedure for an Extensional Theory of Arrays. In *16th IEEE Symposium on Logic in Computer Science*, pages 29–37. IEEE Computer Society, 2001.

[6] A. Stump and L.-Y. Tan. The Algebra of Equality Proofs. In Jürgen Giesl, editor, *16th International Conference on Rewriting Techniques and Applications*, 2005.