SMT-COMP 2015 entry: CVC4 v1.5-prerelease

CVC4 is a tool for determining the satisfiability of a first order formula modulo a first order theory (or a combination of such theories). It is the fourth in the Cooperating Validity Checker family of tools (CVC, CVC Lite, CVC3) but does not directly incorporate code from any previous version.

CVC4 is an open and extensible SMT engine. It can be used as a stand-alone tool or as a library. It has been designed to increase the performance and reduce the memory overhead of its predecessors, especially CVC3. It is written entirely in C++ and is released under an open-source license.

More information about CVC4 is available in [1] and at its website: `http://cvc4.cs.nyu.edu/`

Divisions in main track: all
Divisions in application track: all

Expected Performance: CVC4 has been carefully engineered for performance. We are competing in all divisions as a testament to how broadly applicable CVC4 is. We don't expect to perform particularly well in difference logic (as we aren't using a specialized difference logic solver) or in logics with nonlinear arithmetic, where support is planned but currently lacking. We expect it to perform competitively in most logics.

New Functionality We are entering two versions of the solvers, stable and experimental. The stable entry is similar to last year's entry with improvements in arrays and quantifiers. Experimental version has improvments in bit-vectors, non-linear integers, and further improvements to arrays and quantifiers.

Magic #: 44444

Authors

- Kshitij Bansal (NYU)
- Clark Barrett (NYU)
- François Bobot (CEA)
- Martin Brain (Oxford)
- Christopher Conway (Google)
- Morgan Deters (NYU)
- Liana Hadarean (Oxford)
- Dejan Jovanović (SRI)
- Timothy King (Verimag)
- Tianyi Liang (U. Iowa)
- Andrew Reynolds (EPFL)
- Cesare Tinelli (U. Iowa)

CVC4 incorporates code from MiniSAT for propositional reasoning. The experimental competition submission uses CryptoMiniSat for bit-vector reasoning. CVC4 also uses ABC for cicuit simplification, and GLPK to solve hard arithmetic sub-problems.

References

[1] Clark Barrett, Christopher L. Conway, Morgan Deters, Liana Hadarean, Dejan Jovanović, Tim King, Andrew Reynolds, and Cesare Tinelli. CVC4. In Ganesh Gopalakrishnan and Shaz Qadeer, editors, *Proceedings of the* 23^{rd} *International Conference on Computer Aided Verification (CAV '11)*, volume 6806 of *Lecture Notes in Computer Science*, pages 171–177. Springer, July 2011. Snowbird, Utah.