zbMATH Open
THE FIRST RESOURCE FOR MATHEMATICS

Marić, Filip

Formal verification of a modern SAT solver by shallow embedding into Isabelle/HOL. (English) Zbl 1208.68205

Theor. Comput. Sci. 411, No. 50, 4333-4356 (2010).

Summary: We present a formalization and a formal total correctness proof of a MiniSAT-like SAT solver within the system Isabelle/HOL. The solver is based on the DPLL procedure and employs most state-of-the-art SAT solving techniques, including the conflict-guided backjumping, clause learning, and the two-watched unit propagation scheme. A shallow embedding into Isabelle/HOL is used and the solver is expressed as a set of recursive HOL functions. Based on this specification, the Isabelle's built-in code generator can be used to generate executable code in several supported functional languages (Haskell, SML, and OCaml). The SAT solver implemented in this way is, to our knowledge, the first fully formally and mechanically verified modern SAT solver.

MSC:

68T20 Problem solving in the context of artificial intelligence (heuristics, search strategies, etc.)
68Q60 Specification and verification (program logics, model checking, etc.)

Cited in **11** Documents

Keywords:

formal program verification; SAT problem; DPLL procedure; Isabelle

Software:

Archive Formal Proofs; BerkMin; Chaff; Isabelle; Isabelle/HOL; MiniSat; OCaml; SATO; SAT Solver Verification

Full Text: DOI

References:

[1] Biere, A.; Heule, M.; van Maaren, H.; Walsh, T., Handbook of satisfiability, (2009), IOS Press Amsterdam · Zbl 1183.68568

[2] L. Bulwahn, A. Krauss, F. Haftmann, L. Erkok, J. Matthews, Imperative functional programming with Isabelle/HOL, in: TPHOLs '08, Montreal, 2008. · Zbl 1165.68352

[3] S.A. Cook, The complexity of theorem-proving procedures, in: 3rd STOC, New York, 1971, pp. 151-158.

[4] Davis, M.; Logemann, G.; Loveland, D., A machine program for theorem-proving, Communications of the ACM, 5, 7, 394-397, (1962) · Zbl 0217.54002

[5] Davis, M.; Putnam, H., A computing procedure for quantification theory, Journal of the ACM, 7, 3, 201-215, (1960) · Zbl 0212.34203

[6] N. Een, N. Sorensson, An extensible SAT solver, in: SAT'03, in: LNCS, vol. 2919, S. Margherita Ligure, 2003, pp. 502-518. · Zbl 1204.68191

[7] A. van Gelder, Verifying propositional unsatisfiability: pitfalls to avoid, in: SAT '07, in: LNCS, vol. 4501, Lisbon, 2007, pp. 328-333.

[8] Gomes, C.P.; Kautz, H.; Sabharwal, A.; Selman, B., Satisfiability solvers, ()

[9] E. Goldberg, Y. Novikov, Berkmin: a rast and robust SAT solver, in: DATE'02, Paris, 2002, pp. 142-149.

[10] F. Haftmann, Code generation from Isabelle/HOL theories. http://isabelle.in.tum.de/documentation.html, 2008.

[11] S. Krstić, A. Goel, Architecting solvers for SAT modulo theories: nelson-oppen with DPLL, in: FroCos'07, in: LNCS, vol. 4720, Liverpool, 2007, pp. 1-27. · Zbl 1148.68466

[12] A. Krauss, Defining recursive functions in Isabelle/HOL. http://isabelle.in.tum.de/documentation.html, 2008.

[13] S. Lescuyer, S. Conchon, A reflexive formalization of a SAT solver in coq, in: TPHOLs'08: Emerging Trends, Montreal, 2008.

[14] F. Marić, Formal verification of modern SAT solvers, The Archive of Formal Proofs, 2008. http://afp.sf.net/entries/SATSolverVerification.shtml.

[15] Marić, F., Formalization and implementation of SAT solvers, Journal of automated reasoning, 43, 1, 81-119, (2009) · Zbl 1187.68557

© 2021 FIZ Karlsruhe GmbH

[16] Marić, F.; Janičić, P., Formal correctness proof for DPLL procedure, Informatica, 21, 1, 57-78, (2010) · Zbl 1209.68514

[17] F. Marić, P. Janičić, SAT verification project, in: TPHOLs'09: Emerging Trends, Munich, 2009.

[18] M. Moskewicz, C. Madigan, Y. Zhao, L. Zhang, S. Malik, Chaff: engineering an efficient SAT solver, in: DAC'01, Las Vegas, 2001, pp. 530-535.

[19] Marques-Silva, J˜P.; Sakallah, K˜A., Grasp: A search algorithm for propositional satisfiability, IEEE transactions on computers, 48, 5, 506-521, (1999) · Zbl 1392.68388

[20] Nieuwenhuis, R.; Oliveras, A.; Tinelli, C., Solving SAT and SAT modulo theories: from an abstract davis – putnam – logemann – loveland procedure to DPLL(T), Journal of the ACM, 53, 6, 937-977, (2006) · Zbl 1326.68164

[21] Nipkow, T.; Paulson, L.˜C.; Wenzel, M., ()

[22] Shankar, N., Towards mechanical metamathematics, Journal of automated reasoning, 1, 4, 407-434, (1985) · Zbl 0616.68075

[23] N. Shankar, M. Vaucher, The mechanical verification of a DPLL-based satisfiability solver, Personal correspondence. · Zbl 1347.68307

[24] H. Zhang, SATO: An efficient propositional prover, in: CADE-14, in: LNCS, vol. 1249, Townsville, 1997, pp. 272-275.

[25] Verified software: theories, tools, experiments, Conference. http://vstte.ethz.ch/.

[26] L. Zhang, S. Malik, validating SAT solvers using independent resolution-based checker, in: DATE '03, Washington DC, 2003, p. 10880.

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.

© 2021 FIZ Karlsruhe GmbH