DCCASatToRiss in SAT Race 2015

Chuan Luo[*†], Shaowei Cai[†], Wei Wu[‡] and Kaile Su[§¶]

[*]School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China
[†]Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, Beijing 100190, China
[‡]Center for Quantum Computation and Intelligent Systems, University of Technology, Sydney, Sydney 2007, Australia
[§]Department of Computer Science, Jinan University, Guangzhou 510632, China
[¶]Institute for Integrated and Intelligent Systems, Griffith University, Brisbane 4111, Australia
{chuanluosaber, shaoweicai.cs, william.third.wu}@gmail.com; k.su@griffith.edu.au

Abstract—This document describes the hybrid SAT solver "DCCASatToRiss", which is combination of a local search solver and a complete solver.

I. Introduction

Recently, we proposed a local search strategy, named configuration checking (CC), which contributes to the diversification. The CC strategy is first used in the context of minimum vertex cover solving [1], [2], [3], [4]. As CC is a general idea, it has been successfully applied in the Boolean satisfiability (SAT) problem [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] and the maximum satisfiability (MAX-SAT) problem [15], [16].

In the context of SAT, there are two different CC strategies, i.e., neighboring variables based configuration checking (NVCC) [10] and clause states based configuration checking (CSCC) [14]. On the basis of the *CCASat* solver [10] which is the winner of random SAT track in SAT Challenge 2012, by combining NVCC and CSCC, we developed a local search solver called *DCCASat* [11]. As the performance of local search solvers is complementary to the performance of complete solvers on solving hard-combinatorial SAT instances, we combine *DCCASat* with a complete solver. Inspired by the success of the *SparrowToRiss* solver [17], [18], which combines a local search solver *Sparrow* [19] with a complete solver *Riss3g* [20] and a preprocessor *CP3* [21], by replacing *Sparrow* with *DCCASat*, we obtain the resulting hybrid solver named *DCCASatToRiss*. We would like to note that the *DCCASatToRiss* solver has been presented in the literature [11].

II. Main Techniques

The *DCCASatToRiss* solver is a combination of the preprocessor *CP3*, the local search solver *DCCASat* and the complete solver *Riss3g*.

The main procedures of *DCCASatToRiss* can be described as follows. For solving an SAT instance, the *DCCASatToRiss* solver first employs *CP3* to simplify the instance, and then it calls the local search solver *DCCASat* to solve the resulting simplified instance with t search steps. If the instance is solved by *DCCASat* with t search steps, then *DCCASatToRiss* reports the solution which is found by *DCCASat*; otherwise, *DCCASatToRiss* utilizes the complete solver *Riss3g* as well as some information obtained by *DCCASat* to solve the resulting simplified instance, and reports the solution if the instance is solved by *Riss3g*.

III. Main Parameters

The parameter t is set to 500,000,000 in SAT Race 2015. The parameters used in the current version of *DCCASat* in *DCCASatToRiss* are the same as the ones reported in the literature [11], inspired by the literature [22]. The parameters used in the current version of *CP3* and *Riss3g* in *DCCASatToRiss* are the same as the ones in the literature [17].

IV. Implementation Details

The *DCCASatToRiss* solver is implemented in programming language C++.

Acknowledgment

References

[1] S. Cai, K. Su, and A. Sattar, "Local search with edge weighting and configuration checking heuristics for minimum vertex cover," *Artificial Intelligence*, vol. 175, no. 9-10, pp. 1672–1696, 2011.

[2] ——, "Two new local search strategies for minimum vertex cover," in *Proc. of AAAI 2012*, 2012, pp. 441–447.

[3] S. Cai, K. Su, C. Luo, and A. Sattar, "Numvc: An efficient local search algorithm for minimum vertex cover," *Journal of Artificial Intelligence Research*, vol. 46, pp. 687–716, 2013.

[4] S. Cai, J. Lin, and K. Su, "Two weighting local search for minimum vertex cover," in *Proc. of AAAI 2015*, 2015, pp. 1107–1113.

[5] S. Cai and K. Su, "Local search with configuration checking for SAT," in *Proc. of ICTAI 2011*, 2011, pp. 59–66.

[6] C. Luo, K. Su, and S. Cai, "Improving local search for random 3-SAT using quantitative configuration checking," in *Proc. of ECAI 2012*, 2012, pp. 570–575.

[7] S. Cai and K. Su, "Configuration checking with aspiration in local search for SAT," in *Proc. of AAAI 2012*, 2012, pp. 434–440.

[8] ——, "Comprehensive score: Towards efficient local search for SAT with long clauses," in *Proc. of IJCAI 2013*, 2013, pp. 489–495.

[9] C. Luo, S. Cai, W. Wu, and K. Su, "Focused random walk with configuration checking and break minimum for satisfiability," in *Proc. of CP 2013*, 2013, pp. 481–496.

[10] S. Cai and K. Su, "Local search for Boolean satisfiability with configuration checking and subscore," *Artificial Intelligence*, vol. 204, pp. 75–98, 2013.

[11] C. Luo, S. Cai, W. Wu, and K. Su, "Double configuration checking in stochastic local search for satisfiability," in *Proc. of AAAI 2014*, 2014, pp. 2703–2709.

[12] S. Cai, C. Luo, and K. Su, "Scoring functions based on second level score for k-SAT with long clauses," *Journal of Artificial Intelligence Research*, vol. 51, pp. 413–441, 2014.

[13] C. Luo, K. Su, and S. Cai, "More efficient two-mode stochastic local search for random 3-satisfiability," *Applied Intelligence*, vol. 41, no. 3, pp. 665–680, 2014.

[14] C. Luo, S. Cai, K. Su, and W. Wu, "Clause states based configuration checking in local search for satisfiability," *IEEE Transactions on Cybernetics*, vol. 45, no. 5, pp. 1014–1027, 2015.

[15] C. Luo, S. Cai, W. Wu, Z. Jie, and K. Su, "CCLS: an efficient local search algorithm for weighted maximum satisfiability," *IEEE Transactions on Computers*, vol. 64, no. 7, pp. 1830–1843, 2015.

[16] S. Cai, Z. Jie, and K. Su, "An effective variable selection heuristic in SLS for weighted Max-2-SAT," *Journal of Heuristics*, vol. 21, no. 3, pp. 433–456, 2015.

[17] A. Balint and N. Manthey, "Sparrow+CP3 and SparrowToRiss," in *Proc. of SAT Competition 2013: Solver and Benchmark Descriptions*, 2013, pp. 87–88.

[18] ——, "SparrowToRiss," in *Proc. of SAT Competition 2014: Solver and Benchmark Descriptions*, 2014, pp. 77–78.

[19] A. Balint and A. Fröhlich, "Improving stochastic local search for SAT with a new probability distribution," in *Proc. of SAT 2010*, 2010, pp. 10–15.

[20] N. Manthey, "The SAT solver Riss3g at SC 2013," in *Proc. of SAT Competition 2013: Solver and Benchmark Descriptions*, 2013, pp. 72–73.

[21] ——, "Coprocessor 2.0 - A flexible CNF simplifier," in *Proc. of SAT 2012*, 2012, pp. 436–441.

[22] S. Cai, C. Luo, and K. Su, "CCAnr: A configuration checking based local search solver for non-random satisfiability," in *Proc. of SAT 2015*, 2015, p. To appear.