

Comparing SAT Encodings for Model Checking

Alan Frisch, Daniel Sheridan, and Toby Walsh

Artificial Intelligence Group,
Department of Computer Science,
University of York,
York, England
{frisch, djs, tw}@cs.york.ac.uk

Bounded model checking (BMC) [1] was proposed as a solution to some of the problems (*e.g.* space explosion) of conventional BDD-based symbolic model checking by introducing a temporal bound. The problem can then be encoded as a Boolean formula and used as input to a SAT checker: the output of a BMC tool is a conjunction of state transition functions and state verification functions.

Our work has focused on improving encodings for BMC by using a normal form for temporal logic [2]. With a variety of encodings available, it has become necessary to perform a comparison between them so that the best encoding can be chosen. By making available a method of comparison that is not based on benchmarks, we can develop systems which are able to choose the best encoding for a given input. In addition, a comparison that is not based on benchmarks is less time consuming to perform and can yield more detailed results; for example, the pathological cases for each encoding can easily be found.

Initial work in this area has been based on the method in [3] for predicting the number of clauses produced by the encoding. This analysis has been extended to allow for other clause form conversions, and we have developed new metrics allowing for the consideration of, for example, clause size. We use a variety of different benchmarks in comparing the time taken by a SAT checker running on clauses sets generated by BMC with the predictions based on the number of clauses, size of clauses, and other metrics. We show how these comparisons indicate the strengths of the various encodings, and suggest ways in which the choice of encoding might be made based on these predictions.

References

1. Armin Biere, Alessandro Cimatti, Edmund Clarke, and Yunshan Zhu. Symbolic model checking without BDDs. In W.R. Cleaveland, editor, *Tools and Algorithms for the Construction and Analysis of Systems. 5th International Conference, TACAS'99*, volume 1579 of *Lecture Notes in Computer Science*, pages 193–207. Springer-Verlag Inc., July 1999.
2. Alexander Bolotov and Michael Fisher. A resolution method for CTL branching-time temporal logic. In *Proceedings of the Fourth International Workshop on Temporal Representation and Reasoning (TIME)*. IEEE Press, 1997.
3. Andreas Nonnengart, Georg Rock, and Christoph Weidenbach. On generating small clause normal forms. In Claude Kirchner and Hélène Kirchner, editors, *Fifteenth International Conference on Automated Deduction*, volume 1421 of *Lecture Notes in Artificial Intelligence*, pages 397–411. Springer-Verlag, 1998.