

Encoding Reachability with Quantification

Michael Cashmore¹, Maria Fox¹, and Enrico Giunchiglia²

¹ King's College London, London, WC2R 2LS, UK `firstname.lastname@kcl.ac.uk`

² Universit di Genova, 16145 Genova (GE), Italy `lastname@unige.it`

Abstract. We report on techniques for translating bounded propositional reachability problems into Quantified Boolean Formulae (QBF). The techniques exploit the binary-tree structure of the QBF quantification to produce encodings logarithmic in the size of the instance and thus exponentially smaller than the corresponding SAT encoding with the same bound. We briefly describe our experimental results, and discuss the challenging structure that is implicit in these encodings.

Planning as Satisfiability is one of the most well-known and effective techniques for classical planning: SATPLAN [7] was an award-winning system in the deterministic track for optimal planners in the first International Planning Competition (IPC) in 1998, the 4th IPC in 2004, and the 5th IPC in 2006.

The basic idea is to encode the existence of a plan with $n + 1$ (or fewer) steps as a propositional (SAT) formula obtained by unfolding, n times, the symbolic transition relation of the automaton described by the planning problem. Formulations in QBF [9, 6, 3], present encodings logarithmic in n , resembling the proof of the PSPACE-hardness of solving QBFs [10, 11]. We report on an extension of these encodings called the *Compact Tree Encoding* (CTE) [3, 4]. The CTE embodies a particular quantification structure that is very difficult to solve using QDPLL-based solvers.

We review experimental results [3, 4] gathered with solvers DEPQBF [8], QUBE7 [5], and the resolution-based solver QUANTOR-3.0 [1]. QUBE7, although it is not the latest version, was used as we have a modified version that allows for easier plan extraction.

In brief, we found that the QBF encodings used less memory on average than SAT, but scaled very poorly in terms of time to solve. Interestingly we found that QUANTOR-3.0 performed far better than either QDPLL solver, which we propose is due to the placement of the initial and goal constraints in the innermost existentially quantified layer of the problem.

The outermost existentially quantified variables represent the mid-point of the plan. Also, the initial and goal constraints are upon only the innermost existentially quantified variables. We investigate the way in which a top-down approach interacts with the encoding.

The Planning Problem

Let $\mathcal{F}$ and $\mathcal{A}$ be two finite sets representing the sets of *fluents* and *actions* respectively. The fluents describe the state of the world at a single time-step, the actions describe

the ways in which the world can be changed. In the following, we use X to denote the whole set of variables, i.e., the set of fluents unioned with the set of actions.

A *planning problem* is a triple $\langle I, \tau, G \rangle$ where

- I is a Boolean formula over $\mathcal{F}$ and represents the set of *initial states*;
- τ is a Boolean formula over $X \cup X'$ representing the *transition relation*, describing how all actions affect states.
- G is a Boolean formula over $\mathcal{F}$ and represents the set of *goal states*.

The only assumption that we make is that the description is deterministic: there is only one state satisfying I and the execution of a (complex) action α in a state s can lead to at most one state s' .

Consider a planning problem $\Pi = \langle I, \tau, G \rangle$. As standard in planning as satisfiability, the existence of a parallel plan with makespan n is proved by building a propositional formula with n copies of the set of variables. For $n \geq 1$, the *planning problem Π with makespan n* is the Boolean formula Π_n defined as

$$I(X_1) \wedge \bigwedge_{i=1}^n \tau(X_i, X_{i+1}) \wedge G(X_{n+1}) \quad (n \geq 0) \quad (1)$$

and a *plan for Π_n* is an assignment satisfying (1).

Compact Tree Encoding

In the CTE there is a one-to-one correspondence between the states traversed and the *nodes* of the tree corresponding to the QBF. The *Compact Tree Encoding* of a planning problem $\langle I, \tau, G \rangle$ with makespan $n = 2^{k+1}$ is the formula:

$$I(X_I) \wedge Q_k(X_I, X_G) \wedge G(X_G) \quad (2)$$

where

$$\begin{aligned} Q_k(X_I, X_G) := & \exists X_k \forall y_k \dots \exists X_1 \forall y_1 \exists X (\\ & ((\bigwedge_{i=1}^k \neg y_i) \Rightarrow \tau(X_I, X)) \\ & \wedge ((\bigwedge_{i=1}^k y_i) \Rightarrow \tau(X, X_G)) \\ & \bigwedge_{i=1}^k (((\neg y_i \bigwedge_{j=1}^{i-1} y_j) \Rightarrow \tau(X, X_i)) \\ & \quad \wedge ((y_i \bigwedge_{j=1}^{i-1} \neg y_j) \Rightarrow \tau(X_i, X)))) \end{aligned}$$

This formula states that the goal state X_G is reachable in 2^{k+1} applications of the transition relation. Only $k + 1$ states are quantified (X_k to X_1 and X).

The CTE (2) has k universal variables and $(k+1)|X|$ existential variables, checking the existence of plans with makespan 2^{k+1} . The conversion of (2) to prenex conjunctive normal form has $2(k+1)|\tau|$ clauses.

Proposition 1. *For each $k \geq 0$, if $n = 2^{k+1}$ the existential closures of (1) and the Compact Tree Encoding (2) are equivalent.*

The proof is presented in Cashmore et al. [4].

References

- [1] Biere, A.: Resolve and expand. In: Proceedings of the 7th International Conference on Theory and Applications of Satisfiability Testing (SAT04). pp. 238–246 (2004)
- [2] Bylander, T.: The computational complexity of propositional STRIPS planning. *Artificial Intelligence* 69, 165–204 (1994)
- [3] Cashmore, M., Fox, M.: Planning as QBF. *International Conference on Automated Planning and Scheduling Doctoral Consortium (ICAPS 2010)* (2010)
- [4] Cashmore, M., Fox, M., Giunchiglia, E.: Planning as quantified boolean formulae. In: Proceedings of the 20th European Conference on Artificial Intelligence (ECAI 2012) (2012)
- [5] Giunchiglia, E., Narizzano, M., Tacchella, A.: QUBE: A system for deciding quantified Boolean formulas satisfiability. In: Proceedings of the 1st International Joint Conference on Automated Reasoning (IJCAR'01). pp. 364–369 (2001)
- [6] Jussila, T., Biere, A.: Compressing BMC encodings with QBF. *Electronic Notes in Theoretical Computer Science* 174(3), 45–56 (2007)
- [7] Kautz, H., Selman, B.: Planning as satisfiability. In: Proceedings of the 10th European Conference on Artificial Intelligence (ECAI'92). pp. 359–363 (1992)
- [8] Lonsing, F., Biere, A.: DepQBF: A dependency-aware QBF solver system description. *Journal on Satisfiability, Boolean Modeling and Computation*. 7, 7176 (2010)
- [9] Rintanen, J.: Partial implicit unfolding in the Davis-Putnam procedure for quantified Boolean formulae. In: Proceedings of the 8th International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR'01). pp. 362–376 (2001)
- [10] Savitch, W.J.: Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences* 4(2), 177–192 (1970)
- [11] Stockmeyer, L.J., Meyer, A.R.: Word problems requiring exponential time: Preliminary report. In: Proceedings of the 5th Annual ACM Symposium on Theory of Computing (STOC'73). pp. 1–9 (1973)