Modal Clause Transformation: A Bottom-Up Approach to Modal Theorem Proving (Extended Abstract)

Jun-ichi Akahani
NTT Communication Science Laboratories

Katsumi Inoue
Dept. of Information and Computer Sciences
Toyohashi University of Technology

Ryuzo Hasegawa
ICOT

Modal logics have been gaining popularity in various domains of Computer Science. For example, in the logics of programs and in the analyses of distributed systems. For such applications, modal logics require fast and efficient theorem provers.

Current proof methods for modal logics can be classified into two categories: the *direct* approach and the *translation* approach. In the direct approach, existing proof methods for classical logic are extended for use in modal logics. Typical examples are modal tableau methods [Fit88] and modal resolution methods [Far86]. In the translation approach, on the other hand, modal formulae are translated into classical formulae [Ohl88]. The translation approach has been proven to be useful because it can be applied to various modal systems. Many proof methods have recently been proposed for this approach [Non93], which focus only on implementing *accessibility relations* to be applied to various modal systems. The translation approach has another merit in that it can employ many control strategies developed for theorem proving in classical logic. Unfortunately, the previous proposed methods do not address the issue of controlling inference to reduce the search space.

In this paper, to take advantage of the above merit, we propose a version of the translation method, called the *modal clause transformation method*. This method translates propositional modal formulae into input clauses of the model generation theorem prover MGTP [FH91]. The MGTP is a parallel theorem prover that generates models of input clauses in a bottom-up manner. The basic idea of the modal clause transformation method is partial evaluation of the rewriting rules for the modal tableau method. To avoid the generation of branches irrelevant to the proof, we adapt the Non-Horn Magic Set (NHM) [HOI93] method that transforms input clauses so as to simulate top-down reasoning. We analyze input modal formulae to incorporate control information specific to a given input modal formula into its translated formula.

Since the modal tableau methods may enumerate accessibility relations, they suffer from infinite loop for the transitive logics such as $K4$. To overcome this problem, we adapt the NHM method so as to calculate accessibility relations only when they are required. The basic idea is similar to the functional simulation method [Non93]. Their method, however, requires full unification mechanism for calculating accessibility relations while our method needs just matching.

In the modal clause transformation method, every transformed clause is *range-restricted* [MB88], that is, every variable in the clause appears in the antecedent. Since matching is sufficient instead of full unification for range-restricted clauses, efficient theorem proving is possible for transformed clauses.

References

[Far86] L. Farinas del Cerro. Molog, a system that extends Prolog with modal logic. *New Generation Computing*, 4:35–50, 1986.

[FH91] H. Fujita and R. Hasegawa. A model generation theorem prover in kl1 using ramified-stack algorithm. In *Proc. of ICLP'91*, pages 535–548, 1991.

[Fit88] M. Fitting. First-order modal tableaux. *J. of Automated Reasoning*, 4:191–213, 1988.

[HOI93] R. Hasegawa, Y. Ohta, and K. Inoue. Non-horn magic sets and their relation to relevancy testing. Technical Report 834, ICOT, 1993.

[MB88] R. Manthey and F. Bry. SATCHMO: A theorem prover implemented in Prolog. In *Proc. of the 9th Int. Conf. on Automated Deduction*, pages 415–434, 1988.

[Non93] A. Nonnengart. First-order modal logic theorem proving and functional simulation. In *Proc. of IJCAI'93*, pages 80—85, 1993.

[Ohl88] H. J. Ohlbach. A resolution calculus for modal logics. In *Proc. of the 9th Int. Conf. on Automated Deduction*, pages 500–516, 1988.