TABLEAUX: A General Theorem Prover for Modal Logics

LAURENT CATACH
LITP – Paris 6 University, IBM Paris Scientific Center, France

(Received: 27 June 1988; accepted: 8 April 1991)

Abstract. We present a general theorem proving system for propositional modal logics, called TABLEAUX. The main feature of the system is its generality, since it provides an unified environment for various kinds of modal operators and for a wide class of modal logics, including usual temporal, epistemic or dynamic logics. We survey the modal languages covered by TABLEAUX, which range from the basic one $L(D, O)$ through a complex multimodal language including several families of operators with their transitive-closure and converse. The decision procedure we use is basically a semantic tableaux method, but with slight modifications compared to the traditional one. We emphasize the advantages of such semantical proof methods for modal logics, since we believe that the models construction they provide represents perhaps the most attractive feature of these logics for possible applications in computer science and AI. The system has been implemented in Prolog, and appears to be of reasonable efficiency for most current examples. Experimental results are given in the paper, with two lists of test examples.

Key words. Modal (temporal, dynamic, epistemic) logics, theorem proving, decision procedures, tableaux method, experimental results.

1. Introduction

TABLEAUX is an automated theorem proving system for a wide class of modal logics, based on an adapted version of the traditional semantic tableaux method for these logics [20, 21, 30, 49]. This method also works for many extensions of traditional modal logic, such as temporal logics [3, 4, 15, 58], dynamic logics [29, 43], or epistemic logics [27, 31], which have become of some importance in computer science and AI.

In fact, the tableaux method is frequently used within the theoretical presentation of a modal system; this is due to some of its inherent advantages:

- it is a proof method closely related to the semantics of the modal operators,
- it provides a completeness theorem and a decision procedure, and may also be used to prove complexity results about the logic (e.g. the Fischer–Ladner filtrations method for dynamic logic [22]),
- the technique has a general scheme, so it can be adapted from one logical system to another,
- it works for proving both the validity and the satisfiability of formulas,
- it is constructive, i.e. it explicitly builds a model (resp. a counter-model) when a formula is found to be satisfiable (resp. non-valid).
The basic tableaux method is simple, but faces important problems of complexity; this is certainly the reason why few real implementations have been built [25, 42, 44, 51, 53]. Other methods have been explored to achieve automated deduction for these logics (see Section 5), but complexity is always crucial, especially for modal languages including transitive-closure operators.

Our motivations for developing TABLEAUX and the purposes of such a system are the following:

- To build a general theorem prover for many modal logics, and to use it as a tool for studying these logics. Of course, such a system is interesting for the logician, but we believe that it can also be very helpful for new developments in computer science and especially for experimenting with ideas, exploring possible applications and creating prototypes.
- To be able, in that theorem prover, to choose a modal system in the most declarative way, i.e. by defining the language of modal operators and by giving semantic or axiomatic characterizations of the logic.
- Most automated deduction methods recently proposed for modal logics are syntactic, in the sense that they do not rely on the explicit construction of models. On the contrary, we believe that it is precisely this feature of modal logics which is perhaps the most attractive, if modal logics are to be of any practical interest in computer science and AI. Thus, one of our motivations was to use a semantic deduction method.
- As shown by TABLEAUX, starting with a basic modal language and the tableaux method, one can obtain many of the modal systems developed in the literature, from the aspect of both the expressiveness (operators of temporal, epistemic, dynamic, etc. logics) and of the behavior (simulation [40], model checking [10], epistemic puzzles, etc.); thus, an important feature of TABLEAUX is its provision of a unified version of various existing modal systems and applications.
- To show that it is possible to implement the tableaux method with good efficiency for current examples. Inasmuch as complexity is indeed a handicap in the general case, it also depends very much on the systems and on the formulas considered.
- Starting from the 'classical' tableaux method, to discover where problems arise in an implementation, and to see whether new methods could be defined (with better efficiency). In fact, we believe that such prototypes, written in a powerful language such as Prolog, constitute necessary preliminaries before the realization of more efficient theorem provers, based for example on low-level languages.
- One main motivation for developing TABLEAUX was also to study expressive features of modal languages, and especially those of general multimodal systems, as studied in [7, 8]. This topic is briefly presented in Section 2.7.

The modal operators and systems covered by TABLEAUX are described in Section 2. This section does not contain new results about expressiveness or axiomatisations (though it might suggest interesting extensions, such as the multimodal languages we consider in Section 2.7), but it is necessary for us to give a complete and