A Tableau Calculus for a Temporal Logic with Temporal Connectives

Wolfgang May

Institut für Informatik, Universität Freiburg, Germany,
may@informatik.uni-freiburg.de

Abstract. The paper presents a tableau calculus for a linear time temporal logic for reasoning about processes and events in concurrent systems. The logic is based on temporal connectives in the style of Transaction Logic [BK94] and explicit quantification over states. The language extends first-order logic with sequential and parallel conjunction, parallel disjunction, and temporal implication. Explicit quantification over states via state variables allows to express temporal properties which cannot be formulated in modal logics.
Using the tableau representation of temporal Kripke structures presented for CTL in [MS96] which represents states by prefix terms, explicit quantification over states is integrated into the tableau calculus by an adaptation of the δ-rule from first-order tableau calculi to the linear ordering of the universe of states.
Complementing the CTL calculus, the paper shows that this tableau representation is both suitable for *modal temporal logics* and for logics using *temporal connectives*.

1 Introduction

When extending first-order logic to temporal logic, most approaches are based on *modal operators*, such as LTL/CTL or Dynamic Logic. Here, formulas are modified via modalities – inducing an implicit quantification over states. Formulas are evaluated wrt. states or (infinite) paths, thus they do not support an intuitive notion of sequentiality or parallelism.

For reasoning about processes and events in concurrent systems, *temporal connectives* such as sequential, parallel, and alternative composition or iteration are well-known from process algebraic formalisms. First-order-logic based formalisms using temporal connectives (which implies evaluating formulas wrt. finite path segments) are rare, although they have obvious advantages when reasoning about temporal behavior of processes. For Transaction Logic [BK94], it has been shown how to write executable specifications in such a formalism.

There are some temporal constraints which cannot be expressed in temporal modal logics, e.g., that "if some state is reached such that a given predicate p has the same extension as now, then q holds in this state" (cf. [TN96,CT98]). In [CT98], it is shown that this can be expressed in 2-FOL which is a two-sorted first order language for dealing with a linear temporal state space by

$$\forall s_1, s_2 : (\forall x : p(x, s_1) \leftrightarrow p(x, s_2) \wedge s_1 < s_2) \rightarrow q(s_2) \ .$$

Neil V. Murray (Ed.): TABLEAUX'99, LNAI 1617, pp. 232–246, 1999.

This example motivates that an explicit quantification and addressing of states via state variables would be useful in a temporal logic.

In [MS96], a tableau semantics for first-order Kripke structures has been presented, together with a tableau calculus for first-order CTL. There, states have been described by *prefix terms* which provide a natural way to adapt the γ and δ-rule to quantification by state variables.

In the present paper, it is shown how the same approach applies for this temporal logic based on *temporal connectives* and explicit quantification over states.

The paper is structured as follows: After introducing some basic notions in Section 2, a (linear-time) logic for formulating complex events and dynamic constraints is presented in Section 3. Section 4 contains the tableau semantics for linear Kripke structures, and the tableau calculus is given in Section 5. Section 6 closes with some concluding remarks.

Related Work. Most of the work in Temporal Logics focuses on modal logics, e.g., CTL, modal μ-calculus, or Dynamic Logic. An overview of tableau calculi for (modal) temporal logics have been summarized in [Wol85], a recent one is described in [MP95]. *Interval Logics* contain operators for sequential composition and iteration similar to those known from programming languages [Mos86]. A tableau method for interval logic has, e.g., been presented in [BT98]. Other formalisms for expressing temporal constraints in non-modal logics are dealt with in [Sin95], [BK94], [Pra90], [Jab94], and [TN96,CT98].

2 Basic Notions

Let Σ be a *signature* consisting of a set Σ_{func} of function symbols a set Σ_{pred} of predicate symbols with fixed arities $\mathsf{ord}(f)$ resp. $\mathsf{ord}(p)$, and $\mathsf{Var} := \{x_1, x_2, \ldots\}$ an infinite set of variables. Let Term_Σ denote the set of terms over Σ and Var. The notions of bound and free variables are defined as usual, $\mathsf{free}(\mathcal{F})$ denoting the set of variables occurring free in a set $\mathcal{F}$ of formulas.

A *substitution* (over a signature Σ) is a mapping $\sigma : \mathsf{Var} \to \mathsf{Term}_\Sigma$ where $\sigma(x) \neq x$ for only finitely many $x \in \mathsf{Var}$, here denoted by $[\sigma(x)/x]$. Substitutions are extended to terms and formulas as usual.

A first-order structure $\boldsymbol{I} = (I, \boldsymbol{U})$ over a signature Σ consists of a universe $\boldsymbol{U}$ and a *first-order interpretation* I of Σ which maps every function symbol $f \in \Sigma$ to a function $I(f) : \boldsymbol{U}^{\mathsf{ord}(f)} \to \boldsymbol{U}$ and every predicate symbol $p \in \Sigma$ to a relation $I(p) \subseteq \boldsymbol{U}^{\mathsf{ord}(p)}$.

A *variable assignment* is a mapping $\chi : \mathsf{Var} \to \boldsymbol{U}$. For a variable assignment χ, a variable x, and $d \in \boldsymbol{U}$, the *modified* variable assignment χ_x^d is identical with χ except that it assigns d to x. Let Ξ denote the set of variable assignments.

Every interpretation induces an *evaluation* $\boldsymbol{I} : \mathsf{Term}_\Sigma \times \Xi \to \boldsymbol{U}$ s.t. $\boldsymbol{I}(x, \chi) := \chi(x)$ for $x \in \mathsf{Var}$, and $\boldsymbol{I}(f(t_1, ..., t_n), \chi) := (I(f))(\boldsymbol{I}(t_1, \chi), ..., \boldsymbol{I}(t_n, \chi))$ for $f \in \Sigma$, $\mathsf{ord}(f) = n$ and $t_1, ..., t_n \in \mathsf{Term}_\Sigma$. The truth of a formula F in a first-order structure $\boldsymbol{I}$ under a variable assignment χ, $(\boldsymbol{I}, \chi) \models F$ is defined as usual.