A Tableau for Temporal Logic over the Reals

Mark Reynolds [1]

School of Computer Science and Software Engineering
University of Western Australia
Crawley, Perth Western Australia 6009

Abstract

We provide a simple, sound, complete and terminating tableau decision procedure for the temporal logic of until and since over the real numbers model of time. This logic is an important basis for reasoning about concurrency, metric constraints and planning. Despite its usefulness and long history, there are no existing implementable reasoning techniques for it.
The tableau uses a mosaic-based technique to translate the satisfiability problem into a question about the way that intervals of a real-flowed model relate to each other. It builds on top of recently developed reasoning tools for general linear time by applying some interesting but computationally simple checks.

Keywords: Temporal Logic, Reals, Tableau.

1 Introduction

Although discrete time temporal logics are the most common, there has been a separate thread of steady development of continuous time alternatives since the earliest beginnings. Being able to reason about events and processes unfolding continuously has an enormous range of applications from concurrency and refinement in reactive systems, as a basis for the metric temporal logics used for model checking automated systems, to artificial planning, natural language semantics and philosophical arguments.

In this paper we investigate the most natural and useful such temporal logic: RTL, the propositional temporal logic over real-numbers time using the Until and Since connectives introduced in [6]. RTL is as expressive as first-order logics over linear structures [6]. It is decidable [2,10,8] (in PSPACE) and has complete axioms systems [5,9].

Currently there is no satisfiability or validity checking procedure for RTL that looks remotely amenable to implementation. In this paper we build on the results and techniques of [10] and present what seems to be an intuitive tableau style decision procedure for RTL which will not be hard to implement (albeit only to work with sufficiently small formulas).

[1] mark.reynolds@uwa.edu.au partially supported by ARC

The proof of correctness here uses the *mosaics* which were used to prove PSPACE decidability of RTL in [10]. Mosaics are small pieces of a model. We can decide whether a finite set of mosaics is sufficient to be used to build a real-numbers model of a given formula by considering something like a game tree which can also be viewed as a tableau. Such an idea was suggested for general dense time reasoning in [11] and those ideas have led to recent tableaux [12] and more streamlined implementations [1].

Narrowing our focus to the reals, we have to look carefully at shapes of sub-graphs within the tree to enforce the peculiar properties of the reals: such as density, Dedekind completeness and separability.

The contribution here is presenting a sound and complete mosaic-based tableau system to decide satisfiability in RTL. We aim mainly to show clearly how mosaics can be the building blocks of a tableau with this logic here: the system is not at all streamlined and is intended to provide the foundation of more intelligent tableau construction techniques in future work.

Below we define the logic RTL in section 2, explain mosaics in section 3, show how to make a sufficient set of mosaics in section 4, lay out the basic mosaic tableau in section 5, adjust it for the case of the reals flow in Section 6, soundness in Section 7 and prove completeness in Section 8. Section 9 has a quick overview of complexity and implementation issues.

2 The logic

Fix a countable set $\mathbf{L}$ of atoms. Here, *frames* $(T, <)$, or flows of time, will be irreflexive linear orders. *Structures* $\mathbf{T} = (T, <, h)$ will have a frame $(T, <)$ and a valuation h for the atoms, i.e. for each atom $p \in \mathbf{L}$, $h(p) \subseteq T$. Of particular importance will be *real* structures $\mathbf{T} = (\mathbb{R}, <, h)$, which have the real numbers flow (with their usual irreflexive linear ordering).

The language $L(U, S)$ is generated by the 2-place connectives U and S along with classical $\neg$ and $\wedge$. That is, we define the set of formulas recursively to contain the atoms and for formulas α and β we include $\neg\alpha$, $\alpha \wedge \beta$, $U(\alpha, \beta)$ and $S(\alpha, \beta)$.

Each formula is evaluated at a point in a structure $\mathbf{T} = (T, <, h)$. We write $\mathbf{T}, x \models \alpha$ when α is true at the point $x \in T$. This is defined recursively as follows. Suppose that we have defined the truth of formulas α and β at all points of $\mathbf{T}$. See Figure 1 for the semantics.

Common temporal abbreviations are: $F\alpha = U(\alpha, \top)$, "$\alpha$ will be true (sometime in the future)"; $G\alpha = \neg F(\neg\alpha)$, "$\alpha$ will always hold (in the future)"; and their mirror images P and H. Particularly for dense time applications we also have: $C^{+}\alpha = U(\top, \alpha)$, "$\alpha$ will be constantly true for a while after now"; and $K^{+}\alpha = \neg C^{+}\neg\alpha$, "$\alpha$ will be true arbitrarily soon". They have mirror images C^{-} and K^{-}.

A formula ϕ is $\mathbb{R}$-*satisfiable* if it has a real model: i.e. there is a real structure $\mathbf{S} = (\mathbb{R}, <, h)$ and $x \in \mathbb{R}$ such that $\mathbf{S}, x \models \phi$. A formula is $\mathbb{R}$-*valid* iff it is true at all points of all real structures. Of course, a formula is $\mathbb{R}$-valid iff its negation is not $\mathbb{R}$-satisfiable. We will refer to the logic of L(U,S) over real

For all points x:

$\mathbf{T}, x \models p$	iff	$x \in h(p)$, for p atomic;
$\mathbf{T}, x \models \neg\alpha$	iff	$\mathbf{T}, x \not\models \alpha$;
$\mathbf{T}, x \models \alpha \wedge \beta$	iff	both $\mathbf{T}, x \models \alpha$ and $\mathbf{T}, x \models \beta$;
$\mathbf{T}, x \models U(\alpha, \beta)$	iff	there is $y > x$ in T such that $\mathbf{T}, y \models \alpha$ and for all $z \in T$ such that $x < z < y$, we have $\mathbf{T}, z \models \beta$; and
$\mathbf{T}, x \models S(\alpha, \beta)$	iff	there is $y < x$ in T such that $\mathbf{T}, y \models \alpha$ and for all $z \in T$ such that $y < z < x$, we have $\mathbf{T}, z \models \beta$.

Fig. 1. Sematics

structures as RTL.

Let RTL-SAT be the problem of deciding whether a given formula of $L(U, S)$ is $\mathbb{R}$-satisfiable or not. [10] proves:

Theorem 2.1 *RTL-SAT is PSPACE-complete.*

3 Mosaics for U and S

Each mosaic is a syntactic object intended to represent a small piece, or interval, of a model, i.e. sets of formulas for a pair of points indicating which formulas are true there and in between in the whole model. There will be *coherence* conditions on the mosaic which are necessary for it to be part of a larger model. Full details, definitions and proofs can be found in [10].

Our mosaics will only be concerned with a finite set of formulas:

Definition 3.1 For each formula ϕ, define the *closure* of ϕ to be $\text{Cl}\phi = \{\psi, \neg\psi \mid \psi \leq \phi\}$ where $\chi \leq \psi$ means that χ is a subformula of ψ.

We can think of $\text{Cl}\phi$ as being closed under negation: we treat $\neg\neg\alpha$ as if it was α.

Often we will intend that a set of formulas will be exactly the set of formulas which hold at a particular point in a model. Such a set should at least be consistent in terms of classical propositional logic:

Definition 3.2 Suppose $\phi \in L(U, S)$ and $S \subseteq \text{Cl}\phi$. Say S is *propositionally consistent* (PC) iff there is no substitution instance of a tautology of classical propositional logic of the form $\neg(\alpha_1 \wedge ... \wedge \alpha_n)$ with each $\alpha_i \in S$. Say S is *maximally propositionally consistent* (MPC) iff S is maximal in being a subset of $\text{Cl}\phi$ which is PC.

We will define a mosaic to be a triple (A, B, C) of sets of formulas. The intuition is that this corresponds to two points from a structure: A is the set of formulas (from $\text{Cl}\phi$) true at the earlier point, C is the set true at the later point and B is the set of formulas which hold at all points strictly in between. Look ahead to definition 3.12 to see how mosaics can be found in a real structure.

Definition 3.3 Suppose ϕ is from $L(U, S)$. A *ϕ-mosaic* is a triple (A, B, C) of subsets of $\text{Cl}\phi$ such that:

C0. A and C are maximally propositionally consistent, and the following four *coherency* conditions hold:

C1. if $\neg U(\alpha, \beta) \in A$ and $\beta \in B$ then we have both:
C1.1. $\neg\alpha \in C$ and either $\neg\beta \in C$ or $\neg U(\alpha, \beta) \in C$; and
C1.2. $\neg\alpha \in B$ and $\neg U(\alpha, \beta) \in B$.

C2. if $U(\alpha, \beta) \in A$ and $\neg\alpha \in B$ then we have both:
C2.1 either $\alpha \in C$ or both $\beta \in C$ and $U(\alpha, \beta) \in C$; and
C2.2. $\beta \in B$ and $U(\alpha, \beta) \in B$.

C3-4 mirror images of C1-C2.

Definition 3.4 If $m = (A, B, C)$ is a mosaic then start$(m) = A$ is its *start*, cover$(m) = B$ is its *cover* and end$(m) = C$ is its *end*.

If we start to build a model using mosaics as building blocks then we may realise that the inclusion of one mosaic necessitates the inclusion of others: defects need curing.

Definition 3.5 A *defect* in a mosaic (A, B, C) is either types 1, 2 or 3:

1. a formula $U(\alpha, \beta) \in A$ with either
   1.1 $\beta \notin B$,
   1.2 ($\alpha \notin C$ and $\beta \notin C$), or
   1.3 ($\alpha \notin C$ and $U(\alpha, \beta) \notin C$);
2. mirror image for S;
3. a formula $\beta \in \mathrm{Cl}\phi$ with $\neg\beta \notin B$.

We will need to string mosaics together to build linear orders. This can only be done under certain conditions. We introduce the idea of composition of mosaics and present some results which are straightforward to prove.

Definition 3.6 We say that ϕ-mosaics (A', B', C') and (A'', B'', C'') *compose* iff $C' = A''$. In that case, their *composition* is $(A', B' \cap C' \cap B'', C'')$.

Lemma 3.7 *If mosaics m and m' compose then their composition is a mosaic.*

Lemma 3.8 *Composition of mosaics is associative.*

Thus we can talk of sequences of mosaics composing and then find their composition. We define the composition of a sequence of length one to be just the mosaic itself and we leave the composition of an empty sequence undefined. Write $\sigma = \langle m_1, m_2, ..., m_n \rangle$ for a sequence and $\sigma^\wedge\tau$ for the concatenation of two sequences.

Definition 3.9 A *decomposition* for a mosaic m is any finite sequence $\langle m_1, ..., m_n \rangle$ of mosaics which composes to m.

It will be useful to introduce an idea of fullness of decompositions. This is intended to be a decomposition which provides witnesses to the cure of every defect in the decomposed mosaic.

Definition 3.10 The decomposition above is *full* iff the following three conditions all hold:

1. for all $U(\alpha, \beta) \in A$ we have
   1.1. $\beta \in B$ and either ($\beta \in C$ and $U(\alpha, \beta) \in C$) or $\alpha \in C$,
   1.2. or there is some i with $1 \leq i < n$ such that $\alpha \in C_i$, $\beta \in B_j$ (all $j \leq i$) and $\beta \in C_j$ (all $j < i$);
2. the mirror image of 1.; and
3. for each $\beta \in \mathrm{Cl}\phi$ such that $\neg\beta \notin B$ there is some i such that $1 \leq i < n$ and $\beta \in C_i$.

If 1.2 above holds in the case that $U(\alpha, \beta) \in A$ is a type 1 defect in (A, B, C) then we say that *a cure for the defect is witnessed* (in the decomposition) by the end of (A_i, B_i, C_i) (or equivalently by the start of $(A_{i+1}, B_{i+1}, C_{i+1})$). Similarly for the mirror image for $S(\alpha, \beta) \in C$. If $\beta \in C_i$ is a type 3 defect in (A, B, C) then we also say that *a cure for this defect is witnessed* (in the decomposition) by the end of (A_i, B_i, C_i). If a cure for any defect is witnessed then we say that the defect is cured.

Lemma 3.11 *If $\langle m_1, ..., m_n \rangle$ is a full decomposition of m, then every defect in m is cured in the decomposition.*

For the reals we do not allow full decompositions of length one, although they are allowed in general linear time contexts for mosaics with no defects.

In the rest of this section we define a notion of satisfiability for mosaics and relate the satisfiability of formulas (which is our ultimate interest) to that of mosaics.

Because mosaics represent linear orders with end points, it is inconvenient for us to continue to work directly with $\mathbb{R}$ and because we want to make use of some simple tricks with convergence of sequences in the metric at several places in the proof, we will move to work in the unit interval $[0, 1]$ instead.

If $x < y$ from $\mathbb{R}$ then let $]x, y[$ denote the open interval $\{z \in \mathbb{R} \mid x < z < y\}$ and $[x, y]$ denote the closed interval $\{z \in \mathbb{R} \mid x \leq z \leq y\}$. Similarly for half open intervals.

One can get a mosaic (you can check it is a mosaic) from any two points in a structure.

Definition 3.12 If $\mathbf{T} = (T, <, h)$ is a structure and ϕ a formula then for each $x < y$ from T we define $\mathrm{mos}^{\phi}_{\mathbf{T}}(x, y) = (A, B, C)$ where:

$A = \{\alpha \in \mathrm{Cl}\phi \mid \mathbf{T}, x \models \alpha\}$,
$B = \{\beta \in \mathrm{Cl}\phi \mid$ for all $z \in T$, if $x < z < y$ then $\mathbf{T}, z \models \beta\}$, and
$C = \{\gamma \in \mathrm{Cl}\phi \mid \mathbf{T}, y \models \gamma\}$.

We will now relate the satisfiability of a formula ϕ to that of certain mosaics. Obviously, a formula will be satisfiable over the reals iff it is satisfiable over the $]0, 1[$ flow. Furthermore, this happens iff a relativized version of the formula is satisfiable somewhere in the interior of a model over $[0, 1]$. To define this relativization we need to use a new atom to indicate points in the interior. Hence the next few definitions.

Definition 3.13 Given ϕ and an atom q which does not appear in ϕ, we define a map $* = *_q^\phi$ on formulas in $\mathrm{Cl}\phi$ recursively: $*p = p \wedge q$, $*\neg\alpha = \neg(*\alpha) \wedge q$, $*(\alpha \wedge \beta) = *(\alpha) \wedge *(\beta) \wedge q$, $*U(\alpha, \beta) = U(*\alpha, *\beta) \wedge q$, and similarly S.

With the relativization machinery we can then define a relativized mosaic to be one which could correspond to the whole of a $[0, 1]$ structure in which q is true of exactly the interior $]0, 1[$ and the interior is a model of ϕ.

Definition 3.14 We say that a $*_q^\phi(\phi)$-mosaic (A, B, C) is (ϕ, q)-*relativized* iff

1. $\neg q$ is in A and C, no $S(\alpha, \beta)$ is in A, no $U(\alpha, \beta)$ in C; and
2. $q \in B$ and $\neg *_q^\phi (\phi) \notin B$.

Here we confirm that ϕ is satisfiable over the reals exactly when we can find such a relativized mosaic.

Lemma 3.15 (Lemma 29 from [10]) *Suppose that ϕ is a formula of $L(U, S)$ and q is an atom not appearing in ϕ. Then ϕ is $\mathbb{R}$-satisfiable iff there is a (ϕ, q)-relativized $*_q^\phi(\phi)$-mosaic satisfied on the whole of* $[0, 1]$.

Our satisfiability procedure in [10] was to guess a relativized mosaic (A, B, C) and then check that (A, B, C) is satisfied on the whole of $[0, 1]$. Thus we now turn to the question of deciding whether a relativized mosaic is satisfiable.

4 Real Mosaic Systems

In this section we define a concept of a collection or system of mosaics in which each member is decomposable in terms of simpler members. We will later show that being in such a system is (roughly) equivalent to satisfiability. First two of the simpler tactics for decomposition.

4.1 Tactics Lead and Trail

The mirror image tactics *lead* and *trail* allow mosaics which can be fully decomposed in terms of themselves along with some other mosaics. In a game setting this is a legitimate way for the game to be won: the player who has to keep providing full decompositions can keep supplying a full decomposition $\langle m \rangle^\wedge \sigma$ for m if the other player keeps choosing m to be decomposed. The tactic trail corresponds to an operation in [7] for building a new linear order from a simpler one by laying ω copies of it one after the other towards the future. The tactic lead corresponds to laying the copies towards the past.

Definition 4.1 Suppose $\phi \in L(U, S)$, m is a ϕ-mosaic and σ is a non-empty sequence of ϕ-mosaics. Then, we say that m is fully decomposed by the *tactic* lead(σ) iff $\langle m \rangle^\wedge \sigma$ is a full decomposition of m. We say that m is fully decomposed by the *tactic* trail(σ) iff $\sigma^\wedge \langle m \rangle$ is a full decomposition of m.

4.2 Shuffles

The term shuffle has been used in the literature (see, for example, [7], [2] or [14]) to refer to certain methods of constructing a linear structure (often a monadic one) from a thorough mixture of copies of members of a finite set of other linear structures.

Suppose that $(T_1, ..., T_n)$ are linear structures. A shuffle of the T_i is any linear order made from intervals which are each copies of one of the T_i such that between any two of the intervals lies a copy of each of the T_i. To be precise,

Definition 4.2 Suppose that $(T_1, <_1), ..., (T_n, <_n)$ are linear structures. $(K, <_K)$ is a *shuffle* of $T = \{(T_i, <_i) \mid i = 1, ..., n\}$ iff there is a linear order $(B, <_B)$ and a map $\pi : B \to \{1, ..., n\}$ such that

- $K = \bigcup_{b \in B} \{(b, t) \mid t \in T_{\pi(b)}\}$ and
- for all $b, b' \in B$, for all $t \in T_{\pi(b)}$, for all $t' \in T_{\pi(b')}$, $(b, t) <_K (b', t')$ iff either $b <_B b'$ or $b = b'$ and $t <_{\pi(b)} t'$, and
- if $b <_B b'$ then for all $i \in \{1, ..., n\}$ there is $b'' \in B$ such that $b <_B b'' <_B b'$ and $\pi(b'') = i$.

The intention here is similar except we need to deal with mosaics corresponding to linear structures instead of structures themselves. We consider (a mosaic corresponding to) a shuffle S of linear structures $U_0, U_1, \ldots, U_s$, $V_1, V_2, \ldots, V_r$ where each U_i is a singleton structure and each V_i is a non-singleton structure consisting of the concatenation of a finite sequence of (one or more) mosaics representing other structures. Thus, we actually only consider an MPC set P_i instead of U_i and a non-empty composing sequence λ_i of mosaics instead of V_i. In this case it is possible to construct a certain set of mosaics such that one, o, corresponds to S and each one in the set has a full decomposition in terms of others in the set and/or the mosaics which decompose each λ_i.

In [10], in this vein, there is a rather complex definition of when a mosaic o is *fully decomposed by the tactic shuffle* $(\langle P_0, \ldots, P_s \rangle, \langle \lambda_1, \ldots, \lambda_r \rangle)$. See Definition 31 of that paper. We will not repeat it here to save space and also to save the reader effort.

Instead we present a slightly shorter alternative characterisation that also appeared (and was proved equivalent) in that paper.

The forward $K(m)$ property is supposed to hold of an MPC set if that set could be the end of the last mosaic in some λ_i where mosaic m is fully decomposed by the tactic shuffle $(\langle P_0, \ldots, P_s \rangle, \langle \lambda_1, \ldots, \lambda_r \rangle)$. This is the set of formulas from $\mathrm{Cl}(\phi)$ true at the end point of one of the structures V_i referred to above.

Definition 4.3 (Definition 32 from [10]) Suppose $\phi \in L(U, S)$ and m is a ϕ-mosaic. We say that a set $Q \subseteq \mathrm{Cl}\phi$ satisfies the forward $K(m)$ property iff Q is MPC and for any $U(\alpha, \beta) \in \mathrm{Cl}\phi$ we have $U(\alpha, \beta) \in Q$ iff both $\beta \in \mathrm{cover}(m)$ and (at least) one of the following holds:

K1 $\neg\alpha \notin \mathrm{cover}(m)$;
K2 $\alpha \in \mathrm{end}(m)$; or
K3 $\beta \in \mathrm{end}(m)$ and $U(\alpha, \beta) \in \mathrm{end}(m)$.

The mirror image is the backwards $K(m)$ property.

Lemma 4.4 *(Lemma 33 from [10]) Suppose $\phi \in L(U,S)$, $m = (A,B,C)$ is a ϕ-mosaic, and each $P_i \subseteq \mathrm{Cl}\phi$ $(0 \leq i \leq s)$ and each λ_i $(1 \leq i \leq r)$ is a composing non-empty sequence of ϕ-mosaics.*

Then, m is fully decomposed by the tactic shuffle $(\langle P_0, \ldots, P_s\rangle, \langle \lambda_1, \ldots, \lambda_r\rangle)$ iff the following seven conditions hold:

S0 B is a subset of each P_i and of the start, end and cover of each mosaic in each λ_i;

S1 each P_i satisfies both the forward and backwards $K(m)$ property;

S2 the start of the first mosaic in each λ_i satisfies the backwards $K(m)$ property;

S3 the end of the last mosaic in each λ_i satisfies the forwards $K(m)$ property;

S4 A satisfies the forward $K(m)$ property;

S5 C satisfies the backwards $K(m)$ property;

S6 if $\beta \in \mathrm{Cl}\phi$ but $\neg\beta \notin B$ then either β is contained in some P_i or β is contained in the start or end of some mosaic in some λ_i.

Note that as $s \geq 0$ there is at least one P_i involved in the shuffle. This corresponds to a one point structure. In a general linear order setting we could define a shuffle with no P_is (provided that then $r > 0$) but over the reals it turns out to be crucial to require at least one P_i. This is because, as it is not too hard to see, a shuffle of only non-singleton closed intervals of the reals can not be both Dedekind complete and separable (i.e. having a countable dense suborder).

4.3 The levels that make an RMS

Now we define the hierarchy of membership of the system of mosaics which we need. Mosaics at one level of membership will be constructed from ones at lower levels of membership by concatenation or some combination of the tactics we have introduced above. As we build up, we only want to allow a limited use of leads and trails before a shuffle takes us to the next highest level. As we will only allow nesting of trails and/or leads of depth 2 within shuffles we define some intermediate levels between levels n and $n+1$. So, as we will see now, the levels, in increasing order are actually $0, 0^+, 1^-, 1, 1^+, 2^-, 2, 2^+, \ldots.$

Definition 4.5 For $\phi \in L(U,S)$, suppose S is a set of ϕ-mosaics and $n \geq 0$.

A ϕ-mosaic $m \in S$ is a *level n^+ member of S* iff m is the composition of a sequence of mosaics, each of them being either a level n member of S or fully decomposed by the tactics lead(σ) or trail(σ) with each mosaic in σ being a level n member of S.

A ϕ-mosaic $m \in S$ is a *level $(n+1)^-$ member of S* iff m is the composition of a sequence of mosaics, each of them being either a level n^+ member of S or fully decomposed by the tactics lead(σ) or trail(σ) with each mosaic in σ being a level n^+ member of S.

A ϕ-mosaic $m \in S$ is a *level n member of S* iff m is the composition of a sequence of mosaics with each of them being either a level n^- member of S or a mosaic which is fully decomposed by the tactic shuffle($\langle P_0, ..., P_s \rangle, \langle \sigma_1, ..., \sigma_r \rangle$) with each mosaic in each σ_i being a level n^- member of S.

Note that it is generally possible for mosaics to be level 0 members of some S provided that they are compositions of mosaics which can be fully decomposed by shuffles in which there are no sequences (i.e. $r = 0$). Thus these mosaics will have an interior which is a dense mixture of points where $P_0, ..., P_s$ hold. These are the only mosaics which can be level 0 members of any S.

Also note that if m is a level n member of S then m is the composition of $\langle m \rangle$ so m is clearly a level n^+ member of S. Similarly, level n^+ implies level $(n+1)^-$ and level n^- implies level n.

Finally, note that the set S in the definition above may not be closed under composition. It is even possible that a mosaic is a member of S at a certain level by virtue of being a composition of other mosaics each of which, although being fully decomposed by tactics involving only members of S, is not itself a members of S. Later we will see that for our purposes we mostly work with sets S which are closed under composition.

Definition 4.6 For $\phi \in L(U, S)$, a *real mosaic system* (RMS) of ϕ-mosaics is a set S of ϕ-mosaics such that, for every $m \in S$, there exists some n such that m is a level n member of S. For any n, we say that S is a real mosaic system of depth n iff every $m \in S$ is a level n member of S.

Theorem 4.7 (Theorem 75 in [10]) *Suppose ϕ is a formula of $L(U, S)$ and q is an atom not appearing in ϕ. Suppose $\psi = *^{\phi}_{q}(\phi)$ has length N.*

Then the following are equivalent:

1. *ϕ is $\mathbb{R}$-satisfiable;*
2. *there is a (ϕ, q)-relativized ψ-mosaic which appears in some RMS.*

5 Tableaux

In this section we see how the mosaics and RMS machinery can be the basis of a tableau-style decision procedure. We will start with a formula ϕ and determine whether ϕ is satisfiable in RTL or not.

The tableaux we construct will be roughly tree-shaped, albeit the traditional upside down tree with a root at the top: predecessors and ancestors above, successors and descendants below. They can be thought of as structures for organising and representing iterative full decompositions in the RMS.

We imagine trees growing downwards from the root. A node may have children immediately below it, every node except the root has a unique parent. Each node itself and its parent and the parent's parent and the parent's parent's parent etc. form the set of ancestors of the node. We will also impose an earlier-later relation between siblings (children of the same parent) on some trees and represent it by left-to-right ordering in diagrams.

Here are the basic definitions.

Definition 5.1
1. A *tree* here is just a set (of *nodes*), with a *successor* relation determining (as its transitive closure) a derived, reflexive, anti-symmetric, transitive, *ancestor* relation such that the set of ancestors of any node is finite and well-ordered (by the ancestor relation) and there is a unique *root* with no ancestors (apart from itself).

2. If node x has a successor y then we say that x is the *parent* of y (it is unique) and y is a *child* of x. Any other child of x is called a *sibling* of y. A node with no children will be called a *leaf* node.

3. The *depth* of a node with n ancestors is n.

4. An *ordered tree* is a tree with finite numbers of children for each node and a left-right relation which totally orders siblings. The left-right relation does not relate non-siblings.

5. A ϕ-mosaic labelled tree is a map from nodes of a tree to ϕ-mosaics.

The idea, as we will see, is that the labels of the children of a node form a full decomposition for the label of the node.

Definition 5.2 A (ϕ-) *tableau* (for ϕ-mosaic m) is a ϕ-mosaic labelled ordered tree with root labelled by m; and each node having the labels on the children nodes taken in order forming a full decomposition of the label on the node.

Definition 5.3 Define a leaf node to be a *clone* iff it has the same label as one of its other ancestors. Define a *complete* node of a tableau to be either a non-leaf, or a clone leaf node. Define a *successful* tableau as one in which all nodes are complete (otherwise the tableau is incomplete).

As an example see the successful $U(p,q)$-tableau in Figure 2. The three sets of formulas appearing are: $A = \{p, q, U(p,q)\}$, $B = \{\neg p, q, U(p,q)\}$ and $C = \{p, \neg q, U(p,q)\}$.

Definition 5.4 Suppose that ϕ is a formula of $L(U,S)$ and q is an atom not appearing in ϕ. Say $\psi = *_q^\phi(\phi)$. A ψ-tableau is a *tableau* for ϕ iff the root is labelled by a (ϕ, q)-relativized $*_q^\phi(\phi)$-mosaic.

6 The Reals

The mosaic tableaux of the last section were quite simple and quite general but they are not adequate for the special properties of the reals. Thus, in this section we define a $\mathbb{R}$-tableau to be a type of mosaic tableau. However, we impose some subtle restrictions on the labelling as we travel around the tree. They are essentially simple graph-theoretic properties of the labels on the decomposition tree.

First, we specify that in an $\mathbb{R}$-tableau we do not allow tableau nodes with a single child. Mosaics which have singleton sequences of themselves as full decompositions, are possible in general linear time, they are called *units*, but not allowed in the reals.

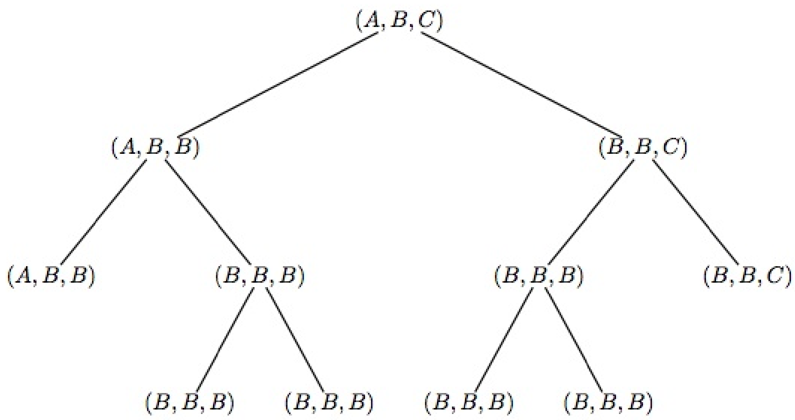

Fig. 2. A successful $U(p, q)$-tableau

6.1 Approval of the labels

Next we need some machinery to enable the other properties to be defined properly. Assume $\psi \in L(U, S)$ and suppose that T is a successful tableau of ψ-mosaics.

In order to determine whether T is a successful $\mathbb{R}$-tableau we will define an iterative process of *approving* individual mosaics in the tree. We approve the mosaic labels themselves regardless of how many times a particular label appears in the tableau. Once it is approved, it is approved everywhere that it appears.

The simplest criterion for approval is that a parent label can be approved whenever all the labels of its child nodes are approved. There are a couple of other ways to gain approval that we will outline below.

If, after some iterations, the root label in the tableau is approved then the tableau is a successful $\mathbb{R}$-tableau.

If at some stage there are no applicable rules to approve any more nodes, and the root mosaic remains unapproved then the tableau has failed to be a $\mathbb{R}$-tableau. We can terminate the check.

6.2 Trails and Leads

The following pattern in the tableau corresponds to a lead tactic and allows the mosaic m to be approved. Suppose $m = m_0$ is decomposed as $\langle m_1 \rangle ^\wedge \sigma_0$ in T, i.e. m is the label of a parent node and $\langle m_1 \rangle ^\wedge \sigma_0$ are the labels of the children in order. Suppose further that for all $i = 1, 2, ...$, m_i is decomposed as

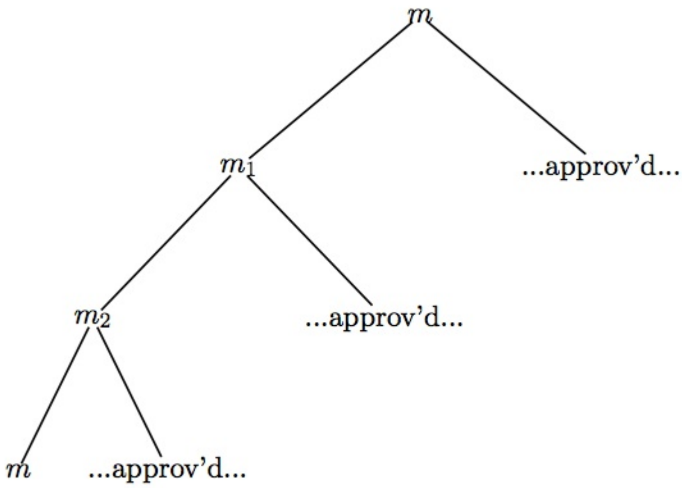

Fig. 3. Approval as a lead

$\langle m_{i+1}\rangle^\wedge\sigma_i$ in T. Suppose that all the mosaics appearing in each σ_i are already approved.

Finally suppose that some $m_i = m$.

Then we can approve m. We say that we approve m as a lead. See Figure 3 for an example of a sub-tree which leads to the approval of a mosaic as a lead.

Similarly we can approve mosaics as trails if we have a looping sequence of decompositions all ending in the m_i.

6.3 Shuffles

The pattern to allow approval of a mosaic as a shuffle is a little bit more complicated to describe and identify. It can involve a set of more than one (as yet) unapproved mosaic labels.

Because the covers of mosaics in decompositions are supersets of the cover of the parent, if there is a sequence $u = m_0, m_1, ..., m_n = v$ of mosaics in respective decompositions such that each m_i is fully decomposed (somewhere in T) as $\sigma_i{}^\wedge\langle m_{i+1}\rangle^\wedge\pi_i$ then the cover of v is a superset of the cover of u.

The conditions for approving a mosaic as a shuffle are SH1-SH6 as set out below. Consider the mosaic m appearing as a label in a tableau.

(SH1) m is an unapproved mosaic.

(SH2) Every unapproved (label of a) descendent of (a node labelled by) m, including m itself, has some descendent which has at least two separate child nodes labelled by unapproved mosaics.

(SH3) All descendants of m which are unapproved have the same cover as m.

(SH4) is the requirement that every unapproved descendent u of m (including m itself) has a "crisp start". That is, there is a sequence $u = m_0, m_1, ...$ of

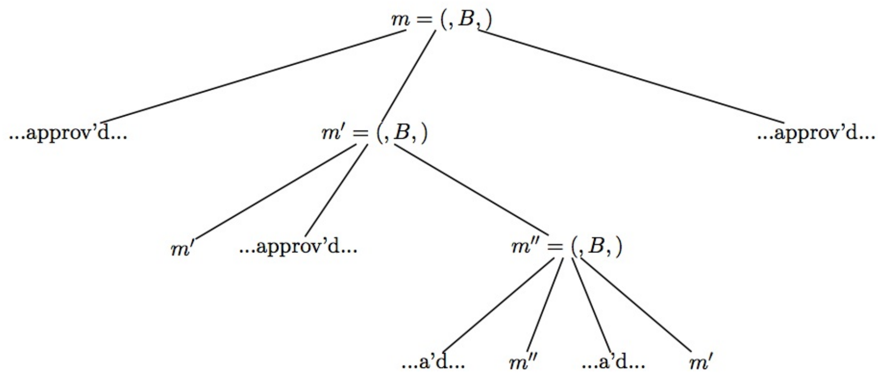

Fig. 4. Almost a shuffle

unapproved mosaics in respective decompositions as follows. Each m_i is fully decomposed as $\sigma_i{}^\wedge\langle m_{i+1}\rangle^\wedge\pi_i$ where each mosaic in σ_i is approved already (and we do not care what the π_i are). We require that $m_i = m_j$ for some $i < j$ and further, we require that for each $k = i, i+1, ..., j-1$, σ_k is actually empty.

We will see below that this condition allows us to identify a start of a possible shuffle involving u.

Similarly, (SH5), we require the unapproved descendants of m to have crisp ends using the mirror image construction.

The last check (SH6) before we approve m as a shuffle is to find an unapproved descendent u of m such that u has two adjacent children with unapproved labels v and w that further satisfy the following pattern.

We have a sequence $v = m_0, m_1, ...$ of unapproved mosaics in respective decompositions as follows. Each m_i is fully decomposed as $\sigma_i{}^\wedge\langle m_{i+1}\rangle$ where σ_i is any (even perhaps empty) sequence of any mosaics, approved or not. However, note that m_{i+1} is always the last mosaic in the decompositions for each m_i. We also have $m_i = m_j$ for some $i < j$.

The mirror image condition is required of w.

In this case it is easy to see that the end of v will be the same as the start of w. SH6 corresponds to making sure that there is a point structure taking part in the shuffle, a condition which we have seen ensures Dedekind completeness.

If SH1-6 hold then we can be sure that the shuffle is acceptable and we can approve m as a shuffle.

In Figure 4 there is a sub-tree which almost allows m to be approved as a shuffle except that condition SH6 is not established.

In Figure 5, however, m can be approved as a shuffle: condition SH6 is established with m' and m'' witnesses.

6.4 $\mathbb{R}$-tableau by approval

This concludes our account of the approval process that defines a successful $\mathbb{R}$-tableau.

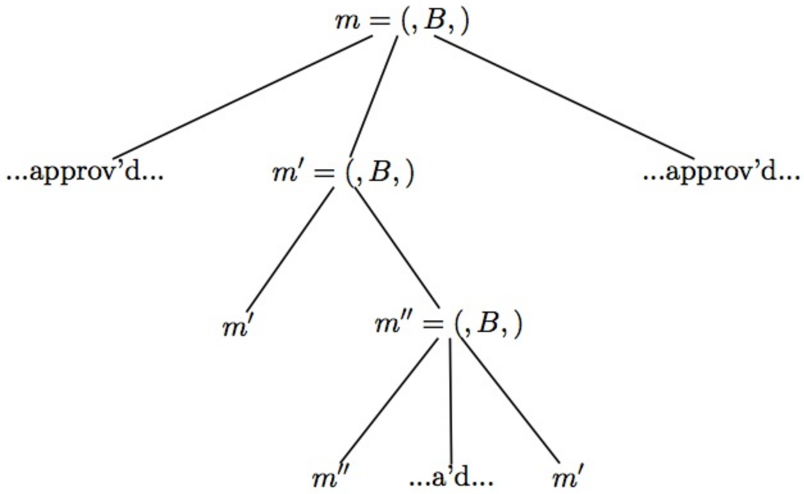

Fig. 5. Approved as a shuffle

Definition 6.1 A successful tableau is a successful $\mathbb{R}$-tableau iff all mosaic labels can be approved according to the iterative process above.

As an example, we find that the tableau in Figure 2 is a successful $\mathbb{R}$-tableau. The mosaic (B, B, B) can be approved as a shuffle, then (A, B, B) and (B, B, C) separately as a lead and trail, then (A, B, C) because its two children are approved.

The main work here is mostly in the appendices (see below). Then we can put the soundness and completeness lemmas together and get our desired overall theorem.

Theorem 6.2 *$L(U, S)$ formula ϕ is $\mathbb{R}$-satisfiable iff ϕ has a successful $\mathbb{R}$-tableau.*

7 Soundness

In [10], we define a concept of realization intended to capture the idea of a mosaic being satisfiable (over the reals) as far as internal information is concerned: i.e. we ignore formulas of the form $U(\alpha, \beta)$ in the end or $S(\alpha, \beta)$ in the start. To do so we generalise the idea of the semantic valuation function— the map which maps a time point to the set of formulas true then— to a more general class of functions (realization maps) which only have some of their properties. This is [10], definition 39:

Definition 7.1 Suppose that $x < y$ from $[0, 1]$. We say that ϕ-mosaic m is realized by the map μ on the closed interval $[x, y]$ iff the following conditions all hold:

R1. for each $z \in [x, y]$, $\mu(z)$ is a maximally propositionally consistent subset of Clϕ;

R2. Suppose $z \in [x, y[$. Then $U(\alpha, \beta) \in \mu(z)$ iff either
R2.1, there is u such that $z < u \leq y$ and $\alpha \in \mu(u)$ and
for all v, if $z < v < u$ then $\beta \in \mu(v)$ or
R2.2, $\beta \in \mu(y)$, $U(\alpha, \beta) \in \mu(y)$ and
for all v, if $z < v < y$, then $\beta \in \mu(v)$;

R3. the mirror image of R2 for $S(\alpha, \beta)$;

R4. $\mu(x)$ is the start of m;

R5. $\mu(y)$ is the end of m; and

R6. for each $\beta \in$ Clϕ, β is in the cover of m iff for all u, if $x < u < y$, $\beta \in \mu(u)$.

A mosaic m is said to be realized on $[x, y]$ iff there exists a map μ such that m is realized on $[x, y]$ by μ. Say mosaic m is realised in $[0, 1]$ iff for all $x < y$ from $[0, 1]$, there is μ such that m is realised by μ on $[x, y]$. Say that m is realised iff it is realised on $[0, 1]$.

Consider the mosaic corresponding to an interval in a structure in the sense of definition 3.12. It should be clear that this mosaic is realized by the semantic valuation function for formulas at points within the interval, i.e. the semantic valuation function is a type of realization map.

Some lemmas from [10]:

Lemma 7.2 (Lemma 41 from [10]) *Suppose ψ is a $L(U, S)$ formula, and ψ-mosaic m is the composition of m' and m'' with each of m' and m'' being realised.*

Then m is realised.

Lemma 7.3 (Lemma 42 from [10]) *Suppose ψ is a $L(U, S)$ formula, ψ-mosaic m is fully decomposed by the tactic lead σ (and similarly trail) and each mosaic in σ is realised. Then m is realised.*

Lemma 7.4 (Lemma 44 from [10]) *Suppose ψ is a $L(U, S)$ formula, ψ-mosaic m is fully decomposed by the tactic shuffle $(\langle P_0, ..., P_s \rangle, \langle \lambda_1, ..., \lambda_r \rangle)$ and each mosaic in each λ_i is realised. Then m is realised.*

7.1 Approval Implies Realised

In the next few lemmas we show that approval in a $\mathbb{R}$-tableau implies being realised.

Lemma 7.5 *Suppose ψ is a $L(U, S)$ formula and ψ-mosaic m is approved in a $\mathbb{R}$-tableau because its children are approved. Then m is realised.*

Proof. By Lemma 7.2. □

Lemma 7.6 *Suppose ψ is a $L(U, S)$ formula and ψ-mosaic m is approved in a $\mathbb{R}$-tableau as a lead. Then m is realised. Similarly trail.*

Proof. By Lemma 7.3. □

7.2 Shuffle

The final possibility is that m is approved as a shuffle. Thus we have SH1-6 as follows.

(SH1) m is an unapproved mosaic.

(SH2) Every unapproved (label of a) descendent of (a node labelled by) m, including m itself, has some descendent which has at least two separate child nodes labelled by unapproved mosaics.

(SH3) All descendants of m which are unapproved have the same cover as m.

(SH4) is the "crisp start" requirement that we outlined above. Thus, there is a sequence $u = m_0, m_1, ...$ of unapproved mosaics in respective decompositions as follows. Each m_i is fully decomposed as $\sigma_i{}^\wedge\langle m_{i+1}\rangle^\wedge\pi_i$ where each mosaic in σ_i is approved already. We have $m_i = m_j$ for some $i < j$ such that for each $k = i, i+1, ..., j-1$, σ_k is empty.

Now some useful terminology when dealing with SH4. We have let σ_i be the (possibly empty) sequence of approved mosaics in the decomposition of m_i before m_{i+1} appears. We put $pre(u) = \sigma_0{}^\wedge\sigma_1{}^\wedge...^\wedge\sigma_{i-1}$ which ay be empty.

Similarly, (SH5), we require the unapproved descendants of m to have crisp ends using the mirror image construction.

Call the corresponding sequence $post(u)$.

The last check (SH6) before we approved m as a shuffle was to find an unapproved descendent u of m such that u has two adjacent children with unapproved labels v and w that further satisfy the following pattern.

We have a sequence $v = m_0, m_1, ...$ of unapproved mosaics in respective decompositions as follows. Each m_i is fully decomposed as $\sigma_i{}^\wedge\langle m_{i+1}\rangle$ and $m_i = m_j$ for some $i < j$. Note that in that case $post(v)$ is empty. The mirror image condition applied to w and we have $pre(w)$ empty as well with the end of v being the same as the start of w.

All the above (SH1-6) were checked before we approved m.

Let K be the set of unapproved mosaics v below m.

We claim that this set defines a shuffle as follows. We define a new set Σ of mosaics and *point-structures*, i.e. MCS subsets from $\mathrm{Cl}(\phi)$.

Suppose $w \in K$ and choose a full decomposition $F(w) = \langle v_1, ..., v_k\rangle$ of w from the tree with $v_i \in K$, $v_j \in K$ and $v_k \notin K$ for all $i < k < j$, for some $i \neq j$. Say that σ is the possibly empty sequence of mosaics $v_{i+1}, v_{i+2}, ..., v_{j-1}$. For all such w, i, j, we include in Σ a mosaic or point-structure corresponding to the composition of $post(v_i)^\wedge\sigma^\wedge pre(v_j)$ if that is non-empty, or a point structure being the start of v_j otherwise.

Note that by the shuffle restriction SH6 on $\mathbb{R}$-tableaux, there will be at least one such point structure in Σ.

If we look at a decompositions of m that are deep enough below m then we can find one of the form $pre(m)^\wedge\pi^\wedge post(m)$. Just keep decomposing mosaics at the start and end.

Let s be the composition of π. By SH3 and SH2, s will have cover the same as m. In fact we will have the following: $start(s)$ is the end of $pre(m)$ (or the start of m if $pre(m)$ is empty); $cover(s)$ is the cover of m; and $end(s)$ is the start of $post(m)$ (or the ned of m if $post(m)$ is empty).

We can also show that s is fully decomposed by the tactic shuffle $(\langle\Sigma_0\rangle, \langle\Sigma'\rangle)$ where Σ_0 is the sequence of point-structures in Σ in any order and Σ' is the rest of Σ in any order.

To do so we use Lemma 4.4. Say $m = (A, B, C)$, $\Sigma_0 = \langle P_0, ..., P_s\rangle$ and $\Sigma' = \langle\lambda_1, ..., \lambda_r\rangle$.

S0) B is a subset of each P_i and of the start, end and cover of each mosaic in each λ_i. S0 holds as each point and each mosaic appears as a descendent of m which has cover B.

S1) each P_i satisfies both the forward and backwards $K(m)$ property. S1 holds as each element of Σ_0 is the start of a B cover mosaic.

Consider why $P_{i'}$ was put in Σ. There was $w \in K$ with a full decomposition $F(w) = \langle v_1, ..., v_k\rangle$ of w from the tree with $v_i \in K$, $v_j \in K$ and $v_k \notin K$ for all $i < k < j$, for some $i \neq j$. The sequence $v_{i+1}, v_{i+2}, ..., v_{j-1}$ is empty so that $j = i+1$. We include in Σ the point-structure $P_{i'}$ when $post(v_i)$, σ and $pre(v_j)$ are all empty. In that case the end of v_i and the start of v_j are the same, and that is $P_{i'}$.

For $post(v_i)$ to be empty, there is a sequence $v_i = m_0, m_1, ...$ of unapproved mosaics in respective decompositions as follows. Each m_k is fully decomposed as $\pi_k{}^\wedge\langle m_{k+1}\rangle$. Suppose $m_l = m_j$ for some $l < j$.

Thus m_l is fully decomposed as $\pi_l{}^\wedge\langle m_{l+1}\rangle$ and v_i is the composition of $\pi_0{}^\wedge\langle m_1\rangle$ which is he composition of $\pi_0{}^\wedge\pi_1{}^\wedge\langle m_2\rangle$, etc which is the composition of $\pi_0{}^\wedge\pi_1{}^\wedge...{}^\wedge\pi_{k-1}\langle m_k\rangle$.

Thus the end of v_i and the end of m_k are the same.

By noting that the cover of m_{k+1}, the last mosaic in the full decomposition $\pi_l{}^\wedge\langle m_{l+1}\rangle$ for m_k, has the same cover as m, we can deduce that the end of m_k satisfies the backward $K(m)$ condition as required.

S2) the start of the first mosaic in each λ_i satisfies the backwards $K(m)$ property. S2 holds as the first mosaic starts with the end of B mosaic.

Ditto S3. S3) the end of the last mosaic in each λ_i satisfies the forwards $K(m)$ property.

S4) A satisfies the forward $K(m)$ property. S4 holds as A starts the mosaic which starts the shuffle.

Similarly S5. S5) C satisfies the backwards $K(m)$ property.

S6) if $\beta \in \text{Cl}\phi$ but $\neg\beta \notin B$ then either β is contained in some P_i or β is contained in the start or end of some mosaic in some λ_i. S6 holds as each B mosaic gets fully decomposed and therefore there is a witness to each such β in one of the sequences that we put together to get Σ.

Recall that m is just the composition of $pre(m)^\wedge s^\wedge post(m)$ and so is realised as well.

In this subsection we have proved the following.

Lemma 7.7 *Suppose ψ is a $L(U, S)$ formula and ψ-mosaic m is approved in a $\mathbb{R}$-tableau as a shuffle. Then m is realised.*

7.3 Putting it all together

Now the main new lemma.

Lemma 7.8 *Suppose ψ is a $L(U,S)$ formula and ψ-mosaic m is approved in a $\mathbb{R}$-tableau. Then m is realised.*

Proof. We show by induction on the order of approving mosaic labels in a successful $\mathbb{R}$-tableau that all such mosaics are realised. Suppose that all mosaics so far approved are realised. Now suppose that ψ-mosaic m appears in a successful tableau T and gets approved.

There are four ways that m can get approved and we consider them case by case.

The simplest way that m is approved is when it labels a node and all the children nodes are approved. In this case we know that m is the composition of the child mosaic labels and all of those are realisable. Then m is approved by Lemma 7.5.

Another possibility is that m is approved as a lead. Use lemma 7.6 and we are done.

Similarly trail and shuffle (Lemma 7.7). □

Lemma 7.9 *Suppose ϕ is a $L(U,S)$ formula, not containing the atom q and $\psi = *_q^\phi(\phi)$. Say that there is a successful tableau for the ψ-mosaic m and it is (ϕ, q)-relativized.*

Then m is satisfied in a structure on the whole of $[0,1]$.

Proof. By Lemma 7.8, as m appears in a successful tableau then there is μ such that m is realised by μ on $[0,1]$.

As m is (ϕ, q)-relativized, m is satisfied in a structure on the whole of $[0,1]$. □

Lemma 7.10 *If $L(U,S)$ formula ϕ has a successful $\mathbb{R}$-tableau then ϕ is $\mathbb{R}$-satisfiable.*

Proof. Suppose $L(U,S)$ formula ϕ has a successful tableau.

Then there is an atom q not appearing in ϕ and $\psi = *_q^\phi(\phi)$ and ψ-mosaic m that is (ϕ, q)-relativized and has a successful tableau. It is the root of the tableau.

By Lemma 7.9, m is satisfied in a structure on the whole of $[0,1]$.

By Lemma 3.15, m, ϕ has a $\mathbb{R}$-flowed model. □

8 Completeness

Showing that satisfiable formulas have successful tableaux is not too hard when we can use the levels of an RMS and the way that we can use leads, trails and shuffles to get to the next level. In [10] it was quite clear that these operations correspond to simple repetitive patterns in a decomposition tree. They translate directly to good behaviour in tableaux.

For example, if m is fully decomposed by tactic lead applied to the sequence σ of mosaics at lower levels, then m has a tableau starting with a root with children m and then the mosaics in σ in order. There will be no central sticks because of the way leads and trails are defined. An induction takes care of the lower level σ mosaics and we are done.

Lemma 8.1 *Suppose ψ is a formula of $L(U, S)$, ψ-mosaic m is fully decomposed by the tactic lead σ (or trail) and each mosaic in σ has a successful $\mathbb{R}$-tableau in which m does not appear.*

Then m has a successful $\mathbb{R}$-tableau.

Equally, a shuffle tells us about a set of mutual decompositions which end up leaving a tableau with only lower level mosaics. See Definition 31, page 16/17 of [10]. The $\mathbb{R}$-tableau conditions can be checked directly on these decompositions.

Lemma 8.2 *Suppose ψ is a formula of $L(U, S)$, ψ-mosaic m is fully decomposed by the tactic shuffle $(\langle P_0, ..., P_s\rangle, \langle \lambda_1, ..., \lambda_r\rangle)$ and each mosaic in each λ_i has a successful $\mathbb{R}$-tableau in which m does not appear.*

Then m has a successful $\mathbb{R}$-tableau.

Put these two lemmas together in an induction and we get:

Lemma 8.3 *Suppose ψ is a formula of $L(U, S)$ and ψ-mosaic m appears in an RMS. Then m is the root of a successful $\mathbb{R}$-tableau.*

Then use the relativisation results to translate from mosaics to formulas:

Lemma 8.4 *If $L(U, S)$ formula ϕ is $\mathbb{R}$-satisfiable then ϕ has a successful $\mathbb{R}$-tableau.*

9 Termination, Complexity and Implementation Issues

It is easy to see that because we can, without loss of generality, stop at clone nodes, and limit branching factors, only a finite number of different tableaux need be considered for a formula. However, that is the end of the good news. There is an exponential bound on the number of different mosaics for a formula (in terms of its length). This also bounds the length of branches in a tableau. With a linear bound on the branching factor (—the defects need to be cured and any mosaics in between can be composed—) we thus have a double exponential bound on the size of any tableau in terms of number of nodes. There is thus a triple exponential bound on the number of tableaux which would govern the complexity of any exhaustive search through the tableaux.

However, by guessing a tableau of double exponential size we have a decision procedure that runs in 2-NEXPTIME.

Lemma 9.1 *In terms of the length of the input formula ϕ, there is a finite triple exponential bound on the number of tableaux for ϕ. A decision procedure runs in 2-NEXPTIME.*

The complexity of reasoning using such tableaux is thus 2-NEXPTIME.

In future work (joint with others) we will report on the possibilities for implementation of this technique. Early Java implementations [12] of a mosaic tableau for the logic US/LIN of $L(U, S)$ over general linear time show that any direct implementation of this tableau technique is quickly overwhelmed by the multi-exponential blow-up in data structures. The number of mosaics for a formula is a particular problem if they all need to be generated and checked. Clearly, more intelligent techniques are needed to make practical use of this basic framework. Our latest work [1] uses a notion of partial mosaics for the US/LIN case shows that there is great potential for speed-ups in practice.

Note that an implementation of the tableau reasoner for RTL would need two parts. First there is the tableau of mosaic decompositions which has a similar task to that of the US/LIN tableau in [12,1]. The second part is a much less computationally complex check through the successful tableau for the graph restrictions corresponding to approval.

References

[1] J. Bian, T. French and M. Reynolds. An Efficient Tableau for Linear Time Temporal Logic. In S Cranefield and A Nayak, editors, *26th Australasian Joint Conference on Artificial Intelligence, AI 2013, 1-6 December 2013, Dunedin, New Zealand*, Volume 8272. Pages 289–300, 2013.

[2] J. P. Burgess and Y. Gurevich. The decision problem for linear temporal logic. *Notre Dame J. Formal Logic*, 26(2):115–128, 1985.

[3] E. Emerson and E. C. Clarke. Using branching time temporal logic to synthesise synchronisation skeletons. *Sci. of Computer Programming*, 2, 1982.

[4] E. Emerson and J. Halpern. Decision procedures and expressiveness in the temporal logic of branching time. *J. Comp and Sys. Sci*, 30(1):1–24, 1985.

[5] D. M. Gabbay and I. M. Hodkinson. An axiomatisation of the temporal logic with until and since over the real numbers. *J. Logic and Computation*, 1(2):229 – 260, 1990.

[6] H. Kamp. *Tense logic and the theory of linear order.* PhD thesis, University of California, Los Angeles, 1968.

[7] H. Läuchli and J. Leonard. On the elementary theory of linear order. *Fundamenta Mathematicae*, 59:109–116, 1966.

[8] A. Rabinovich. Temporal logics over linear time domains are in PSPACE. *Inf. Comput*, 201:40–67, 2012.

[9] M. Reynolds. An axiomatization for Until and Since over the reals without the IRR rule. *Studia Logica*, 51:165–193, May 1992.

[10] M. Reynolds. The complexity of the temporal logic over the reals. *Annals of Pure and Applied Logic*, 161(8):1063–1096, 2010. Online at doi:10.1016/j.apal.2010.01.002.

[11] M. Reynolds. Dense time reasoning via mosaics. In *TIME '09: Proceedings of the 2009 16th International Symposium on Temporal Representation and Reasoning*, pages 3–10, Washington, DC, USA, 2009. IEEE Computer Society.

[12] M. Reynolds. A Tableau for General Linear Time. *Journal of Logic and Computation*, 23(5):1057–1080, 2013.

[13] M. Reynolds. A tableau for RTL (long version). Report online 2013 at `http://www.csse.uwa.edu.au/~mark/research/Online/rtltab.html`.

[14] J. G. Rosenstein. *Linear Orderings.* Academic Press, New York, 1982.