A Tableau for Until and Since over Linear Time

Mark Reynolds,
The University of Western Australia.

August 3, 2010

Abstract

We use mosaics and games to provide a simple, sound and complete tableau reasoning procedure for the temporal logic of until and since over general linear time.

1 Introduction

Temporal logic is a widely used formalism for reasoning about hardware and software systems. Propositional reasoning on a natural numbers model of time and its logic PLTL [Pnu77] is the standard approach. However, other models of time may be better for many applications, ranging from philosophical, natural language and AI modelling of human reasoning to computing and engineering applications of concurrency, refinement, open systems, analogue devices and metric information. See for example [KMP94], [BG85] or [Rey09a].

For these sorts of applications, a range of logics have been developed such as intervals [HS86], metric temporal logics [AH93], the duration calculus [CHR91], rational numbers time [GHR93], and real time with finite variability [Rab98, KMP94]. Many such problems, however, can be expressed easily by various members of a family of natural and expressive logics using the Until and Since connectives introduced in [Kam68].

The propositional language $L(U, S)$ with Until and Since can be given a uniform and natural semantics over any class of linear orders (at least). Thus there is a basic logic which we will call US/LIN which is the logic of $L(U, S)$ over the class of all linear flows of time. Other, more specialised logics, are then obtained by restricting the application of the semantics to sub-classes of linear orders. One such logic is U and S over the reals which is called RTL in recent work in [Rey10b]. Another is the logic of $L(U, S)$ over the natural numbers which is as expressive as PLTL with past operators.

We have, from [Bur82], a complete axiom system to allow derivation of the valid formulas of US/LIN. Decidability of US/LIN follows directly from the fact that the universal monadic theory of the class $\mathcal{L}$ of all linear flows of time is decidable: and this was first proved by Gurevich in [Gur64]. This is discussed more fully in [GHR94] where the decidability is also derived from the case of decidability of $L(U, S)$ over the rationals. The complexity of the decision problem was investigated in [Rey10a] where it is proved that US/LIN is in PSPACE (so no more complex than PLTL [SC85]).

The PSPACE proof in [Rey10a] was built on the recent result in [Rey10b] that deciding RTL is PSPACE-complete. The RTL proof uses linear time mosaic techniques. Mosaics were used to prove decidability in relation algebras [Nem95] and have been used since quite generally in algebraic logic and modal logic [MMR00, Rey03, HHM$^+$99]. These mosaics are small pieces of a model, in our case, a small piece of a linear-flowed structure. We decide whether a finite set of such small pieces is sufficient to be used to build a model of a given formula.

In this paper we portray the task of finding a suitable set of mosaics as a two-player game in the style of the logic games of [Ehr61] and [Fra54]. Games have often been used in temporal logic [HH02, LS02]. The existence of the suitable mosaics is equivalent to the existence of a winning strategy for one player in the game played with mosaics. We also see that the search for a winning strategy can be arranged into a search through a tree of mosaics. This leads us on to the tableau.

Tableaux are a popular style of modal reasoning technique and there has been a substantial amount of work on applying them to temporal logics: see [Gor99] and [RD05] for surveys. They can be presented in an intuitive way, they are often suitable for automated reasoning and it is often not hard to prove complexity results for their use. Tableaux were used for modal logics in [HC68] and [Fit83] and there has been much work since on tableaux for temporal logics [Wol85, Gou89, EC82, EH85, Sch98, MMR00].

It was noted in [Rey10b] and again in [Rey09a] that the mosaic-based complexity proof for RTL also vaguely suggests a tableau based method for determining RTL validity. In this paper, progress this research programme by providing full details of a mosaic-based and game-based tableau system for US/LIN. It is hoped that in future work the tableau could be built upon and modified to be able to handle RTL-validity instead of US/LIN validity.

In this paper we define the logic US/LIN in section 2, explain mosaics in section 3, play games in section 4, lay out the tableau in section 5, face complexity issues in section 6 and conclude in section 7.

2 The logic

Fix a countable set $\mathbf{L}$ of propositional atoms. *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 $p \in \mathbf{L}$, $h(p) \subseteq T$. The idea is that if $t \in h(p)$ then the proposition p is true at time

t.

The language $L(U,S)$ is generated by the 2-place connectives U (Until) and S (Since) 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 $S(\alpha,\beta)$.

Note that in most of the literature on temporal logics *for discrete time*, the "until" connective is written in an infix manner: $\beta U \alpha$ rather than $U(\alpha,\beta)$. This corresponds to the natural language reading "I will be here until I become hungry" rather than our equally natural alternative "until I am hungry, I will be here". We choose to use the prefix notation for until and since because it agrees with important previous work on the language for general or dense linear time such as [Kam68], [BG85], [GHR94] and [Rey10b]. It also helps us distinguish our connective from the until connective usually seen with discrete time which is written in an infix manner and which is a slightly different connective, the *non-strict* until connective mentioned below.

Formulas are evaluated at points in structures $\mathbf{T} = (T,<,h)$. 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 $\mathbf{T}$. Then 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 $\mathbf{T}, z \models \beta$.

Many definitions, results and proofs below have a *mirror image*. This means that U and S are swapped and $<$ and $>$ are swapped.

2.1 Abbreviations

There are many common and generally useful other connectives which can be defined as abbreviations in the language. These include the classical $\alpha \vee \beta = \neg(\neg\alpha \wedge \neg\beta)$; $\top = p \vee \neg p$ (where p is some particular atom from $\mathbf{L}$); $\bot = \neg\top$; and $\alpha \rightarrow \beta = (\neg\alpha) \vee \beta$.

Then there are the common temporal ones: $F\alpha = U(\alpha,\top)$, "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: $\Gamma^{+}\alpha = U(\top,\alpha)$, "alpha will be constantly true for a while after now"; and $K^{+}\alpha = \neg\Gamma^{+}\neg\alpha$, "alpha will be true arbitrarily soon". They have mirror images Γ^{-} and K^{-}. See [GHR94] for details.

The non-strict "until" connective [SC85], used in PLTL and other temporal logics over the natural numbers (i.e. over sequences of states) is just "α until β" given as $\beta \vee (\alpha \wedge U(\beta, \alpha))$ in terms of our strict "until". There is a mirror image non-strict "since". Comparisons between strict and non-strict connectives are discussed more fully in [Rey03].

In situations where time is discrete then we can also define "next" $X\alpha \equiv U(\alpha, \bot)$ and (strong) yesterday $Y\alpha \equiv S(\alpha, \bot)$. Weak yesterday is $\neg S(\neg\alpha, \bot)$. It is not sensible to use this connective with dense time.

2.2 Validity and Satisfiability

US/LIN is the logic with $L(U, S)$ formulas evaluated over the class of all linear structures.

We say that an $L(U, S)$ formula is *valid* (in US/LIN) iff for all linear structures $\mathbf{T} = (\mathbf{T}, <, \mathbf{g})$, for all $x \in T$, $\mathbf{T}, x \models \phi$. In that case write $\models \phi$.

As usual, satisfiability is dual to validity: we say that an $L(U, S)$ formula is *satisfiable* iff there is a linear structure $\mathbf{T} = (\mathbf{T}, <, \mathbf{g})$, and some $x \in T$, such that $\mathbf{T}, x \models \phi$.

We can also use validity to determine a finitary consequence relation. Say that γ is a *consequence* of $\Gamma = \{\gamma_1, ..., \gamma_n\}$ iff for all linear structures $\mathbf{T} = (\mathbf{T}, <, \mathbf{g})$, for all $x \in T$, if for all $i = 1, ..., n$, $\mathbf{T}, x \models \gamma_i$ then $\mathbf{T}, x \models \gamma$. In that case write $\Gamma \models \gamma$. It is clear that $\Gamma \models \gamma$ iff $\models \bigwedge_{i=1}^{n} \gamma_i \rightarrow \gamma$.

2.3 Reasoning with US/LIN

A Hilbert style axiom system for US/LIN comes from [Bur82] (with streamlining by Ming Xu [Xu88]). The system has the usual rules for a temporal logic: i.e. modus ponens, generalizations and substitution:

$$\frac{A, A \rightarrow B}{B} \qquad \frac{A}{GA} \quad \frac{A}{HA} \quad \frac{A(q)}{A(q/B)}$$

The axioms are all classical tautologies and the six Burgess-Xu axioms:

$$G(p \rightarrow q) \rightarrow (U(p, r) \rightarrow U(q, r))$$

$$G(p \rightarrow q) \rightarrow (U(r, p) \rightarrow U(r, q))$$

$$p \wedge U(q, r) \rightarrow U(q \wedge S(p, r), r)$$

$$U(p, q) \rightarrow U(p, q \wedge U(p, q))$$

$$U(q \wedge U(p, q), q) \rightarrow U(p, q)$$

$$\begin{aligned} &U(p, q) \wedge U(r, s) \rightarrow \\ &\qquad U(p \wedge r, q \wedge s) \vee U(p \wedge s, q \wedge s) \vee U(q \wedge r, q \wedge s) \end{aligned}$$

along with each of their mirror images.

Decidability follows the fact that the universal monadic theory of the class $\mathcal{L}$ of all linear flows of time is decidable as first proved by Gurevich in [Gur64]. See also [BG85] and [GHR94].

It also follows from the recent complexity result in [Rey10a]. The complexity of a decision procedure for US/LIN was determined as part of a wide range of linear temporal logic complexity results in [Rey10a]. A PSPACE result was provided, although it is not known if the logic is PSPACE-complete.

3 Mosaics for U and S

We will decide the satisfiability of formulas by considering sets of simple labelled structures which represent small pieces of linear structures. The idea is based on the mosaics seen in [Nem95] and used in many other subsequent proofs.

Each mosaic is a small piece of a model, i.e. a small set of objects (points), relations between them and a set of formulas for each point indicating which formulas are true there in the whole model. There will be (syntactic) *coherence* conditions on the sets of formulas which make up the mosaic which are necessary for it to be part of a larger model.

We want to show the equivalence of the existence of a model to the existence of a certain set of mosaics: enough mosaics to build a whole model. So the whole set of mosaics also has to obey some conditions. These are called *saturation* conditions. For example, a particular small piece of a model might require a certain other piece to exist somewhere else in the model. We talk of the first mosaic having a *defect* which is cured by the latter mosaic.

3.1 A Mosaic

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

DEFINITION 1 *For each formula ϕ, define the* closure *of ϕ to be* $\mathrm{Cl}\phi = \{\psi, \neg\psi \mid \psi \leq \phi\}$ *where $\chi \leq \psi$ means that χ is a subformula of ψ.*

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

Various parts of each mosaic, consist in a set of formulas which is intended to be a set of formulas which all hold at one point in a model. Thus such a set should be at least consistent in terms of classical propositional logic:

DEFINITION 2 *Suppose $\phi \in L(U, S)$ and $S \subseteq \mathrm{Cl}\phi$. 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* $\mathrm{Cl}\phi$ *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 $\mathrm{Cl}\phi$) 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.

Given two points in a structure we can find such a triple as follows:

DEFINITION 3 *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:*

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

If $\mathbf{T}$ and ϕ are clear from context then we just write $\mathrm{mos}(x, y)$ for $\mathrm{mos}^{\phi}_{\mathbf{T}}(x, y)$.

As we will see, definition 3 gives us actual satisfiable mosaics once we have a model. However, in computing a decision procedure we have no model to build from and we want to construct such a triple as a syntactic object. Thus we define a mosaic to be a triple of sets of formulas satisfying some reasonable but simple syntactic requirements: the coherency conditions.

The coherency conditions are given as part of the following definition. It will be easy to see that they are *necessary* for a mosaic to represent a small part of a linear structure. However, they are only simple syntactic criteria and are therefore not subtle enough to be also *sufficient* for a mosaic to represent a piece of structure. Thus, as we will see later, an important task in this paper is to identify which mosaics are actually *satisfiable.*

DEFINITION 4 *Suppose ϕ is from $L(U, S)$. A ϕ-*mosaic *is a triple (A, B, C) of subsets of* $\mathrm{Cl}\phi$ *such that:*

C0.1 *A and C are maximally propositionally consistent, and*

C0.2 *for all $\neg\neg\beta \in \mathrm{Cl}\phi$ we have $\neg\neg\beta \in B$ iff $\beta \in B$*

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. *if* $\neg S(\alpha, \beta) \in C$ *and* $\beta \in B$ *then we have both:*
C3.1 $\neg\alpha \in A$ *and*
either $\neg\beta \in A$ *or* $\neg S(\alpha, \beta) \in A$*; and*
C3.2 $\neg\alpha \in B$ *and* $\neg S(\alpha, \beta) \in B$*.*

C4. *if* $S(\alpha, \beta) \in C$ *and* $\neg\alpha \in B$ *then we have both:*
C4.1 either $\alpha \in A$
or both $\beta \in A$ *and* $S(\alpha, \beta) \in A$*; and*
C4.2 $\beta \in B$ *and* $S(\alpha, \beta) \in B$*.*

The reader can check that these coherence conditions are reasonable (i.e. sound) in terms of the intended meaning of a mosaic. For example, considering C2.2, if $U(\alpha, \beta)$ holds at one point x of some structure and $\neg\alpha$ holds at all points between x and $y > x$, then it is clear from the semantics of U that there must be some $z \geq y$ with α true there and β (and so also $U(\alpha, \beta)$) holding everywhere between x and y and beyond until z.

Continuing in such a way through all the conditions, it is straightforward to show "the soundness of coherence" i.e. that any $\mathrm{mos}^{\phi}_{\mathbf{T}}(x, y)$ is a mosaic.

LEMMA 5 *For any structure* $\mathbf{T}$ *and any* $L(U, S)$ *formula* ϕ*,* $\mathrm{mos}^{\phi}_{\mathbf{T}}(x, y)$ *is a mosaic.*

Some helpful terminology:

DEFINITION 6 *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*.*

3.2 Defects

If we start to build a model using mosaics as building blocks then we may realize that the inclusion of one mosaic necessitates the inclusion of others. If we claim to have in a certain set all the mosaics needed to build a model, then the other mosaics should be in our set too. For example,—this is 1.2 below—if we have $U(\alpha, \beta)$ holding at $x < y$ and neither α nor β true at y then it is clear that there is a point z with $x < z < y$, α true at z and β true everywhere between x and z. This is an example of what we will call a *defect* in the mosaic. If there is such a point z and we claim to have a sufficient set of mosaics then we should have the mosaics corresponding to the pairs (x, z) and (z, y) as well as (x, y). These other mosaics are said to *cure* the defect in the first mosaic. Below we will see that we cure defects *en masse* via a whole sequence of other mosaics rather than just having a pair to cure one defect at a time as in this example.

DEFINITION 7 *A* defect *in a mosaic (A, B, C) is either*

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. a formula $S(\alpha, \beta) \in C$ with either
2.1 $\beta \notin B$,
2.2 ($\alpha \notin A$ and $\beta \notin A$), or
2.3 ($\alpha \notin A$ and $S(\alpha, \beta) \notin A$);
3. a formula $\beta \in \mathrm{Cl}\phi$ with $\neg\beta \notin B$; or
4. a formula $U(\alpha, \beta) \in C$;
5. a formula $S(\alpha, \beta) \in A$.

We refer to defects of type 1 to 5 (as listed here). Note that the same formula may be at the same time one type of defect in a mosaic and another type of defect in the same mosaic. Or it could even be three different types of defect. For example, a formula may be a type 1, a type 3 and a type 4 defect in the same mosaic. In such cases we count it as multiple separate defects.

Types 1,2 and 3 are called *internal* defects while 4 and 5 are *external.* Note that in earlier work using mosaics [Rey03, Rey10b] for linear time, the author used a slightly different approach and did not consider external defects.

We will need to string mosaics together to build linear orders. This can only be done under certain conditions. Here we introduce the idea of composition of mosaics.

DEFINITION 8 *We say that ϕ-mosaics (A', B', C') and (A'', B'', C'')* compose *iff $C' = A''$. In that case, their* composition *is $(A', B' \cap C' \cap B'', C'')$.*

It is straightforward to prove that this is a mosaic and that composition of mosaics is associative.

LEMMA 9 *If mosaics m and m' compose then their composition is a mosaic.*

LEMMA 10 *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. We leave the composition of an empty sequence undefined.

We write sequences as follows:

$$\langle (A_1, B_1, C_1), (A_2, B_2, C_2), ..., (A_n, B_n, C_n) \rangle.$$

If σ is a finite sequence then $\sigma\, ^\wedge \rho$ is the *concatenation* of σ followed by ρ.

DEFINITION 11 *A mosaic with no internal defects is called a* unit.

DEFINITION 12 *A mosaic with no defects at all is called* perfect.

DEFINITION 13 *A* decomposition *for a mosaic (A, B, C) is any finite sequence of mosaics which composes to (A, B, C).*

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 internal defect in the decomposed mosaic.

DEFINITION 14 *The decomposition*

$$\langle (A_1, B_1, C_1), (A_2, B_2, C_2), ..., (A_n, B_n, C_n) \rangle$$

is full *iff the following three conditions all hold:*

1. for all $U(\alpha, \beta) \in A$ we have either
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 15 *If $\langle m_1, ..., m_n \rangle$ is a full decomposition of m then every internal defect in m is cured in the decomposition.*

Note that if m is a unit then $\langle m\rangle$ is a full decomposition of m.

In order to handle external defects as well we now employ a pair of similar concept which allow cures for external defects to be witnessed.

DEFINITION 16 *A* full left expansion *for a mosaic* (A, B, C) *is any finite, possibly empty, composing sequence of mosaics*

$$\langle(A_1, B_1, C_1), (A_2, B_2, C_2), ..., (A_n, B_n, C_n)\rangle$$

such that $C_n = A$ *composes with* (A, B, C) *and for all* $S(\alpha, \beta) \in A$ *there is some* i *with* $1 \leq i \leq n$ *such that* $\alpha \in A_i$, $\beta \in B_j$ *(all* $j \geq i$*) and* $\beta \in C_j$ *(all* j, $i < j < n$*).*

Full right expansion is mirror.

We say that a triple of finite sequences of mosaics is a *full expansion* of a mosaic iff it consists of a full left expansion, a full decomposition and a full right expansion of the mosaic.

Note that a perfect mosaic m has $(\langle\rangle, \langle m\rangle, \langle\rangle)$ as a full decomposition.

3.3 Satisfiability of Mosaics

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

DEFINITION 17 *Say that* ϕ*-mosaic* m *is satisfible iff there is* $\mathbf{T}$ *and* $x < y$ *from* T *such that* $m = \mathrm{mos}^{\phi}_{\mathbf{T}}(x, y)$*.*

LEMMA 18 *Each satisfiable mosaic has a full expansion consisting only of satisfiable mosaics.*

> PROOF: Consider ϕ, $\mathbf{T}$ and $x, y \in T$ such that $x < y$. Let $m = (A, B, C) = \mathrm{mos}^{\mathbf{T}}_{\phi}(x, y)$. We will work on the left expansion, decomposition and right expansion separately.
>
> First consider the decomposition. There are three types of internal defect.
>
> 1. a formula $\delta = U(\alpha, \beta) \in A$ with $\beta \notin B$. In this case there must be $u \in T$ such that $x < u < y$ and $\mathcal{T}, u \models \alpha$ and for all $s \in T$ with $x < s < u$ we have $\mathcal{T}, s \models \beta$. Let $u_\delta = u$.
> 2. $S(\alpha, \beta)$ is mirror.
> 3. a formula $\delta = \beta \in \mathrm{Cl}(\phi)$ such that $\neg\beta \notin B$. In this case there must be $v \in T$ such that $x < v < y$ and $\mathcal{T}, v \models \beta$. Let $v_\delta = v$.

Put the various u_δ and v_δ in a sequence in order as $z_1 < z_2 < ... < z_n$. The decomposition is simply $\langle \mathrm{mos}(x, z_1), \mathrm{mos}(z_1, z_2), ..., \mathrm{mos}(z_{n-1}, z_n), \mathrm{mos}(z_n, y) \rangle$. It is clear that we have a full decomposition and that all the mosaics are satisfiable.

Now consider a left expansion. (The right expansion is mirror.) Suppose that we have a formula $\delta = S(\alpha, \beta) \in A$. In this case there must be $u \in T$ such that $u < x$ and $\mathcal{T}, u \models \alpha$ and for all $s \in T$ with $u < s < x$ we have $\mathcal{T}, s \models \beta$. Let $u_\delta = u$.

Put the various u_δ in a sequence in order as $z_1 < z_2 < ... < z_n < x$. The left expansion is simply $\langle \mathrm{mos}(z_1, z_2), ..., \mathrm{mos}(z_{n-1}, z_n), \mathrm{mos}(z_n, x) \rangle$. It is clear that we have a full left expansion and that all the mosaics are satisfiable. □

A mosaic for ϕ is a ϕ-mosaic (A, B, C) such that either $\phi \in A$ or $\phi \in C$. Thus a formula is satisfiable iff it has a one point model or there is a satisfiable mosaic for it.

4 The Game

Our US/LIN satisfiability procedure will be to guess a mosaic (A, B, C) for ϕ and then check that (A, B, C) is satisfiable. Thus we now turn to the question of deciding whether a mosaic is satisfiable. This is done via a game.

We introduce a game for two players. The game can be used to summarise the organisation of a structured collection of mosaics so that we can effectively find cures for any defects in any mosaic in the collection.

DEFINITION 19 *For each ϕ in the language $L(U, S)$ and each ϕ-mosaic m, there is a two-player game called the m-game. The players, A and E say, have alternate moves. E has the first move. If m has no defects then E wins immediately. Otherwise, E must place a triple of finite sequences of ϕ-mosaics on the table which, taken in order, form a full expansion of m.*

Then, and subsequently, there will be a sequence of mosaics on the table and A's move is to choose one of them. If there is no defect in this mosaic then E wins. Otherwise, E must clear the table and present a full expansion of the chosen mosaic. Then it is A's turn again.

If E fails to be able to make a legal move then she loses. If the game continues for ω moves then E wins.

A *strategy* for a player is just a map which specifies possible next moves for the player at each round: given the sequence of moves up until the player's turn in some finite round, the map defines a set of possible moves. We say that a player plays according to a strategy if the player always selects a move from the set specified at each turn in the play of the game. We say that the strategy is *winning* iff the player wins every possible game in which he or she plays according to the strategy.

We will show that the satisfiability of mosaics is equivalent to the existence of a winning strategy for E. This is important as it provides a very natural foundation for many possible approaches to reasoning with linear temporal logics.

LEMMA 20 *If m is satisfiable then E has a winning strategy in the game for m.*

> PROOF: In fact, we know by lemma 18 above that if E ever has to fully expand a satisfiable mosaic then she can play so that all the mosaics which she uses are themselves satisfiable. Thus, at the next move A loses or A has to choose a satisfiable mosaic. By keeping this up E will go on to win. □

We now show that if E has a winning strategy in the game for a mosaic then the mosaic is satisfiable. By playing many games at once we can use E's plays to gradually build up the details of a model of the mosaic.

First, note that if E has a winning strategy then she has a deterministic one, i.e. at most one move is specified for every possible round. Because she can win by making the game go on forever she can use a strategy which depends only on the mosaic in front of her and not on the history of play up until then: if she has a not necessarily deterministic strategy she just needs to fix on any one of the responses which she might use for any mosaic that arises.

LEMMA 21 *Suppose that ϕ is a formula of $L(U, S)$. Say m is a ϕ-mosaic and that E has a winning strategy in the m-game.*

Then m is satisfiable.

> PROOF: Say $m = (A_0, B_0, C_0)$.
>
> We will build a model of m from the limit of a construction in which a labelled structure is extended by the addition of new points at each step. Unfortunately, we can not just examine one play of the game to build the structure because any one play will only be concentrating on an ever diminishing local region of the structure. Thus, we need to start off an infinite number of plays of the game and at each step we divide the set of plays up into infinite subsets each of which is concentrating on one part of the structure.
>
> The infinitely many plays of the m-game, called game instances, are played in parallel with E using her deterministic winning strategy in each one. So we need to suppose that we have ω copies of the game table, ω copies of the player E and ω

copies of the player A. For each $i < \omega$, the ith game instance will have the ith version of E sitting across the ith version of the table from the ith version of A. By progressively dividing up the set of game instances into subsets, we build a model of m.

We shall specify the play of A in each of these game instances. Some of the concurrently played games may go on forever but some A will lose finitely. At the start of the ith round there will be an infinite set $G^i_{(r,s)}$ of game instances for each adjacent $r < s$ from a finite set Q_i of rationals. The $G^i_{(r,s)}$ are all disjoint. All the game instances in any $G^i_{(r,s)}$ will agree up to the ith round.

We will also define a map m_i from adjacent pairs of rationals in Q_i into the set of ϕ-mosaics. For adjacent r, s from Q_i, $m_i(r, s)$ will be one of the mosaics on the table at the start of the ith round in any game in $G^i_{(r,s)}$.

Finally, we will define a maximally propositionally consistent set $\lambda(r)$ of formulas (from $\mathrm{Cl}(\phi)$) for each r which ever appears in any Q_i. Each $m_i(r, s)$ will start with $\lambda(r)$ and end with $\lambda(s)$, i.e. the start of $m_i(r, s)$ is $\lambda(r)$ and the end of $m_i(r, s)$ is $\lambda(s)$ independently of i.

We make sure that the following *inheritance* conditions are preserved (for each $i < j$):

- if $r \leq r' < s' \leq s$, r and s are adjacent in Q_i, and r' and s' are adjacent in Q_j then the cover of $m_j(r', s')$ contains the cover of $m_i(r, s)$;
- if $r < t < s$, r and s are adjacent in Q_i, and $t \in Q_j$ then $\lambda(t)$ contains the cover of $m_i(r, s)$.

We start with one infinite set $G^0_{(0,1)}$ containing all ω of the game instances, $Q_1 = \{0, 1\}$, $m_1(0, 1) = (A_0, B_0, C_0)$, $\lambda(0) = A_0$ and $\lambda(1) = C_0$. In all games, (A_0, B_0, C_0) is on the table. Now we are ready to move on to round 1.

During the parallel playing of all the game instances we proceed as follows. At the start of the ith round we consider each adjacent pair $r < s$ from Q_i separately. If the games in $G^i_{(r,s)}$ have not finished then we get all the player As in these games to choose the mosaic $m_i(r, s)$ which is one of those on the table.

If there is no defect in $m_i(r, s)$ then those games just stop (E has won).

If there is a defect in $m_i(r, s)$ then, as E is using a winning strategy, she will choose a full expansion of that mosaic.

Now we define Q_{i+1}, G^{i+1} and m_{i+1}. Q_{i+1} contains Q_i. Into Q_{i+1} we also put $n - 1$ distinct rationals between r and s if E played a decomposition of length n (as part of the expansion) in the games in $G^i_{(r,s)}$. For each adjacent pair $r < s$ from Q_i for which the games in $G^i_{(r,s)}$ have all stopped, we put $G^{i+1}_{(r,s)} = G^i_{(r,s)}$ and $m_{i+1}(r,s) = m_i(r,s)$. We actually ignore external witnesses, i.e. the left and right expansion parts of the play, unless s is the greatest element of Q_i or r is the least (or both).

However, if s is the greatest element of Q_i, and r is second greatest, then look at the right expansion played by E in $G^i_{(r,s)}$. If E played a right expansion of length n then put n new points of $\mathbb{Q}$ after s in Q_{i+1}. (Similarly use a left expansion for the smallest element of Q_i.)

For each adjacent pair $r < s$ from Q_i, if an n long decomposition was played in $G^i_{(r,s)}$, and so Q_{i+1} contains $r < r_1 < r_2 < ... < r_{n-1} < s$, we divide the games in $G^i_{(r,s)}$ into n disjoint infinite sets and call these $G^{i+1}_{(r,r_1)}$, $G^{i+1}_{(r_1,r_2)}, ..., G^{i+1}_{(r_{n-2},r_{n-1})}$ and $G^{i+1}_{(r_{n-1},s)}$. We define $m_{i+1}(r,r_1)$ to be the first mosaic in the decomposition, each $m_{i+1}(r_j,r_{j+1})$ to be the $(j+1)$th mosaic in the decomposition and $m_{i+1}(r_{n-1},s)$ to be the last mosaic in the decomposition.

For the greatest adjacent pair $r < s$ from Q_i, if an n long right expansion was played in $G^i_{(r,s)}$, and so Q_{i+1} contains $s < r_1 < r_2 < ... < r_n$, we reserve another disjoint infinite subset of $G^i_{(r,s)}$ and we divide it into n disjoint infinite sets and call these $G^{i+1}_{(s,r_1)}$, $G^{i+1}_{(r_1,r_2)}, ..., G^{i+1}_{(r_{n-2},r_{n-1})}$ and $G^{i+1}_{(r_{n-1},r_n)}$. We define $m_{i+1}(s,r_1)$ to be the first mosaic in the expansion, each $m_{i+1}(r_j,r_{j+1})$ to be the $(j+1)$th mosaic in the expansion and $m_{i+1}(r_{n-1},r_n)$ to be the last mosaic in the expansion.

The definition of composition allows us to see that our inheritance conditions are preserved.

Some of the games may go on forever. Nevertheless, the E player will win them all.

Now we define $T = \bigcup Q_i$ and order it via the usual ordering on rationals. Then we define a structure $\mathcal{T} = (T, <, h)$ where h is defined by $r \in h(p)$ iff $p \in \lambda(r)$, for any atom $p \in \mathbf{L}$.

We are almost done when we prove the truth lemma 22 below. Assume that we have proved that now. To complete the main proof, we just need check that the inheritance conditions and truth lemma give us $m = \text{mos}^{\phi}_{\mathcal{T}}(0,1)$. It is clear that A_0 contains exactly the formulas true at 0 and C_0 those at 1.

Certainly $B_0 \subseteq \bigcap_{s\in(0,1)} \lambda(s)$ by inheritance.

To see that $\bigcap_{s\in(0,1)} \lambda(s) \subseteq B_0$ suppose that $\beta \in \text{Cl}(\phi) \setminus B_0$. Thus $\neg\beta$ is a type 3 defect in m and will be cured in the very first move by E in all the game instances. Suppose that the witness to the cure is put at $r_j \in (0,1)$ in Q_2. Thus $\neg\beta \in \lambda(r_j)$.

We are done. $\Box$

So it remains to prove the truth lemma:

LEMMA 22 *For each $r \in T$ and $\alpha \in \mathrm{Cl}(\phi)$,*

$$\mathcal{T}, r \models \alpha \text{ iff } \alpha \in \lambda(r).$$

PROOF: We proceed by induction on the construction of α. The cases of atoms, truth and the Boolean connectives are straightforward. The harder cases concern both directions for a formula $U(\alpha,\beta)$. ($S(\alpha,\beta)$ is mirror.)

($\Rightarrow$) Suppose $\mathcal{T}, r \models U(\alpha,\beta)$. So there is some $s \in T$ such that $r < s$, $\mathcal{T}, s \models \alpha$ and for each $t \in T$, if $r < t < s$ then $\mathcal{T}, t \models \beta$. By the induction hypothesis $\alpha \in \lambda(s)$ and $\beta \in \lambda(t)$ for any such t.

We need to show $U(\alpha,\beta) \in \lambda(r)$. Suppose for contradiction that $\neg U(\alpha,\beta) \in \lambda(r)$.

Now look at any Q_i containing both s and r. Let r' be the greatest element of Q_i which is less than s but for which we have $\neg U(\alpha,\beta) \in \lambda(r')$. So $r \leq r' < s$.

Let $s' \in Q_i$ be the next biggest element of Q_i after r'. Say that $m_i(r',s') = (A,B,C)$. We may have $s' = s$ or $s' < s$. If $s = s'$ we have $\neg U(\alpha,\beta) \in \lambda(r')$ and $\alpha \in \lambda(s')$. Coherency condition 1 gives us $\beta \notin B$. If $s' < s$, $U(\alpha,\beta)) \in \lambda(s')$ (by definition of r') and $\beta \in \lambda(s')$ (as $r < s' < s$). Coherency condition 1.2 gives us $\beta \notin B$. In either case, $\beta \notin B$, and we know that $m_i(r',s')$ is a mosaic with a type 3 defect $\neg\beta$.

Thus all the games in $G^i_{(r',s')}$ will continue to the $(i+1)$th round. The defect $\neg\beta$ can not be cured at this stage as we know $\beta \in \lambda(t)$ for any t between r and s. But E must have played a full decomposition (as part of the full expansion) and we have a contradiction.

($\Leftarrow$) Suppose $U(\alpha,\beta) \in \lambda(r)$.

Find i with $r \in Q_i$ and some adjacent pair $r' < s'$ in Q_i with $r \leq r'$ and at least one of the following properties of $(A',B',C') = m_i(r',s')$ holding:

1. β and $U(\alpha,\beta)$ are not both in B',
2. β and $U(\alpha,\beta)$ are not both in C',
3. $\neg\alpha$ is not in B',
4. or α is in C'.

To see that there is such an i, r' and s' suppose not for contradiction. Choose any i. Thus $U(\alpha,\beta)$ holds in the end of

$m_i(r'', s'')$ where s'' is the last element of Q_i and r'' is the penultimate one. Thus $m_i(r'', s'')$ has a defect and each game in $G^i_{(r'',s'')}$ continues. Look at the full right expansion of $m_i(r'', s'')$ played by E in these games. There will be a cure for the $U(\alpha, \beta)$ defect played. Thus in Q_{i+1} there is some $r' < s'$ with $m_{i+1}(r', s') = (A', B', C')$ with $\alpha \in C'$. This is our contradiction.

Assume we have the first such $r' < s'$ in some Q_i. Note that $U(\alpha, \beta) \in \lambda(r')$ as either $r = r'$ or, if $r < r'$, then we could replace r' by its predecessor (and have property 2 hold).

We are looking for some $s > r$ from T such that $\alpha \in \lambda(s)$ and for all $t \in T$ with $r < t < s$, $\beta \in \lambda(t)$. The inductive hypothesis will then give us the result. Furthermore, as we will see, thanks to the inheritance conditions, it will do if we find some $j \geq i$, some $s \in Q_j$, such that

1. $r' < s$,
2. $\alpha \in \lambda(s)$,
3. for each adjacent $v < w$ from Q_j, if $r' \leq v < w \leq s$, we have β in the cover of $m_j(v, w)$, and
4. for each adjacent $v < w$ from Q_j, if $r' < v < w \leq s$, we have β in the start of $m_j(v, w)$.

Either we are done already if $\alpha \in \lambda(s')$ and β is in the cover of $m_i(r', s')$, or we can show that $m_i(r', s') = (A', B', C')$ has what we will call a *useful* defect:

1. $U(\alpha, \beta) \in \lambda(r') = A'$ and $\beta \notin B'$ so we have a type 1 defect; or

2. $\neg\alpha \notin B'$ and we have a type 3 defect.

To see this suppose (A', B', C') does not have such a useful defect. Thus we have $\beta \in B'$ and $\neg\alpha \in B'$. We already know that $U(\alpha, \beta) \in \lambda(r') = A'$ and we have assumed $\alpha \notin \lambda(s') = C'$ (as we noted that we would have already found $s = s'$ if $\alpha \in \lambda(s')$ and $\beta \in B'$). By coherency 3, $U(\alpha, \beta) \in B'$ and by coherency 2, both $\beta \in C'$ and $U(\alpha, \beta) \in C'$. This all contradicts the definition of r', s'.

The existence of the useful defect in $m_i(r', s')$ shows that the games in $G^i_{(r',s')}$ continue for another round at least. E's play gives us a full decomposition for $m_i(r', s')$ which is allocated to some $r' < r_1 < r_2 < ... < r_{n-1} < s'$ in round $i + 1$ of the construction. The witness $s = r_a$ to the cure of the useful defect in this decomposition is the s we are looking for.

This s will do as $\alpha \in \lambda(s)$, β is in the cover of all mosaics between r and r' and in starts or ends of mosaics strictly between in the ith round, and β is in the cover of all mosaics between

> r' and s and in starts or ends of mosaics strictly between in the $(i+1)$th round (if we had needed to go on for this next round to find s). By inheritance we are done. □

4.1 The length of full decompositions

In proposing procedures for checking for full decompositions it is useful to know how many mosaics might be needed. We can easily find an exponential upper bound.

LEMMA 23 *Suppose ϕ is a formula of $L(U,S)$ of length N and m is a ϕ-mosaic.*

If m has a full decomposition $\langle m_1, m_2, ..., m_K \rangle$ then it has a full decomposition of length $\leq 2^{9N}$ using only the m_i.

> PROOF: First note that there are at most $2N$ formulas in the closure of ϕ and so there are at most 2^{6N} different ϕ-mosaics.
>
> Suppose that m has a full decomposition $m_1, m_2, ..., m_K$. We show that we can throw away some of these if $K > 2^{9N}$.
>
> For each internal defect in m, of which there are at most $4N$, choose a mosaic m_i from the full decomposition which witnesses the cure and call that mosaic *important*.
>
> I claim that if there are any mosaics repeated in between successive important mosaics in the decomposition then we can chop out one copy of the repeated mosaic and all the mosaics in between the two copies and still be left with a full decomposition. It is easy to check that the shorter sequence still composes and that it still contains cures for the defects in m.
>
> Continuing to chop in this way leaves us with at most 2^{6N} mosaics in between important mosaics. Thus we have a full decomposition of length at most $4N.(2^{6N}+1) \leq 2^{9N}$. □

There are similar results for left and right expansions which are proved in an analagous way.

LEMMA 24 *Suppose ϕ is a formula of $L(U,S)$ of length N and m is a ϕ-mosaic.*

If m has a full right (resp. left) expansion $\langle m_1, m_2, ..., m_K \rangle$ then it has a full right (resp. left) expansion of length $\leq 2^{9N}$ using only the m_i.

4.2 Using games to decide $L(U,S)$

We finish the section by summarising the relationship between the mosaic games and our decision problem. We also show that this relationship gives us a powerful tool for reasoning with linear time (and related logics): we can easily get a decision procedure albeit not the most efficient.

Putting together lemmas 21 and 20 we have:

THEOREM 25 *Suppose that ϕ is a formula of $L(U, S)$. Then ϕ-mosaic m is satisfiable iff E has a winning strategy in the game for m.*

This relationship and the simple form of the mosaic game immediately give us a decision procedure. The general idea comes from [Pra79].

Suppose that we want to decide whether ϕ is satisfiable. Look for a ϕ-mosaic m such that E has a winning strategy in the game for m. If there is such an m, it can be found by considering the set of all ϕ-mosaics at once.

Say that the length of ϕ is N so that the length of $\mathrm{Cl}(\phi)$ is at most $2N$. There are only 2^{6N} sets of formulas from $\mathrm{Cl}(\phi)$ so there are at most 2^{6N} ϕ-mosaics.

Start with the set S containing all the ϕ-mosaics. In the procedure, we loop around doing some checks and removing mosaics from S.

Each loop we check whether there is any ϕ-mosaic at all inside S. If there is not then we can stop and conclude that ϕ is unsatisfiable.

If S does contain a ϕ-mosaic then we continue. We go through each mosaic m in S and check whether m has a full expansion into mosaics from S. As we will see, in the context of this procedure, S is closed under composition, and so this only requires checking for decompositions of length $13N$ because there are only $12N$ possible internal defects in each mosaic. If m has no such full decomposition then mark it for removal. Similarly for external defects.

After checking all mosaics in m then remove all the marked ones. If any mosaics are removed then repeat the process. Otherwise we can stop and conclude that ϕ is satisfiable.

The correctness of the procedure follows easily from the theorem above. It is clear that S is indeed always closed under composition because concatenation of full decompositions/left expansions/right expansions gives us a full decomposition/left expansion/right expansion of a composition.

It is straightforward to check that the procedure terminates in exponential time and we have:

THEOREM 26 *LUS-SAT is in EXPTIME.*

It should be noted that [Rey10a] shows that LUS-SAT is actually in PSPACE. The proof there is by translating LUS-SAT into a satisfiability query in terms of sub-orders of the real-line and then using the PSPACE decision procedure for RTL from [Rey10b]. That result in turn involves a tree of mosaics and their decompositions. By establishining limits on the depth of the tree (a polynomial in terms of the length of the formula) and on the branching factor (exponential) we can ensure that we have a PSPACE algorithm as we only need to remember a small fixed amount of information about all the previous siblings of a given node.

5 Tableaux

Most of the work on temporal tableaux involves a move away from the traditional tree-shaped tableau building process of other modal logics. The standard approach for temporal logics is, as we have done already in this paper, to start with a graph and repeatedly prune away nodes, according to certain removal rules, until there is nothing more to remove (success) or some failure condition is detected. This approach is seen for the linear PLTL in [Wol85] and [Gou89] and for the simple branching CTL in [EC82] and [EH85]. The PLTL tableau in [Sch98] and the CTL* tableau in [Rey09b] are interesting because of the return to a tree shape.

We also want to use a tree tableau approach to decide validity of a formula in US/LIN. We will start with a formula ϕ and determine whether ϕ is satisfiable in US/LIN or not. To decide validity simply determine satisfiabilty of the negation.

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 winning strategies in the mosaic games.

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 e.t.c. 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 27

1. A tree *here is just a set (of* nodes*), with a binary* 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 partial map from nodes of a tree to ϕ-mosaics. The map is partial as, in certain circumstances, we will allow some leaf nodes to be un-labelled, or null-labelled.

In this paper trees will be drawn with the root at the top and children below parents joined by lines. The left to right ordering will be just that across the page.

The idea, as we will see, is that generally the labels of the children of a node form a full decomposition for the label of the node. However, we have to further elaborate our trees to provide for left and right expansions. As we will show later, this is only needed for the nodes on the left-most or right-most branches. The left-most branch is just that consisting of the root, its left-most child, and that nodes left-most child, etc until (if) we reach a leaf. Thus we require any non-leaf node on the left-most branch to have one or more of its left-most children being denoted as *left-expanding* children. Simarly on the right-most branch. In our diagrams we indicate these by putting $-$ or $+$ (for left and right, respectively) on the lines to these children.

DEFINITION 28 *A (non-singleton)* tableau *(for ϕ-mosaic m) is a ϕ-mosaic labelled ordered tree with (0) root labelled by m; (1) each node having a non-null label apart from possibly the leaf-node on the left-most branch and the leaf-node on the right-most branch; (2) each non-leaf node on the left-most branch (and no other nodes) having one or more of its left-most children being left-expanding children; (3) similarly right; (4a) the labels of the left-expanding children of any node (which has such) taken in order forming a full left-expansion of the label on the node; (4b) a node which has one left-expanding child with a null label must have a mosaic label with no left defects; (5) similarly right; (6) the labels on the non-expanding children of any non-leaf node taken in order forming a full decomposition of the label on the node.*

DEFINITION 29 *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, a clone leaf node or a null-labelled leaf.*

Define a successful *tableau as one in which all nodes are complete (otherwise the tableau is incomplete).*

5.1 Examples

Here we provide some short examples of mosaics and tableaux. Diagrams show tableaux as trees growing down from the root. Left to right ordering of child nodes is across the page. Left-expanding children are indicated by $-$, right by $+$. Successful branches are indicated by $\surd$.

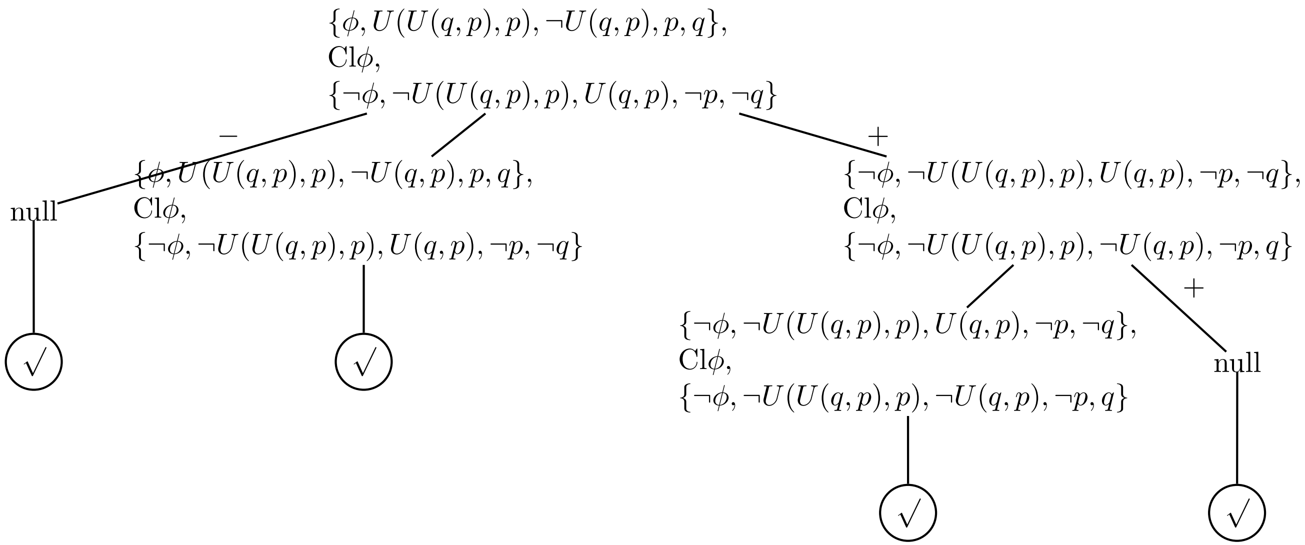
$\{\phi, U(U(q,p),p), \neg U(q,p), p, q\}$,
Clϕ,
$\{\neg\phi, \neg U(U(q,p),p), U(q,p), \neg p, \neg q\}$
$-$
$+$
null
$\{\phi, U(U(q,p),p), \neg U(q,p), p, q\}$,
Clϕ,
$\{\neg\phi, \neg U(U(q,p),p), U(q,p), \neg p, \neg q\}$
$\{\neg\phi, \neg U(U(q,p),p), U(q,p), \neg p, \neg q\}$,
Clϕ,
$\{\neg\phi, \neg U(U(q,p),p), \neg U(q,p), \neg p, q\}$
$+$
$\{\neg\phi, \neg U(U(q,p),p), U(q,p), \neg p, \neg q\}$,
Clϕ,
$\{\neg\phi, \neg U(U(q,p),p), \neg U(q,p), \neg p, q\}$
null
$\surd$
$\surd$
$\surd$
$\surd$

Figure 1: Tableau for $\phi = U(U(q,p),p) \wedge \neg U(q,p)$

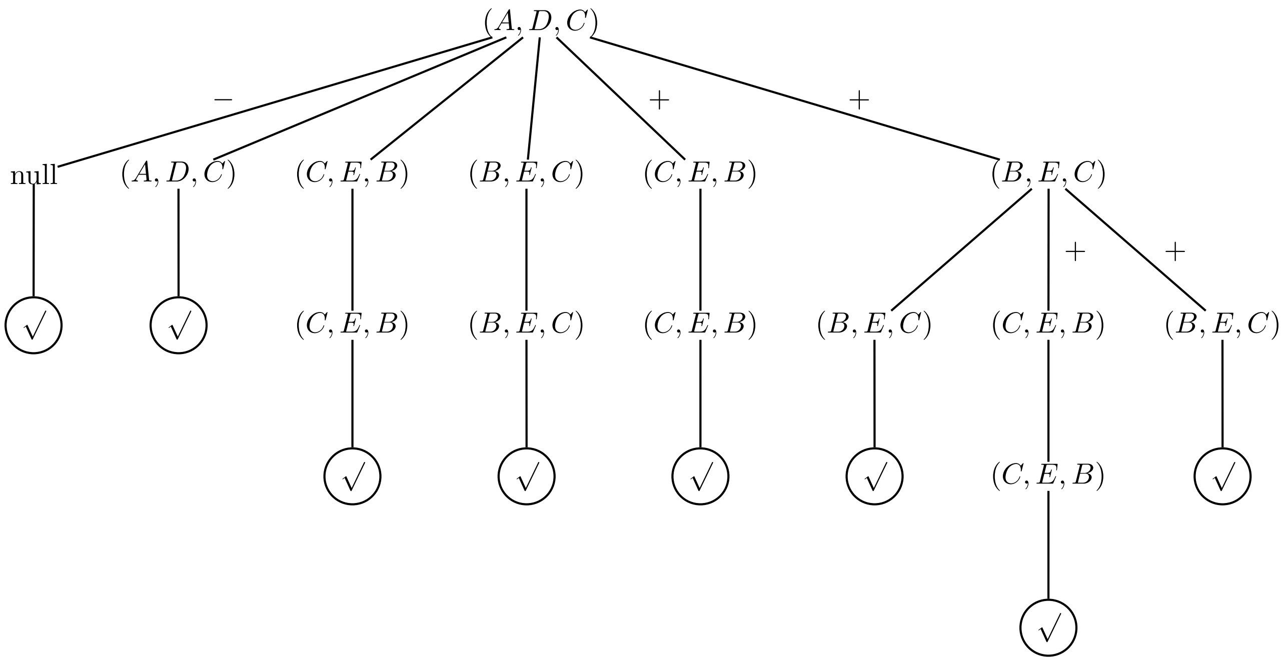

Figure 2: Tableau for $\phi = \neg U(\neg p, p) \wedge \neg U(p, \neg p)$: the sets named in the tableau are defined in Figure 3.

$$\begin{array}{rl} A = & \{\phi, \neg U(\neg p, p), \neg U(p, \neg p), p\} \\ B = & \{\neg\phi, \neg U(\neg p, p), U(p, \neg p), \neg p\} \\ C = & \{\neg\phi, U(\neg p, p), \neg U(p, \neg p), p\} \\ D = & \{\neg\phi\} \\ E = & \mathrm{Cl}\phi \end{array}$$

Figure 3: Sets seen in the tableau for $\phi = \neg U(\neg p, p) \wedge \neg U(p, \neg p)$ in Figure 2

See the example in Figure 1 for a successful tableau for $\phi = U(U(q, p), p) \wedge \neg U(q, p)$. Here the second and third successful branches terminate in clones.

See the example in Figure 2 for a successful tableau for $\phi = \neg U(\neg p, p) \wedge \neg U(p, \neg p)$. The sets named in the tableau are defined in Figure 3.

Example 3: an unsatisfiable mosaic:

$$(\{\neg U(\neg p, q), p, q\}, \{\neg p, q, \neg U(\neg p, q)\}, \{p, q, \neg U(\neg p, q)\}).$$

As p is not in the cover, it is a type 3 defect. However, any cure will not be consistent with $\neg U(\neg p, q)$ holding at the start.

Example 4: a perfect mosaic for $U(p, q)$ is

$$(\{p, q, U(p, q)\}, \{p, \neg p, q, \neg q, U(p, q), \neg U(p, q)\}, \{p, q, \neg U(p, q)\}).$$

5.2 Sound and Complete

In this section we show that the US/LIN tableau approach is sound and complete for deciding satisfiability of mosaics.

We will see that we can use tableaux to organise and present strategies for E in mosaic games. The labels on the children of a node provide a possible play for E if A chooses the label on the node. A play of a game can continue in such a way down a branch. The non-expanding children of the node give a full expansion directly. We will see that we can also find left and right expansions from the labels. The converse is immediate: strategies give tableaux.

LEMMA 30 *A ϕ-mosaic m has a successful tableau iff E has a winning strategy for the game on m.*

> PROOF: We first show that if a ϕ-mosaic m has a complete tableau then E has a winning strategy in the game on m.
>
> First note that for every mosaic appearing as a label in the tree we can find directly from the tree a full decomposition consisting only of mosaics which also appear as labels. If a label appears on a clone node then it will also appear on an ancestor

node which does sport a full decomposition. Thus E simply uses those to determine the full decomposition part of her plays.

The slightly hard part is to see that we can also extract full left (resp. right) expansions for any mosaic in the tree also from the tree. We will examine the full left expansion as the full right expansion argument is mirror.

Suppose that mosaic m' appears as the label of a node in the tree. Choose any node x which is labelled with m'. Let x_0 be the most recent ancestor of x which is on the left-most branch of the tree. So x_0 may be the root. Let $\langle x_0, x_1, x_2, ..., x_n \rangle$ be the sequence of ancestors of x in order from x_0 down to $x = x_n$.

Let π be a possibly empty full left expansion of the label of x_0. This can be read directly as the labels of the left expanding children of x_0.

For each $i = 0, 1, 2, ..., n-1$ do the following. Suppose that the non-expanding children of x_i are in left to right order $\langle y_1, y_2, ..., y_u, x_{i+1}, z_1, z_2, ..., z_v \rangle$. Let τ_i be the possibly empty sequence of mosaic labels of just the $y_1, y_2, ..., y_u$ in order. Note that thus this is a composing sequence of mosaics. If it is empty, i.e. if $u = 0$, then the start of the label of x_{i+1} equals the start of the label of x_i. Otherwise, the start of the first mosaic in τ_i is the same as the start of x_i and the end of the last mosaic in τ_i is the same as the start of x_{i+1}.

Let ρ be $\tau_0 \, ^\wedge \tau_1 \, ^\wedge ... \, ^\wedge \tau_{n-1}$. This is actually a composing sequence of mosaics: this follows from the matching of starts and ends with the labels of the x_i as just observed.

Note also that the start of the first mosaic in ρ is the start of the label of x_0 and the end of the last mosaic in the sequence ρ is the same as the start of m'.

Let σ consist of a concatenation of full decompositions of the respective mosaics in ρ in order. We know that we can find full decompositions for each of the mosaics in ρ from the tableau. This is also a composing sequence of mosaics as the starts and ends match.

We claim that $\pi \, ^\wedge \sigma$ is a full left expansion of m'.

To see this consider a formula $S(\alpha, \beta)$ in the start of m'. Note that this formula also appears in the end of the last mosaic in ρ as that equals the start of m'.

Now find the largest index i such that ρ_i either has a start containing α or a cover not containing $\neg\alpha$.

If there is no such i then $\neg\alpha$ is in every start and $\neg\alpha$ is in every cover of every mosaic in ρ. A simple induction using the coherency of the mosaics in ρ (condition C4) and the fact that $S(\alpha, \beta)$ will be in the end thus shows us that β is also in every start and every cover.

Also $S(\alpha, \beta)$ will be in the start of ρ_0 (and all the other starts) which is the start of the label of x_0. Thus one can find a witness to it from the left expansion π (then continuing on through σ).

Now consider the case when ρ_i starts with α or has a type 3 defect of α. The full decomposition of ρ_i in σ will give us our witness.

That completes the forward part.

Conversely, if E has a winning strategy (without loss deterministic) in the game on m, then writing out her plays in all possible games on m, gives us a successful tableau. Note that if a game finishes with a mosaic with no internal defects then, in the tableau, it may be necessary to add a clone node to that node to complete the branch. □

5.3 Using tableaux to decide US/LIN

We have seen that tableaux are useful for checking satisfiability of mosaics. To decide satisfiability of US/LIN formulas is closely related. It is also easy, after a few preliminary definitions.

One complication is that some formulas such as $\neg U(\top, \top) \wedge \neg S(\top, \top)$ are satisfiable but only have one point models. To handle them we can not use mosaics (at least as we have defined them). Instead, we simply invent a special basic one set tableau. This is just an MPC set of subformulas of ϕ with the proviso that it contains no until formulas no sinces.

DEFINITION 31 *A singleton tableau for a formula ϕ is a separate special structure just consisting of an MPC subset of the closure of ϕ which contains no formulas of the form $U(\alpha, \beta)$ or $S(\alpha, \beta)$.*

Thus, if $U(\alpha, \beta)$ is in the closure set then $\neg U(\alpha, \beta)$ must be in the tableau set. Such a tableau can be used for formulas that have one point models (a special case of a linear model).

LEMMA 32 *ϕ has a one point model iff ϕ has a successful singleton tableau.*

PROOF: The tableau D is related to the one point model M by $D = \{\alpha \in \mathrm{Cl}\phi | M, 0 \models \phi\}$ where 0 is the one element of the domain of M.

□

This allows us to now prove our main result.

THEOREM 33 *ϕ is satisfiable iff ϕ has a successful tableau.*

PROOF: ($\Rightarrow$) Suppose ϕ is satisfiable.

If ϕ has a one-point model then lemma 32 tells us that ϕ has a successful (singleton) tableau.

Otherwise, there is a non-singleton linear order $(T, <)$ with valuation g for the atoms and $x \in T$ such that $(T, <, g), x \models \phi$. In this case choose some other $y \in T$. Assume that $x < y$: the other case is similar.

Thus $m = mos_{\phi}^{(T,<,g)}(x, y)$ is satisfiable and is a mosaic for ϕ.

By Theorem 25, E has a winning strategy in the game on m. By lemma 30, m has a successful tableau.

($\Leftarrow$)

If ϕ has a singleton tableau then use lemma 32.

Otherwise, ϕ appears in the start or end of a mosaic m and m has a successful tableau. So, by lemma 30, E has a winning strategy in the game for m. Thus, by Theorem 25, m is satisfiable and clearly any model of m is a model of ϕ.

□

6 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 an exponential bound on the branching factor (lemmas 23 and 24) 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.

LEMMA 34 *There is a finite bound on the number of tableaux for ϕ, triple exponential in the length of ϕ.*

The complexity of reasoning using such tableaux is thus triple exponential time.

A simple Java implementation shows 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. For example, the formula $\phi = U(p, q)$ of length 3 has $2,304$ different mosaics to consider; $U(U(p, q), q)$ of length 5 has $22,848$ different mosaics; and

$U(U(p,q),q) \wedge \neg U(p,q)$ of length 11 has 228,864 mosaics. Clearly, more intelligent techniques are needed to make practical use of this basic framework.

7 Conclusion and Future Work

We have traveled via mosaics and games to provide a sound and complete tableau reasoning system for the temporal logic of until and since over general linear time, a logic which underlies a full range of logics of linear time. The syntactic checks involved in relating the labels of nodes in the tableau tree are quite straightforward.

However, there are blow-ups in the number of mosaics needing to considered which render any unsophisticated direct use of this technique impractical. Instead, it is hoped that it can form the foundation for more intelligent tableau construction techniques.

Particular directions to pursue in future work include finding good heuristics for the choice of mosaics to expand nodes. There is also potential for being clever in the construction of full expansions: for example, proceeding one defect at a time, to split a mosaic into two.

Also, it would be useful to develop a notion of partial mosaics to label nodes, whereby some formulas are specified to be in the start, end or cover, some specified to be out and others left undecided.

The other direction to work towards is modifying the tableau to work for closely related logics such as, most importantly, RTL, the logic of U and S over the reals. As shown in recent work [Rey10c] this also leads us on to a practical metric temporal logic. Also, as shown in [Rey10a], an RTL tableau can help with deciding many other linear temporal logics.

References

[AH93] Rajeev Alur and Thomas A. Henzinger. Real-time logics: Complexity and expressiveness. *Inf. Comput.*, 104(1):35–77, 1993.

[BG85] J. P. Burgess and Y. Gurevich. The decision problem for linear temporal logic. *Notre Dame J. Formal Logic*, 26(2):115–128, 1985.

[Bur82] J. P. Burgess. Axioms for tense logic I: “Since” and “Until”. *Notre Dame J. Formal Logic*, 23(2):367–374, 1982.

[CHR91] Zhou Chaochen, C. A. R. Hoare, and Anders P. Ravn. A calculus of durations. *Information Processing Letters*, 40(5):269–276, 1991.

[EC82] E. Emerson and E. C. Clarke. Using branching time temporal logic to synthesise synchronisation skeletons. *Sci. of Computer Programming*, 2, 1982.

[EH85] 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.

[Ehr61] A. Ehrenfeucht. An application of games to the completeness problem for formalized theories. *Fund. Math.*, 49:128–141, 1961.

[Fit83] M. Fitting. *Proof methods for modal and intuitionistic logics.* Reidel, 1983.

[Fra54] Roland Fraïssé. Sur quelques classifications des systèmes de relations, thesis, Paris, 1953. *Publications Scientifiques de l'Université d'Alger*, series A(1):35–182, 1954.

[GHR93] D. M. Gabbay, I. M. Hodkinson, and M. A. Reynolds. Temporal expressive completeness in the presence of gaps. In J. Oikkonen and J. Väänänen, editors, *Logic Colloquium '90, Proceedings ASL European Meeting 1990, Helsinki*, number 2 in Lecture Notes in Logic, pages 89–121. Springer-Verlag, 1993.

[GHR94] D. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic: Mathematical Foundations and Computational Aspects, Volume 1.* Oxford University Press, 1994.

[Gor99] R. Goré. Tableau methods for modal and temporal logics. In M. D'Agostino, D. Gabbay, R. Hähnle, and J. Posegga, editors, *Handbook of Tableau Methods*, pages 297–396. Kluwer Academic Publishers, 1999.

[Gou89] G. Gough. Decision procedures for temporal logics. Technical Report UMCS-89-10-1, Department of Computer Science, University of Manchester, 1989.

[Gur64] Y. Gurevich. Elementary properties of ordered abelian groups. *Algebra and Logic*, 3:5–39, 1964. (Russian; an English version is in Trans. Amer. Math. Soc. 46 (1965), 165–192).

[HC68] G. Hughes and M. Cresswell. *An Introduction to Modal Logic.* Methuen, London, 1968.

[HH02] R. Hirsch and I. Hodkinson. *Relation algebras by games.* Studies in Logic and the Foundations of Mathematics, Volume 147. North-Holland, 2002.

[HHM+99] Robin Hirsch, Ian Hodkinson, Maarten Marx, Szabolcs Mikulas, and Mark Reynolds. Mosaics and step-by-step - remarks on 'a modal logic of relations' by venema marx. In *Logic at Work. Essays Dedicated to the Memory of Helena Rasiowa, volume 24 of Studies in Fuzziness and Soft Computing*, pages 158–167. Springer-Verlag, 1999.

[HS86] J. Halpern and Y. Shoham. A propositional modal logic of time intervals. In *Proceedings, Symposium on Logic in Computer Science*. IEEE, Boston, 1986.

[Kam68] H. Kamp. *Tense logic and the theory of linear order*. PhD thesis, University of California, Los Angeles, 1968.

[KMP94] Y. Kesten, Z. Manna, and A. Pnueli. Temporal verification of simulation and refinement. In *A decade of concurrency: reflections and perspectives: REX school/symposium, Noordwijkerhout, the Netherlands, June 1–4, 1993*, pages 273–346. Springer–Verlag, 1994.

[LS02] Martin Lange and Colin Stirling. Model checking games for branching time logics. *J. Log. Comput.*, 12(4):623–639, 2002.

[MMR00] M. Marx, S. Mikulas, and M. Reynolds. The mosaic method for temporal logics. In R. Dyckhoff, editor, *Automated Reasoning with Analytic Tableaux and Related Methods, Proceedings of International Conference, TABLEAUX 2000, Saint Andrews, Scotland, July 2000*, LNAI 1847, pages 324–340. Springer, 2000.

[Nem95] I. Németi. Decidable versions of first order logic and cylindric-relativized set algebras. In L. Csirmaz, D. Gabbay, and M. de Rijke, editors, *Logic Colloquium '92*, pages 171–241. CSLI Publications, 1995.

[Pnu77] A. Pnueli. The temporal logic of programs. In *Proceedings of the Eighteenth Symposium on Foundations of Computer Science*, pages 46–57, 1977. Providence, RI.

[Pra79] V. R. Pratt. Models of program logics. In *Proc. 20th IEEE. Symposium on Foundations of Computer Science, San Juan*, pages 115–122, 1979.

[Rab98] A. Rabinovich. On the decidability of continuous time specification formalisms. *Journal of Logic and Computation*, 8:669–678, 1998.

[RD05] M. Reynolds and C. Dixon. Theorem-proving for discrete temporal logic. In M. Fisher, D. Gabbay, and L. Vila, editors, *Handbook of Temporal Reasoning in Artificial Intelligence*, pages 279–314. Elsevier, 2005.

[Rey03] Mark Reynolds. The complexity of the temporal logic with "until" over general linear time. *J. Comput. Syst. Sci.*, 66(2):393–426, 2003.

[Rey09a] Mark Reynolds. Dense time reasoning via mosaics. In Carsten Lutz and Jean-François Raskin, editors, *TIME 2009*, pages 3–10. IEEE Computer Society, 2009.

[Rey09b] Mark Reynolds. A tableau for CTL*. In *FM 2009: Formal Methods, Second World Congress, Eindhoven, The Netherlands, November 2009. Proceedings*, volume 5850, pages 403 – 418. Springer LNCS, 2009.

[Rey10a] M. Reynolds. The complexity of decision problems for linear temporal logics. *Journal of Studies in Logic*, 3:19–50, 2010.

[Rey10b] M. Reynolds. The complexity of temporal logic over the reals. *Ann. Pure Appl. Logic*, 161(8):1063–1096, 2010.

[Rey10c] M. Reynolds. Metric temporal logic revisited, submitted 2010.

[SC85] A. Sistla and E. Clarke. Complexity of propositional linear temporal logics. *J. ACM*, 32:733–749, 1985.

[Sch98] Stefan Schwendimann. A new one-pass tableau calculus for PLTL. In Harrie C. M. de Swart, editor, *TABLEAUX*, volume 1397 of *Lecture Notes in Computer Science*, pages 277–292. Springer, 1998.

[Wol85] P. Wolper. The tableau method for temporal logic: an overview. *Logique et Analyse*, 28:110–111, June–Sept 1985.

[Xu88] Ming Xu. On some U, S-tense logics. *J. of Philosophical Logic*, 17:181–202, 1988.