Strings over intervals

Tim Fernando
Computer Science Department
Trinity College, Dublin 2
Ireland
Tim.Fernando@tcd.ie

Abstract

Intervals and the events that occur in them are encoded as strings, elaborating on a conception of events as "intervals cum description." Notions of satisfaction in interval temporal logics are formulated in terms of strings, and the possibility of computing these via finite-state machines/transducers is investigated. This opens up temporal semantics to finite-state methods, with entailments that are decidable insofar as these can be reduced to inclusions between regular languages.

1 Introduction

It is well-known that Kripke models for *Linear Termporal Logic* (LTL) can be formulated as strings (e.g. Emerson, 1990). For the purposes of natural language semantics, however, it has been argued since at least (Bennett and Partee, 1972) that intervals should replace points. It is less clear (than in the case of LTL) how to view models as strings for intervals drawn (say) from the real line $\mathbb{R}$, as in one of the more recent interval temporal logics proposed for English, the system $\mathcal{TPL}$ of (Pratt-Hartmann, 2005). But if we follow $\mathcal{TPL}$ in restricting our models to finite sets, we can encode satisfaction of a formula ψ in a set $\mathcal{L}(\psi)$ of strings $str(\mathcal{A}, I)$ representing models $\mathcal{A}$ and intervals I

$$(\dagger) \qquad \mathcal{A} \models_I \psi \iff str(\mathcal{A}, I) \in \mathcal{L}(\psi) .$$

The present paper shows how to devise encodings $str(\mathcal{A}, I)$ and $\mathcal{L}(\psi)$ that establish (†) in a way that opens temporal semantics up to finite-state methods (e.g. Beesley and Karttunen, 2003). Notice that the entailment from ψ to ψ' given by

$$(\forall \mathcal{A}, I) \quad \text{if } \mathcal{A} \models_I \psi \text{ then } \mathcal{A} \models_I \psi'$$

is equivalent, under (†), to the inclusion $\mathcal{L}(\psi) \subseteq \mathcal{L}(\psi')$. This inclusion is decidable provided $\mathcal{L}(\psi)$ and $\mathcal{L}(\psi')$ are regular languages. (The same cannot be said for context-free languages.)

1.1 $\mathcal{TPL}$-models and strings

We start with $\mathcal{TPL}$, a model in which is defined, relative to an infinite set E of *event-atoms*, to be a finite set $\mathcal{A}$ of pairs $\langle I, e \rangle$ of closed, bounded intervals $I \subseteq \mathbb{R}$ and event-atoms $e \in E$. (A closed, bounded interval in $\mathbb{R}$ has the form

$$[r_1, r_2] \quad \stackrel{\text{def}}{=} \quad \{r \in \mathbb{R} \mid r_1 \leq r \leq r_2\}$$

for some $r_1, r_2 \in \mathbb{R}$.) The idea is that $\langle I, e \rangle$ represents "an occurrence of an event of type e over the interval" I (Pratt-Hartmann, 2005; page 17). That is, we can think of $\mathcal{A}$ as a finite set of events, conceived as "intervals cum description" (van Benthem, 1983; page 113). Our goal below is to string out this conception beyond event-atoms, and consider relations between intervals other than sub-intervalhood (the focus of $\mathcal{TPL}$). To get some sense for what is involved, it is useful to pause for examples of the strings we have in mind.[1]

[1]Concrete English examples connected with text inference can be found in (Pratt-Hartmann, 2005; Pratt-Hartmann, 2005a), the latter of which isolates a fragment $\mathcal{TPL}^*$ of $\mathcal{TPL}$ related specifically to TimeML (Pustejovsky et al., 2003). The finite-state encoding below pays off in expanding the coverage

Proceedings of the TextInfer 2011 Workshop on Textual Entailment, EMNLP 2011, pages 50–58,
Edinburgh, Scotland, UK, July 30, 2011.

$$\rho_X(\alpha_1 \cdots \alpha_n) \stackrel{\text{def}}{=} (\alpha_1 \cap X) \cdots (\alpha_n \cap X)$$

$$bc(\mathsf{s}) \stackrel{\text{def}}{=} \begin{cases} bc(\alpha \mathsf{s}') & \text{if } \mathsf{s} = \alpha\alpha\mathsf{s}' \\ \alpha bc(\alpha'\mathsf{s}') & \text{if } \mathsf{s} = \alpha\alpha'\mathsf{s}' \text{ and } \alpha \neq \alpha' \\ \mathsf{s} & \text{otherwise} \end{cases}$$

Table 1: Two useful functions

Example A Given event-atoms e and e', let $\mathcal{A}$ be the $\mathcal{TPL}$-model $\{x_1, x_2, x_3\}$, where

$$x_1 \stackrel{\text{def}}{=} \langle [1,4], e \rangle$$
$$x_2 \stackrel{\text{def}}{=} \langle [3,9], e \rangle$$
$$x_3 \stackrel{\text{def}}{=} \langle [9,100], e' \rangle \ .$$

Over the alphabet *Pow*($\mathcal{A}$) of subsets of $\mathcal{A}$, let us represent $\mathcal{A}$ by the string

$$s(\mathcal{A}) \stackrel{\text{def}}{=} \boxed{x_1}\boxed{x_1, x_2}\boxed{x_2}\boxed{x_2, x_3}\boxed{x_3}$$

of length 5, each box representing a symbol (i.e. a subset of $\mathcal{A}$) and arranged in chronological order with time increasing from left to right much like a film/cartoon strip (Fernando, 2004). Precisely how $s(\mathcal{A})$ is constructed from $\mathcal{A}$ is explained in section 2. Lest we think that a box represents an indivisible instant of time, we turn quickly to

Example B The 12 months, January to December, in a year are represented by the string

$$s_{y/m} \stackrel{\text{def}}{=} \boxed{\text{Jan}}\boxed{\text{Feb}} \cdots \boxed{\text{Dec}}$$

of length 12, and the 365 days of a (common) year by the string

$$s_{y/m,d} \stackrel{\text{def}}{=} \boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}} \cdots \boxed{\text{Dec,d31}}$$

of length 365. These two strings are linked by two functions on strings: a function ρ_{months} that keeps only the months in a box so that

$$\rho_{months}(s_{y/m,d}) = \boxed{\text{Jan}}^{31}\boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}$$

and *block compression* bc, which compresses consecutive occurrences of a box into one, mapping $\rho_{months}(s_{y/m,d})$ to

$$bc(\boxed{\text{Jan}}^{31}\boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}) = s_{y/m} \ .$$

to examples discussed in (Fernando, 2011a) and papers cited therein. These matters are given short shrift below (due to space and time constraints); I hope to make amends at my talk in the workshop.

(A_1)	$x \bigcirc x$ (i.e. $\bigcirc$ is reflexive)
(A_2)	$x \bigcirc x' \implies x' \bigcirc x$
(A_3)	$x \prec x' \implies$ not $x \bigcirc x'$
(A_4)	$x \prec x' \bigcirc x'' \prec x''' \implies x \prec x'''$
(A_5)	$x \prec x'$ or $x \bigcirc x'$ or $x' \prec x$

Table 2: Axioms for event structures

That is,

$$bc(\rho_{months}(s_{y/m,d})) = s_{y/m}$$

where, as made precise in Table 1, ρ_X "sees only X" (equating *months* with {Jan, Feb, ... Dec} to make ρ_{months} an instance of ρ_X), while bc discards duplications, in accordance with the view that time passes only if there is change. Or rather: we observe time passing only if we observe a change in the contents of a box. The point of this example is that temporal granularity depends on the set X of what are observable — i.e., the *boxables* (we can put inside a box). That set X might be a $\mathcal{TPL}$-model $\mathcal{A}$ or more generally the set $\mathbf{E}$ of events in an *event structure* $\langle \mathbf{E}, \bigcirc, \prec \rangle$, as defined in (Kamp and Reyle, 1993).

Example C Given a $\mathcal{TPL}$-model $\mathcal{A}$, let $\bigcirc$ and $\prec$ be binary relations on $\mathcal{A}$ given by

$$\langle I, e \rangle \bigcirc \langle I', e' \rangle \stackrel{\text{def}}{\iff} I \cap I' \neq \emptyset$$
$$\langle I, e \rangle \prec \langle I', e' \rangle \stackrel{\text{def}}{\iff} (\forall r \in I)(\forall r' \in I')\ r < r'$$

for all $\langle I, e \rangle$ and $\langle I', e' \rangle \in \mathcal{A}$. Clearly, the triple $\langle \mathcal{A}, \bigcirc, \prec \rangle$ is an event structure — i.e., it satisfies axioms (A_1) to (A_5) in Table 2. But for finite $\mathcal{A}$, the temporal structure the real line $\mathbb{R}$ confers on $\mathcal{A}$ is reduced considerably by the Russell-Wiener-Kamp derivation of time from event structures (RWK). Indeed, for the particular $\mathcal{TPL}$-model $\mathcal{A}$ in Example A above, RWK yields exactly two temporal points, constituting the substring $\boxed{x_1, x_2}\boxed{x_2, x_3}$ of the string $s(\mathcal{A})$ of length 5. As an RWK-moment from an event structure $\langle \mathbf{E}, \bigcirc, \prec \rangle$ is required to be a $\subseteq$-maximal set of pairwise $\bigcirc$-overlapping events, RWK discards the three boxes $\boxed{x_1}$, $\boxed{x_2}$ and $\boxed{x_3}$ in $s(\mathcal{A})$. There is, however, a simple fix from (Fernando, 2011) that reconciles RWK not only with $s(\mathcal{A})$ but also with block compression bc: enlarge the set $\mathcal{A}$ of events/boxables to include *pre-* and *post-*

events, turning $s(\mathcal{A})$ into

$$\boxed{x_1, \mathit{pre}(x_2), \mathit{pre}(x_3)}\boxed{x_1, x_2, \mathit{pre}(x_3)}$$
$$\boxed{x_2, \mathit{post}(x_1), \mathit{pre}(x_3)}\boxed{x_2, x_3, \mathit{post}(x_1)}$$
$$\boxed{x_3, \mathit{post}(x_1), \mathit{post}(x_2)} \; .$$

Note that $\mathit{pre}(x_i)$ and $\mathit{post}(x_i)$ mark the past and future relative to x_i, injecting, in the terminology of (McTaggart, 1908), A-series ingredients for tense into the B-series relations $\prec$ and $\bigcirc$ (which is just $\prec$-incomparability). For our present purposes, these additional ingredients allow us to represent all 13 relations between intervals x and x' in (Allen, 1983) by event structures over $\{x, x', \mathit{pre}(x), \mathit{post}(x')\}$, including the sub-interval relation x *during* x' at the center of (Pratt-Hartmann, 2005),[2] which strings out to

$$\boxed{\mathit{pre}(x), x'}\boxed{x, x'}\boxed{\mathit{post}(x), x'}.$$

It will prove useful in our account of $\mathcal{TPL}$-formulas below to internalize the demarcation of x by $\mathit{pre}(x)$ and $\mathit{post}(x)$ when forming $\mathit{str}(\mathcal{A}, I)$.

1.2 Outline

The remainder of the paper is organized as follows. Section 2 fills in details left out in our presentation of examples above, supplying the ingredient $\mathit{str}(\mathcal{A}, I)$ in the equivalence

$$(\dagger) \qquad \mathcal{A} \models_I \psi \iff \mathit{str}(\mathcal{A}, I) \in \mathcal{L}(\psi) \; .$$

The equivalence itself is not established before section 3, where every $\mathcal{TPL}$-formula ψ is mapped to a language $\mathcal{L}(\psi)$ via a translation ψ_+ of ψ to a minor variant $\mathcal{TPL}_+$ of $\mathcal{TPL}$. That variant is designed to smoothen the step in section 4 from $\mathcal{TPL}$ to other interval temporal logics which can be strung out similarly, and can, under natural assumptions, be made amenable to finite-state methods.

[2] Or to be more correct, the version of $\mathcal{TPL}$ in (Pratt-Hartmann, 2005a), as the strict subset relation $\subset$ between intervals assumed in the *Artificial Intelligence* article amounts to the disjunction of the Allen relations *during*, *starts* and *finishes*. For concreteness, we work with $\subset$ below; only minor changes are required to switch to *during*.

2 Strings encoding finite interval models

This section forms the string $\mathit{str}(\mathcal{A}, I)$ in three stages described by the equation

$$\mathit{str}(\mathcal{A}, I) \overset{\text{def}}{=} s(\mathcal{A}_I)^\bullet \; .$$

First, we combine $\mathcal{A}$ and I into the restriction $\mathcal{A}_I$ of $\mathcal{A}$ to pairs $\langle J, e\rangle$ such that J is a strict subset of I

$$\mathcal{A}_I \overset{\text{def}}{=} \{\langle J, e\rangle \in \mathcal{A} \mid J \subset I\}$$

Second, we systematize the construction of the string $s(\mathcal{A})$ in Example A. And third, we map a string s to a string $s^\bullet$ that internalizes the borders externally marked by the *pre*- and *post*-events described in Example C. The map $\mathcal{A} \mapsto s(\mathcal{A})$ is the business of §2.1, and $s \mapsto s^\bullet$ of §2.2. With an eye to interval temporal logics other than $\mathcal{TPL}$, we will consider the full set $\mathit{Ivl}(\mathbb{R})$ of (non-empty) intervals in $\mathbb{R}$

$$\mathit{Ivl}(\mathbb{R}) \overset{\text{def}}{=} \{a \subseteq \mathbb{R} \mid a \neq \emptyset \text{ and } (\forall x, y \in a) \; [x, y] \subseteq a\} \; ,$$

and write $]r_1, r_2[$ for the open interval

$$]r_1, r_2[\overset{\text{def}}{=} \{r \in \mathbb{R} \mid r_1 < r < r_2\}$$

where we allow $r_1 = -\infty$ for intervals unbounded to the left and $r_2 = +\infty$ for intervals unbounded to the right. The constructs $\pm\infty$ are convenient for associating *endpoints* with every interval I, whether or not I is bounded. For I bounded to the left and to the right, we refer to real numbers r and r' as I's endpoints provided $I \subseteq [r, r']$ and

$$[r, r'] \subseteq [r'', r'''] \qquad \text{for all } r'' \text{ and } r''' \text{ such that } I \subseteq [r'', r'''] \; .$$

We write $\mathit{Endpoints}(I)$ for the (non-empty) set consisting of I's endpoints (including possibly $\pm\infty$).

2.1 Order, box and compress

Given a finite subset $\mathcal{A} \subseteq \mathit{Ivl}(\mathbb{R}) \times E$, we collect all endpoints of intervals in $\mathcal{A}$ in the finite set

$$\mathit{Endpoints}(\mathcal{A}) \overset{\text{def}}{=} \bigcup_{\langle I, e\rangle \in \mathcal{A}} \mathit{Endpoints}(I)$$

and construct $s(\mathcal{A})$ in three steps.

Step 1 Order *Endpoints*$(\mathcal{A})$ into an increasing sequence

$$r_1 < r_2 < \cdots < r_n.$$

Step 2 Box the $\mathcal{A}$-events into the sequence of $2n-1$ intervals

$$\{r_1\},\ ]r_1, r_2[,\ \{r_2\},\ ]r_2, r_3[,\ \ldots \{r_n\}$$

(partitioning the closed interval $[r_1, r_n]$), forming the string

$$\alpha_1 \beta_1 \alpha_2 \beta_2 \cdots \alpha_n$$

(of length $2n-1$) where

$$\alpha_j \stackrel{\text{def}}{=} \{\langle i, e\rangle \in \mathcal{A} \mid r_j \in i\}$$
$$\beta_j \stackrel{\text{def}}{=} \{\langle i, e\rangle \in \mathcal{A} \mid]r_j, r_{j+1}[\subseteq i\}\ .$$

Step 3 Block-compress $\alpha_1 \beta_1 \alpha_2 \beta_2 \cdots \alpha_n$

$$s(\mathcal{A}) \stackrel{\text{def}}{=} bc(\alpha_1 \beta_1 \alpha_2 \beta_2 \cdots \alpha_n)\ .$$

For example, revisiting Example A, where $\mathcal{A}$ is $\{x_1, x_2, x_3\}$ and

$$x_1 \stackrel{\text{def}}{=} \langle [1, 4], e\rangle$$
$$x_2 \stackrel{\text{def}}{=} \langle [3, 9], e\rangle$$
$$x_3 \stackrel{\text{def}}{=} \langle [9, 100], e'\rangle$$

we have from Step 1, the 5 endpoints

$$\vec{r} = 1, 3, 4, 9, 100$$

and from Step 2, the 9 boxes

x_1	x_1	x_1, x_2	x_1, x_2	x_1, x_2	x_2	x_2, x_3	x_3	x_3

that block-compresses in Step 3 to the 5 boxes $s(\mathcal{A})$

x_1	x_1, x_2	x_2	x_2, x_3	x_3

.

Notice that if we turned the closed intervals in x_1 and x_3 to open intervals $]1, 4[$ and $]9, 100[$ respectively, then Step 2 gives

	x_1	x_1, x_2	x_1, x_2	x_2	x_2	x_2	x_3	

which block-compresses to the 6 boxes

	x_1	x_1, x_2	x_2	x_3	

.

2.2 Demarcated events

Block compression accounts for part of the Russell-Wiener-Kamp constuction of moments from an event structure (RWK). We can neutralize the requirement of $\subseteq$-maximality on RWK moments by adding $\mathit{pre}(x_i), \mathit{post}(x_i)$, turning, for instance, $s(\mathcal{A})$ for $\mathcal{A}$ given by Example A into

$x_1, \mathit{pre}(x_2), \mathit{pre}(x_3)$	$x_1, x_2, \mathit{pre}(x_3)$
$\mathit{post}(x_1), x_2, \mathit{pre}(x_3)$	$\mathit{post}(x_1), x_2, x_3$
$\mathit{post}(x_1), \mathit{post}(x_2), x_3$	

(which $\rho_{\mathcal{A}}$ maps back to $s(\mathcal{A})$). In general, we say a string $\alpha_1 \alpha_2 \cdots \alpha_n$ is $\mathcal{A}$-*delimited* if for all $x \in \mathcal{A}$ and integers i from 1 to n,

$$\mathit{pre}(x) \in \alpha_i \iff x \in (\bigcup_{j=i+1}^{n} \alpha_j) - \bigcup_{j=1}^{i} \alpha_j$$

and

$$\mathit{post}(x) \in \alpha_i \iff x \in (\bigcup_{j=1}^{i-1} \alpha_j) - \bigcup_{j=i}^{n} \alpha_j\ .$$

Clearly, for every string $s \in \mathit{Pow}(\mathcal{A})^*$, there is a unique $\mathcal{A}$-delimited string s' such that $\rho_{\mathcal{A}}(s') = s$. Let $s_{\pm}$ be that unique string.

Notice that $\mathit{pre}(x)$ and $\mathit{post}(x)$ explicitly mark the borders of x in $s_{\pm}$. For the application at hand to $\mathcal{TPL}$, it is useful to internalize the borders within x so that, for instance in Example A, $s(\mathcal{A})_{\pm}$ becomes

$x_1, \mathit{begin}\text{-}x_1$	$x_1, x_2, x_1\text{-}\mathit{end}, \mathit{begin}\text{-}x_2$

x_2	$x_2, x_3, x_2\text{-}\mathit{end}, \mathit{begin}\text{-}x_3$	$x_3, x_3\text{-}\mathit{end}$

(with $\mathit{pre}(x_i)$ shifted to the right as *begin*-x_i and $\mathit{post}(x_i)$ to the left as x_i-*end*). The general idea is that given a string $\alpha_1 \alpha_2 \cdots \alpha_n \in \mathit{Pow}(\mathcal{A})^n$ and $x \in \mathcal{A}$ that occurs at some α_i, we add *begin*-x to the first box in which x appears, and x-*end* to the last box in which x appears. Or economizing a bit by picking out the first component I in a pair $\langle I, e\rangle \in \mathcal{A}$, we form the *demarcation* $(\alpha_1 \alpha_2 \cdots \alpha_n)^\bullet$ of $\alpha_1 \alpha_2 \cdots \alpha_n$ by adding *bgn-I* to α_i precisely if

> there is some e such that $\langle I, e\rangle \in \alpha_i$ and either $i = 1$ or $\langle I, e\rangle \notin \alpha_{i-1}$

$$\begin{array}{rcl} \varphi & ::= & \mathit{mult}(e) \mid \neg\varphi \mid \varphi \wedge \varphi' \mid \langle\beta\rangle\varphi \\ \alpha & ::= & e \mid e^f \mid e^l \\ \beta & ::= & \alpha \mid \alpha^< \mid \alpha^> \end{array}$$

Table 3: $\mathcal{TPL}_+$-formulas φ from extended labels β

and adding I-*end* to α_i precisely if

> there is some e such that $\langle I, e\rangle \in \alpha_i$ and either $i = n$ or $\langle I, e\rangle \notin \alpha_{i+1}$.

Returning to Example A, we have

$$s(\mathcal{A})^\bullet \;=\; \boxed{x_1, \mathit{bgn}\text{-}I_1}\boxed{x_1, x_2, I_1\text{-}\mathit{end}, \mathit{bgn}\text{-}I_2}\boxed{x_2}\boxed{x_2, x_3, I_2\text{-}\mathit{end}, \mathit{bgn}\text{-}I_3}\boxed{x_3, I_3\text{-}\mathit{end}}$$

which is $\mathit{str}(\mathcal{A}, I)$ for any interval I such that $[1, 100] \subset I$.

3 $\mathcal{TPL}$-satisfaction in terms of strings

This section defines the set $\mathcal{L}(\psi)$ of strings for the equivalence (†)

$$(\dagger) \qquad \mathcal{A} \models_I \psi \iff \mathit{str}(\mathcal{A}, I) \in \mathcal{L}(\psi)$$

by a translation to a language $\mathcal{TPL}_+$ that differs ever so slightly from $\mathcal{TPL}$ and its extension $\mathcal{TPL}^+$ in (Pratt-Hartmann, 2005). As in $\mathcal{TPL}$ and $\mathcal{TPL}^+$, formulas in $\mathcal{TPL}_+$ are closed under the modal operator $\langle e\rangle$, for every event-atom $e \in E$. Essentially, $\langle e\rangle\top$ says at least one e-transition is possible. In addition, $\mathcal{TPL}_+$ has a formula $\mathit{mult}(e)$ stating that multiple (at least two) e-transitions are possible. That is, $\mathit{mult}(e)$ amounts to the $\mathcal{TPL}^+$-formula

$$\langle e\rangle\top \wedge \neg\{e\}\top$$

where the $\mathcal{TPL}^+$-formula $\{e\}\psi$ can be rephrased as

$$\langle e\rangle\psi \wedge \neg\mathit{mult}(e)$$

(and $\top$ as the tautology $\neg(\mathit{mult}(e) \wedge \neg\mathit{mult}(e))$). More formally, $\mathcal{TPL}_+$-formulas φ are generated according to Table 3 without any explicit mention of the $\mathcal{TPL}$-constructs $\{\alpha\}$, $\{\alpha\}_<$ and $\{\alpha\}_>$. Instead, a $\mathcal{TPL}^+$-formula ψ is translated to a $\mathcal{TPL}_+$-formula ψ_+ so that (†) holds with $\mathcal{L}(\psi)$ equal to $\mathcal{T}(\psi_+)$, where $\mathcal{T}(\varphi)$ is a set of strings (defined below) characterizing satisfaction in $\mathcal{TPL}_+$. The translation ψ_+ commutes with the connectives common to $\mathcal{TPL}^+$ and $\mathcal{TPL}_+$

$$\text{e.g., } (\neg\psi)_+ \stackrel{\text{def}}{=} \neg(\psi_+)$$

and elsewhere,

$$\begin{array}{rcl} \top_+ & \stackrel{\text{def}}{=} & \neg(\mathit{mult}(e) \wedge \neg\mathit{mult}(e)) \\ (\{e\}\psi)_+ & \stackrel{\text{def}}{=} & \langle e\rangle\psi_+ \wedge \neg\mathit{mult}(e) \\ ([e]\psi)_+ & \stackrel{\text{def}}{=} & \neg\langle e\rangle\neg\psi_+ \\ (\{e\}_<\psi)_+ & \stackrel{\text{def}}{=} & \langle e^<\rangle\psi_+ \wedge \neg\mathit{mult}(e) \\ (\{e\}_>\psi)_+ & \stackrel{\text{def}}{=} & \langle e^>\rangle\psi_+ \wedge \neg\mathit{mult}(e) \end{array}$$

and as minimal-first and minimal-last subintervals are unique (Pratt-Hartmann, 2005, page 18),

$$\begin{array}{rcl} (\{e^g\}_<\psi)_+ & \stackrel{\text{def}}{=} & \langle e^{g<}\rangle\psi_+ \;\text{ for } g \in \{f, l\} \\ (\{e^g\}_>\psi)_+ & \stackrel{\text{def}}{=} & \langle e^{g>}\rangle\psi_+ \;\text{ for } g \in \{f, l\} \;. \end{array}$$

3.1 The alphabet $\Sigma = \Sigma_{\mathfrak{I},E}$ and its subscripts

The alphabet from which we form strings will depend on a choice $\mathfrak{I}, E$ of a set $\mathfrak{I} \subseteq \mathit{Ivl}(\mathbb{R})$ of real intervals, and a set E of event-atoms. Recalling that the demarcation $s(\mathcal{A})^\bullet$ of a string $s(\mathcal{A})$ contains occurrences of bgn-I and I-*end*, for each $I \in \mathit{domain}(\mathcal{A})$, let us associate with $\mathfrak{I}$ the set

$$\mathfrak{I}_\bullet \stackrel{\text{def}}{=} \{\mathit{bgn}\text{-}I \mid I \in \mathfrak{I}\} \cup \{I\text{-}\mathit{end} \mid I \in \mathfrak{I}\}$$

from which we build the alphabet

$$\Sigma_{\mathfrak{I},E} \stackrel{\text{def}}{=} \mathit{Pow}((\mathfrak{I} \times E) \cup \mathfrak{I}_\bullet)$$

so that a symbol (i.e., element of $\Sigma_{\mathfrak{I},E}$) is a set with elements of the form $\langle I, e\rangle$, bgn-I and I-*end*. Notice that

$$(\forall \mathcal{A} \subseteq \mathfrak{I} \times E) \qquad \mathit{str}(\mathcal{A}, I) \in \Sigma^*_{\mathfrak{I},E}$$

for any real interval I. To simplify notation, we will often drop the subscripts $\mathfrak{I}$ and E, restoring them when we have occasion to vary them. This applies not only to the alphabet $\Sigma = \Sigma_{\mathfrak{I},E}$ but also to the truth sets $\mathcal{T}(\psi) = \mathcal{T}_{\mathfrak{I},E}(\psi)$ below, with $\mathfrak{I}$ fixed in the case of (†) to the full set of closed, bounded real intervals.

3.2 The truth sets $\mathcal{T}(\varphi)$

We start with $mult(e)$, the truth set $\mathcal{T}(mult(e))$ for which consists of strings properly containing at least two e-events. We first clarify what "properly contain" means, before turning to "e-events." The notion of containment needed combines two ways a string can be part of another. The first involves deleting some (possibly null) prefix and suffix of a string. A *factor of* a string s is a string s' such that $s = us'v$ for some strings u and v, in which case we write s *fac* s'

$$s \; fac \; s' \quad \stackrel{\text{def}}{\iff} \quad (\exists u, v) \; s = us'v \; .$$

A factor of s is *proper* if it is distinct from s. That is, writing s *pfac* s' to mean s' is a proper factor of s,

$$s \; pfac \; s' \quad \iff \quad (\exists u, v) \; s = us'v \text{ and } uv \neq \epsilon$$

where ϵ is the null string. The relation *pfac* between strings corresponds roughly to that of proper inclusion $\supset$ between intervals.

The second notion of part between strings applies specifically to strings s and s' of sets: we say s *subsumes* s', and write $s \trianglerighteq s'$, if they are of the same length, and $\supseteq$ holds componentwise between them

$$\alpha_1 \cdots \alpha_n \trianglerighteq \alpha'_1 \cdots \alpha'_m \quad \stackrel{\text{def}}{\iff} \quad n = m \text{ and } \alpha'_i \subseteq \alpha_i \text{ for } 1 \leq i \leq n$$

(Fernando, 2004). Now, writing $R; R'$ for the *composition of* binary relations R and R' in which the output of R is fed as input to R'

$$s \; R; R' \; s' \quad \stackrel{\text{def}}{\iff} \quad (\exists s'') \; sRs'' \text{ and } s''R's' \; ,$$

we compose *fac* with $\trianglerighteq$ for *containment* $\sqsupseteq$

$$\sqsupseteq \quad \stackrel{\text{def}}{=} \quad fac \; ; \trianglerighteq \qquad (= \trianglerighteq \; ; fac)$$

and *pfac* with $\trianglerighteq$ for *proper containment* $\sqsupset$

$$\sqsupset \quad \stackrel{\text{def}}{=} \quad pfac \; ; \trianglerighteq \qquad (= \trianglerighteq \; ; pfac) \; .$$

Next, for e-events, given $I \in \mathfrak{I}$, let

$$\mathcal{D}(e, I) \quad \stackrel{\text{def}}{=} \quad \{s^{\bullet} \mid s \in \boxed{\langle I, e \rangle}^{+}\}$$

and summing over intervals $I \in \mathfrak{I}$,

$$\mathcal{D}_{\mathfrak{I}}(e) \quad \stackrel{\text{def}}{=} \quad \bigcup_{I \in \mathfrak{I}} \mathcal{D}(e, I) \; .$$

Dropping the subscripts on Σ and $\mathcal{D}(e)$, we put into $\mathcal{T}(mult(e))$ all strings in Σ^* properly containing more than one string in $\mathcal{D}(e)$

$$s \in \mathcal{T}(mult(e)) \quad \stackrel{\text{def}}{\iff} \quad (\exists s_1, s_2 \in \mathcal{D}(e)) \; s_1 \neq s_2 \text{ and } s \sqsupset s_1 \text{ and } s \sqsupset s_2.$$

Moving on, we interpret negation $\neg$ and conjunction $\wedge$ classically

$$\mathcal{T}(\neg\varphi) \quad \stackrel{\text{def}}{=} \quad \Sigma^* - \mathcal{T}(\varphi)$$
$$\mathcal{T}(\varphi \wedge \varphi') \quad \stackrel{\text{def}}{=} \quad \mathcal{T}(\varphi) \cap \mathcal{T}(\varphi')$$

and writing $R^{-1}L$ for $\{s \in \Sigma^* \mid (\exists s' \in L) \; sRs'\}$, we set

$$\mathcal{T}(\langle\beta\rangle\varphi) \quad \stackrel{\text{def}}{=} \quad \mathcal{R}(\beta)^{-1}\mathcal{T}(\varphi)$$

which brings us to the question of $\mathcal{R}(\beta)$.

3.3 The accessibility relations $\mathcal{R}(\beta)$

Having defined $\mathcal{T}(mult(e))$, we let $\mathcal{R}(e)$ be the restriction of proper containment $\sqsupset$ to $\mathcal{D}(e)$

$$s \; \mathcal{R}(e) \; s' \quad \stackrel{\text{def}}{\iff} \quad s \sqsupset s' \text{ and } s' \in \mathcal{D}(e) \; .$$

As for e^f and e^l, some preliminary notation is useful. Given a language L, let us collect strings that have at most one factor in L in $nmf(L)$ (for *n*on-*m*ultiple *f*actor)

$$nmf(L) \quad \stackrel{\text{def}}{=} \quad \{s \in \Sigma^* \mid \text{at most one factor of } s \text{ belongs to } L\}$$

and let us shorten $\trianglerighteq^{-1}L$ to $L^{\trianglerighteq}$

$$s \in L^{\trianglerighteq} \quad \stackrel{\text{def}}{\iff} \quad (\exists s' \in L) \; s \trianglerighteq s' \; .$$

Now,

$$s \; \mathcal{R}(e^f) \; s' \quad \stackrel{\text{def}}{\iff} \quad (\exists u, v) \; s = us'v \text{ and } uv \neq \epsilon \text{ and } s' \in \mathcal{D}(e)^{\trianglerighteq} \text{ and } us' \in nmf(\mathcal{D}(e)^{\trianglerighteq})$$

and similarly,

$$s\ \mathcal{R}(e^l)\ s' \quad \stackrel{\text{def}}{\iff} \quad (\exists u,v)\ s = us'v \\ \text{and } uv \neq \epsilon \\ \text{and } s' \in \mathcal{D}(e)^{\trianglerighteq} \\ \text{and } s'v \in \mathit{nmf}(\mathcal{D}(e)^{\trianglerighteq})\ .$$

Finally,

$$s\ \mathcal{R}(\alpha^<)\ s' \quad \stackrel{\text{def}}{\iff} \quad (\exists s'', s''')\ s = s's''s''' \\ \text{and } s\ \mathcal{R}(\alpha)\ s''$$

$$s\ \mathcal{R}(\alpha^>)\ s' \quad \stackrel{\text{def}}{\iff} \quad (\exists s'', s''')\ s = s'''s''s' \\ \text{and } s\ \mathcal{R}(\alpha)\ s''\ .$$

A routine induction on $\mathcal{TPL}^+$-formulas ψ establishes that for $\mathfrak{I}$ equal to the set $\mathcal{I}$ of all closed, bounded real intervals,

Proposition 1. *For all finite $\mathcal{A} \subseteq \mathcal{I} \times E$ and $I \in \mathcal{I}$,*

$$\mathcal{A} \models_I \psi \quad \iff \quad \mathit{str}(\mathcal{A}, I) \in \mathcal{T}_{\mathcal{I},E}(\psi_+)$$

for every $\mathcal{TPL}^+$-formula ψ.

3.4 $\mathcal{TPL}$-equivalence and $\mathfrak{I}$ revisited

When do two pairs $\mathcal{A}, I$ and $\mathcal{A}', I'$ of finite subsets $\mathcal{A}, \mathcal{A}'$ of $\mathcal{I} \times E$ and intervals $I, I' \in \mathcal{I}$ satisfy the same $\mathcal{TPL}$-formulas? A sufficient condition suggested by Proposition 1 is that $\mathit{str}(\mathcal{A}, I)$ is the same as $\mathit{str}(\mathcal{A}', I')$ up to renaming of intervals. More precisely, recalling that $\mathit{str}(\mathcal{A}, I) = s(\mathcal{A}_I)^\bullet$, let us define $\mathcal{A}$ to be *congruent with* $\mathcal{A}'$, $\mathcal{A} \cong \mathcal{A}'$, if there is a bijection between the intervals of $\mathcal{A}$ and $\mathcal{A}'$ that turns $s(\mathcal{A})$ into $s(\mathcal{A}')$

$$\mathcal{A} \cong \mathcal{A}' \quad \stackrel{\text{def}}{\iff} \quad (\exists f : \mathit{domain}(\mathcal{A}) \to \mathit{domain}(\mathcal{A}')) \\ f \text{ is a bijection, and} \\ \mathcal{A}' = \{\langle f(I), e\rangle \mid \langle I, e\rangle \in \mathcal{A}\} \\ \text{and } f[s(\mathcal{A})] = s(\mathcal{A}')$$

where for any string $s \in \mathit{Pow}(\mathit{domain}(f) \times E)^*$,

$$f[s] \quad \stackrel{\text{def}}{=} \quad s \text{ after renaming each} \\ I \in \mathit{domain}(f) \text{ to } f(I)\ .$$

As a corollary to Proposition 1, we have

Proposition 2. *For all finite subsets $\mathcal{A}$ and $\mathcal{A}'$ of $\mathcal{I} \times E$ and all $I, I' \in \mathcal{I}$, if $\mathcal{A}_I \cong \mathcal{A}'_{I'}$ then for every $\mathcal{TPL}^+$-formula ψ,*

$$\mathcal{A} \models_I \psi \quad \iff \quad \mathcal{A}' \models_{I'} \psi\ .$$

The significance of Proposition 2 is that it spells out the role the real line $\mathbb{R}$ plays in $\mathcal{TPL}$ — nothing apart from its contribution to the strings $s(\mathcal{A})$. Instead of picking out particular intervals over $\mathbb{R}$, it suffices to work with interval symbols, and to equate the subscript $\mathfrak{I}$ on our alphabet Σ and truth relations $\mathcal{T}(\psi)$ to say, the set $\mathbb{Z}_+$ of positive integers $1, 2, \ldots$. But lest we confuse $\mathcal{TPL}$ with Linear Temporal Logic, note that the usual order on $\mathbb{Z}_+$ does *not* shape the accessibility relations in $\mathcal{TPL}$. We use $\mathbb{Z}_+$ here only because it is big enough to include any finite subset $\mathcal{A}$ of $\mathcal{I} \times E$.

Turning to entailments, we can reduce entailments

$$\psi \vdash_{\mathcal{I},E} \psi' \quad \stackrel{\text{def}}{\iff} \quad (\forall \text{ finite } \mathcal{A} \subseteq \mathcal{I} \times E)(\forall I \in \mathcal{I}) \\ \mathcal{A} \models_I \psi \text{ implies } \mathcal{A} \models_I \psi'$$

to satisfiability as usual

$$\psi \vdash_{\mathcal{I},E} \psi' \quad \iff \quad \mathcal{T}_{\mathcal{I},E}(\psi \wedge \neg\psi') = \emptyset\ .$$

The basis of the decidability/complexity results in (Pratt-Hartmann, 2005) is a lemma (number 3 in page 20) that, for any $\mathcal{TPL}^+$-formula ψ, bounds the size of a minimal model of ψ. That is, as far as the satisfiability of a $\mathcal{TPL}^+$-formula ψ is concerned, we can reduce the subscript $\mathfrak{I}$ on $\mathcal{T}(\psi)$ to a finite set — or in the aforementioned reformulation, to a finite segment $\{1, 2, \ldots, n\}$ of $\mathbb{Z}_+$. We shall consider an even more drastic approach in the next section. For now, notice that the shift from the real line $\mathbb{R}$ towards strings conforms with

The Proposal of (Steedman, 2005)

> *the so-called temporal semantics of natural language is not primarily to do with time at all. Instead, the formal devices we need are those related to representation of causality and goal-directed action.* [p ix]

The idea is to move away from some absolute (independently given) notion of time (be they points or intervals) to the changes and forces that make natural language temporal.

4 The regularity of $\mathcal{TPL}$ and beyond

Having reformulated $\mathcal{TPL}$ in terms of strings, we proceed now to investigate the prospects for a finite-state approach to temporal semantics building on that reformulation. We start by bringing out the finite-state character of the connectives in $\mathcal{TPL}$ before considering some extensions.

4.1 $\mathcal{TPL}_+$-connectives are regular

It is well-known that the family of regular languages is closed under complementation and intersection — operations interpreting negation and conjunction, respectively. The point of this subsection is to show that all the $\mathcal{TPL}_+$-connectives map regular languages and regular relations to regular languages and regular relations. A relation is *regular* if it is computed by a finite-state transducer. If $\mathfrak{I}$ and E are both finite, then $\mathcal{D}_{\mathfrak{I},E}(e)$ is a regular language and $\sqsupset$ is a regular relation. Writing R_L for the relation $\{(s, s') \in R \mid s' \in L\}$, note that

$$\mathcal{R}(e) \quad = \quad \sqsupset_{\mathcal{D}(e)}$$

and that in general, if R and L are regular, then so is R_L.

Moving on, the set of strings with at least two factors belonging to L is

$$\begin{aligned} \textit{twice}(L) \quad \stackrel{\text{def}}{=} \quad & \Sigma^*(L\Sigma^* \cap (\Sigma^+ L\Sigma^*)) \; + \\ & \Sigma^*(L\Sigma^+ \cap L)\Sigma^* \end{aligned}$$

and the set of strings that have a proper factor belonging to L is

$$[L] \quad \stackrel{\text{def}}{=} \quad \Sigma^+ L\Sigma^* + \Sigma^* L\Sigma^+ \; .$$

It follows that we can capture the set of strings that properly contain at least two strings in L as

$$\textit{Mult}(L) \quad \stackrel{\text{def}}{=} \quad [\textit{twice}(L^{\trianglerighteq})] \; .$$

Note that

$$\mathcal{T}(\textit{mult}(e)) \quad = \quad \textit{Mult}(\mathcal{D}(e))$$

and recalling $\mathcal{R}(e^f)$ and $\mathcal{R}(e^l)$ use *nmf*,

$$\textit{nmf}(L) \quad = \quad \Sigma^* - \textit{twice}(L) \; .$$

$\mathcal{R}(e^f)$ is $\textit{minFirst}(\mathcal{D}(e)^{\trianglerighteq})$ where

$$\begin{aligned} s \; \textit{minFirst}(L) \; s' \quad \stackrel{\text{def}}{\iff} \quad & (\exists u, v) \; s = us'v \\ & \text{and } uv \neq \epsilon \\ & \text{and } s' \in L \\ & \text{and } us' \in \textit{nmf}(L) \end{aligned}$$

and $\mathcal{R}(e^l)$ is $\textit{minLast}(\mathcal{D}(e)^{\trianglerighteq})$ where

$$\begin{aligned} s \; \textit{minLast}(L) \; s' \quad \stackrel{\text{def}}{\iff} \quad & (\exists u, v) \; s = us'v \\ & \text{and } uv \neq \epsilon \\ & \text{and } s' \in L \\ & \text{and } s'v \in \textit{nmf}(L). \end{aligned}$$

Finally, $\mathcal{R}(\alpha^<)$ is $\textit{init}(\mathcal{R}(\alpha))$ where

$$\begin{aligned} s \; \textit{init}(R) \; s' \quad \stackrel{\text{def}}{\iff} \quad & (\exists s'', s''') \; s = s's''s''' \\ & \text{and } s \; R \; s'' \end{aligned}$$

while $\mathcal{R}(\alpha^>)$ is $\textit{fin}(\mathcal{R}(\alpha))$ where

$$\begin{aligned} s \; \textit{fin}(R) \; s' \quad \stackrel{\text{def}}{\iff} \quad & (\exists s'', s''') \; s = s'''s''s' \\ & \text{and } s \; R \; s'' \; . \end{aligned}$$

Proposition 3. *If L is a regular language and R is a regular relation, then*

(i) $\textit{Mult}(L)$, $R^{-1}L$, and $\textit{nmf}(L)$ are regular languages

(ii) R_L, $\textit{minFirst}(L)$, $\textit{minLast}(L)$, $\textit{init}(R)$ and $\textit{fin}(R)$ are regular relations.

4.2 Beyond sub-intervals

As is clear from the relations $\mathcal{R}(e)$, $\mathcal{TPL}$ makes do with the sub-interval relation $\subset$ and a "quasi-guarded" fragment at that (Pratt-Hartmann, 2005, page 5). To string out the interval temporal logic $\mathcal{HS}$ (Halpern and Shoham, 1991), the key is to combine $\mathcal{A}$ and I using some $r \notin E$ to mark I (rather than forming $\mathcal{A}_I$)

$$\mathcal{A}_r[I] \quad \stackrel{\text{def}}{=} \quad \mathcal{A} \cup \{\langle I, r \rangle\}$$

and modify $\textit{str}(\mathcal{A}, I)$ to define

$$\textit{str}_r(\mathcal{A}, I) \quad \stackrel{\text{def}}{=} \quad s(\mathcal{A}_r[I])^\bullet \; .$$

Let us agree that (i) a string $\alpha_1 \cdots \alpha_n$ *r-marks* I if $\langle I, r\rangle \in \bigcup_{i=1}^{n} \alpha_i$, and that (ii) a string is *r-marked* if there is a unique I that it r-marks. For every r-marked string s, we define two strings: let $s \upharpoonright r$ be the factor of s with *bgn-I* in its first box and *I-end* in its last, where s r-marks I; and let s_{-r} be $\rho_\Sigma(s \upharpoonright r)$.[3] We can devise a finite-state transducer converting r-marked strings s into s_{-r}, which we can then apply to evaluate an event-atom e as an $\mathcal{HS}$-formula

$$s \in \mathcal{T}_r(e) \quad \stackrel{\text{def}}{\iff} \quad (\exists s' \in \mathcal{D}(e))\ s_{-r} \trianglerighteq s' \ .$$

It is also not difficult to build finite-state transducers for the accessibility relations $\mathcal{R}_r(\mathtt{B})$,$\mathcal{R}_r(\mathtt{E})$, $\mathcal{R}_r(\overline{\mathtt{B}})$, and $\mathcal{R}_r(\overline{\mathtt{E}})$, showing that, as in $\mathcal{TPL}$, the connectives in $\mathcal{HS}$ map regular languages and regular relations to regular languages and regular relations. The question for both $\mathcal{TPL}$ and $\mathcal{HS}$ is can we start with regular languages $\mathcal{D}(e)$? As noted towards the end of section 3, one way is to reduce the set $\mathfrak{I}$ of intervals to a finite set. We close with an alternative.

4.3 A modest proposal: splitting event-atoms

An alternative to $\mathcal{D}(e) = \bigcup_{I \in \mathfrak{I}} \mathcal{D}(e, I)$ is to ask what it is that makes an e-event an e-event, and encode that answer in $\mathcal{D}(e)$. In and of itself, an interval [3,9] cannot make $\langle[3, 9], e\rangle$ an e-event, because in and of itself, $\langle[3, 9], e\rangle$ is *not* an e-event. $\langle[3, 9], e\rangle$ is an e-event only *in* a model $\mathcal{A}$ such that $\mathcal{A}([3, 9], e)$.

Putting $\mathfrak{I}$ aside, let us suppose, for instance, that e were the event *Pat swim a mile*. We can represent the "internal temporal contour" of e through a parametrized temporal proposition $f(r)$ with parameter r ranging over the reals in the unit interval $[0, 1]$, and $f(r)$ saying *Pat has swum* r*·(a mile)*. Let $\mathcal{D}(e)$ be

$$\boxed{f(0)}\boxed{f_\uparrow}^{+}\boxed{f(1)}$$

where $f_\uparrow$ abbreviates the temporal proposition

$$(\exists r < 1)\ f(r) \wedge \textit{Previously}\ \neg f(r) \ .$$

[3] Σ is defined as in §3.1, and ρ_X as in §1.1 above. Were we to weaken $\subset$ to $\subseteq$ in the definition of $\mathcal{A}_I$ and the semantics of $\mathcal{TPL}$, then we would have $(\mathit{str}_r(\mathcal{A}, I))_{-r} = \mathit{str}(\mathcal{A}, I)$, and truth sets $\mathcal{T}_r(\varphi)$ and accessibility relations $\mathcal{R}_r(\beta)$ such that

$$\begin{aligned} \mathcal{T}(\varphi) &= \{s_{-r} \mid s \in \mathcal{T}_r(\varphi)\} \\ \mathcal{R}(\beta) &= \{\langle s_{-r}, s'_{-r}\rangle \mid s\ \mathcal{R}_r(\beta)\ s'\} \end{aligned}$$

for $\mathcal{TPL}_{+}$-formulas φ and extended labels β.

Notice that the temporal propositions $f(r)$ and $f_\uparrow$ are to be interpreted over points (as in LTL); as illustrated in Example B above, however, these points can be split by adding boxables. Be that as it may, it is straightforward to adjust our definition of a model $\mathcal{A}$ and $\mathit{str}_r(\mathcal{A}, I)$ to accommodate such changes to $\mathcal{D}(e)$. Basing the truth sets $\mathcal{T}(\varphi)$ on sets $\mathcal{D}(e)$ of e-denotations independent of a model $\mathcal{A}$ (Fernando, 2011a) is in line with the proposal of (Steedman, 2005) mentioned at the end of §3.4 above.

References

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of the Association for Computing Machinery* 26(11): 832–843.

Kenneth R. Beesley and Lauri Karttunen. 2003. *Finite State Morphology*. CSLI, Stanford, CA.

Michael Bennett and Barbara Partee. 1972. Toward the logic of tense and aspect in English. Indiana University Linguistics Club, Bloomington, IN.

J.F.A.K. van Benthem. 1983. *The Logic of Time*. Reidel.

E. Allen Emerson. 1990. Temporal and modal logic. In (J. van Leeuwen, ed.) *Handbook of Theoretical Computer Science*, volume B. MIT Press, 995–1072.

Tim Fernando. 2004. A finite-state approach to events in natural langue semantics. *J. Logic & Comp* 14:79–92.

Tim Fernando. 2011. Constructing situations and time. *J. Philosophical Logic* 40(3):371–396.

Tim Fernando. 2011a. Regular relations for temporal propositions. *Natural Language Engineering* 17(2): 163–184.

Joseph Y. Halpern and Yoav Shoham. 1991. A Propositional Modal Logic of Time Intervals. *J. Association for Computing Machinery* 38(4): 935–962.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic*. Kluwer, Dordrecht.

John E. McTaggart. 1908. The Unreality of Time. *Mind* 17:456–473.

Ian Pratt-Hartmann. 2005. Temporal prepositions and their logic. *Artificial Intelligence* 166: 1–36.

Ian Pratt-Hartmann. 2005a. From TimeML to TPL*. In (G. Katz et al., eds.) *Annotating, Extracting and Reasoning about Time and Events*, Schloss Dagstuhl.

James Pustejovsky, José Castaño, Robert Ingria, Roser Saurí, Robert Gaizauskas, Andrea Setzer and Graham Katz. 2003. TimeML: Robust Specification of Event and Temporal Expressions in Text. In *5th International Workshop on Computational Semantics*. Tilburg.

Mark Steedman. 2005. The Productions of Time: Temporality and Causality in Linguistic Semantics. Draft, homepages.inf.ed.ac.uk/steedman/papers.html.