A Finite-State Temporal Ontology and Event-intervals

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

Abstract

A finite-state approach to temporal ontology for natural language text is described under which intervals (of the real line) paired with event descriptions are encoded as strings. The approach is applied to an interval temporal logic linked to TimeML, a standard mark-up language for time and events, for which various finite-state mechanisms are proposed.

1 Introduction

A model-theoretic perspective on finite-state methods is provided by an important theorem due to Büchi, Elgot and Trakhtenbrot (Thomas, 1997). Given a finite alphabet Σ, a system MSO_Σ of *monadic second-order logic* is set up with a binary relation symbol (for successor) and a unary relation symbol for each symbol in Σ so that the formulae of MSO_Σ define precisely the regular languages over Σ (minus the null string ϵ). Extensions of this theorem to infinite strings and trees are fundamental to work on formal verification associated with *Model Checking* (Clarke et al., 1999). In that work, a well-defined computational system (of hardware or software) can be taken for granted, against which to evaluate precise specifications. The matter is far more delicate, however, with natural language semantics. It is not clear what models, if any, are appropriate for natural language. Nor is it obvious what logical forms natural language statements translate to. That said, there is a considerable body of work in linguistic semantics that uses model theory, and no shortage of natural language text containing information that cries out for extraction.

A step towards (semi-)automated reasoning about temporal information is taken in TimeML (Pustejovsky et al., 2003), a "mark-up language for temporal and event expressions" (www.timeml.org). The primary aim of the present paper is to show how finite-state methods can push this step further, by building strings, regular languages and regular relations to represent some basic semantic ingredients proposed for TimeML. An instructive example is sentence (1), which is assigned in (Pratt-Hartmann, 2005a; ISO, 2007) the logical form (2).

(1) After his talk with Mary, John drove to Boston.

(2) $p(e) \wedge q(e') \wedge \text{after}(e, e')$

If we read $p(e)$ as "e is an event of John talking with Mary" and $q(e')$ as "e' is an event of John driving to Boston" then (2) says "an event e' of John driving to Boston comes after an event e of John talking with Mary ." Evidently, (1) follows from (3) and (4) below (implicitly quantifying the variables e and e' in (2) existentially).

(3) John talked with Mary from 1pm to 2pm.

(4) John drove to Boston from 2pm to 4pm.

But is (3) not compatible with (5) — and indeed implied by (5)?

(5) John talked with Mary from 1pm to 4pm.

Could we truthfully assert (1), given (4) and (5)? Or if not (1), perhaps (6)?

(6) After talking with Mary for an hour, John drove to Boston.

Proceedings of the 10th International Workshop on Finite State Methods and Natural Language Processing, pages 80–89,
Donostia–San Sebastián, July 23–25, 2012.

The acceptability of (6) suffers, however, if we are told (7).

(7) John drove toward Boston from 1pm to 2pm.

Clearly, individuating events, as (2) does, opens up a can of worms. But since at least (Davidson, 1967), there has been no retreating from events (Parsons, 1990; Kamp and Reyle, 1993; Pratt-Hartmann, 2005). Be that as it may, an appeal to events carries with it an obligation to provide a minimal account of what holds during these events and perhaps even a bit beyond. It is for such an account that finite-state methods are deployed below, viewed through the lens of the Büchi-Elgot-Trakhtenbrot theorem.

That lens gives temporal logic, the formulae of which — hereafter called *fluents* (for brevity) — may or may not hold at a string position, conceived as time and ordered according to succession within the string. For example, we can introduce a fluent p for "John talked with Mary" and a fluent q for "John drove to Boston" to form the string $\fbox{p}\fbox{q}$ (of length 2) for "after John talked with Mary, John drove to Boston." The idea is that a string $\alpha_1 \cdots \alpha_n$ of boxes α_i describes a sequence $t_1, \ldots, t_n$ of n times, t_i coming before t_{i+1}, such that every fluent in α_i holds at t_i.[1] To a first approximation, a box α_i is a snapshot at time t_i, making $\alpha_1 \cdots \alpha_n$ a cartoon or filmstrip. But just what is a time t_i: a temporal point or an interval?

For $\fbox{p}\fbox{q}$ to apply to (3) and (4), it is natural to regard t_i as an interval, setting up an account of the entailment from (5) to (3) in terms of the so-called *subinterval property* of John-talking-with-Mary (Bennett and Partee, 1972). John-driving-to-Boston, by contrast, does *not* have this property, necessitating the change from *to Boston* in (4) to *toward Boston* in (7). We can bring out this fact by representing individual events as strings, refining, for instance, our picture $\fbox{q}$ of John's drive to Boston by adding a fluent r for "John in Boston" to form $\fbox{q}\fbox{q,r}$. An event of motion is conceptualized as a finite sequence of snapshots in (Tenny, 1987) and elsewhere — a conceptualization resoundingly rejected in (Jackendoff, 1996) because

> it misrepresents the essential continuity of events of motion. For one thing, aside from the beginning and end points, the choice of a finite set of subevents is altogether arbitrary. How many subevents are there, and how is one to choose them? Notice that to stipulate the subevents as equally spaced, for instance one second or 3.5 milliseconds apart, is as arbitrary and unmotivated as any other choice.
>
> Another difficulty with a snapshot conceptualization concerns the representation of nonbounded events (activities) such as John ran along the river (for hours). A finite sequence of subevents necessarily has a specified beginning and ending, so it cannot encode the absence of endpoints. And excluding the specified endpoints simply exposes other specified subevents, which thereby become new endpoints. Thus encoding nonbounded events requires major surgery in the semantic representation. [page 316]

Jackendoff's objections are overcome below by finite-state manipulations that may well be called surgery. Following details supplied in the next section,[2] strings are formed from a finite set X of fluents that is allowed to vary so that

(i) the continuity desired by Jackendoff arises in the inverse limit of a system of projections π_X (defined below; Table 1), and

(ii) the temporal span of any finite string may, on expanding the set X, stretch without bound to the left (past) and/or to the right (future).

Applying π_X, section 2 proceeds to encode a model $\mathcal{A}$ of an interval temporal logic as a string $\mathsf{s}(\mathcal{A})$. Building on that encoding, section 3 develops finite-state methods for interval temporal logic. Section 4 concludes with proposals (drawing on work of the earlier sections) for extending the empirical (linguistic) coverage.

2 From event-intervals to strings

Before equating the set X of fluents with a model interpreting TimeML, let us bring out the intuition

[1] The alphabet Σ from which strings are formed is the family *Pow*(X) of subsets of some set X of fluents. A fluent corresponds to a monadic second-order variable in the Büchi-Elgot-Trakhtenbrot theorem.

[2] The present work extends a line of research most recently reported in (Fernando, 2011, 2011a, 2011b, 2012). That line is related to (Niemi and Koskenniemi, 2009), from which it differs in adopting an alphabet *Pow*(X) that equates sucession in a string with temporal succession.

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

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

$$\mathit{unpad}(s) \stackrel{\text{def}}{=} \begin{cases} \mathit{unpad}(s') & \text{if } s = \boxed{}\, s' \text{ or } s' \boxed{} \\ s & \text{otherwise} \end{cases}$$

Table 1: Behind $\pi_X(s) \stackrel{\text{def}}{=} \mathit{unpad}(\mathit{bc}(\rho_X(s)))$

underlying the function π_X through a familiar example. We can represent a calendar year by the string

$$s_{mo} \stackrel{\text{def}}{=} \boxed{\text{Jan}}\boxed{\text{Feb}} \cdots \boxed{\text{Dec}}$$

of length 12 (with a month in each box), or (adding one of 31 days d1, d2,. . ., d31) the string

$$s_{mo,dy} \stackrel{\text{def}}{=} \boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}} \cdots \boxed{\text{Jan,d31}}\boxed{\text{Feb,d1}} \cdots \boxed{\text{Dec,d31}}$$

of length 365 (a box per day in a non-leap year).[3] Unlike the points in say, the real line $\mathbb{R}$, a box can split if we enlarge the set X of fluents we can put in it, as illustrated by the change from $\boxed{\text{Jan}}$ in s_{mo} to $\boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}} \cdots \boxed{\text{Jan,d31}}$ in $s_{mo,dy}$. Two functions link the strings $s_{mo,dy}$ and s_{mo}

(i) a function ρ_{mo} that keeps only the months in a box so that

$$\rho_{mo}(s_{mo,dy}) = \boxed{\text{Jan}}^{31}\boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}$$

(ii) *block compression* bc, which compresses consecutive occurrences of a box into one, mapping $\rho_{mo}(s_{mo,dy})$ to

$$\mathit{bc}(\boxed{\text{Jan}}^{31}\boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}) = s_{mo}$$

so that $\mathit{bc}(\rho_{mo}(s_{mo,dy})) = s_{mo}$. As made precise in Table 1, ρ_X "sees only X" (setting $mo \stackrel{\text{def}}{=} \{$Jan, Feb, . . . Dec$\}$ to make ρ_{mo} an instance of ρ_X), while bc eliminates stutters, hardwiring the view that time passes only if there is change (or rather: we observe time passing only if we observe a change within a box). As this example shows, temporal granularity depends on the set X of observables that may go inside a box. Writing bc_X for the composition mapping s to $\mathit{bc}(\rho_X(s))$, we have

$$\mathit{bc}_{\{\text{Jan}\}}(s_{mo,dy}) = \mathit{bc}_{\{\text{Jan}\}}(s_{mo}) = \boxed{\text{Jan}}\boxed{}$$
$$\mathit{bc}_{\{\text{Feb}\}}(s_{mo,dy}) = \mathit{bc}_{\{\text{Feb}\}}(s_{mo}) = \boxed{}\boxed{\text{Feb}}\boxed{}$$
$$\mathit{bc}_{\{\text{d3}\}}(s_{mo,dy}) = (\boxed{}\boxed{\text{d3}})^{12}\boxed{}.$$

Now, the function π_X is bc_X followed by the deletion *unpad* of any initial or final empty boxes $\boxed{}$ (Table 1).[4] We can then define a fluent x to be an *s-interval* if $\pi_{\{x\}}(s)$ is $\boxed{x}$. Next, let $\mathcal{L}_\pi(X)$ be the language $\pi_X[\bigcap_{x \in X} \pi^{-1}_{\{x\}}\boxed{x}]$ consisting of strings $\pi_X(s)$ for $s \in \mathit{Pow}(X)^*$ such that $\pi_{\{x\}}(s) = \boxed{x}$ for all $x \in X$. Note that $\mathcal{L}_\pi(\{x\}) = \{\boxed{x}\}$ while for $x \neq x'$, $\mathcal{L}_\pi(\{x, x'\})$ consists of 13 strings s_R, one per interval relation R in (Allen, 1983); see columns 1 and 2 of Table 2

$$\mathcal{L}_\pi(\{x, x'\}) \;=\; \{s_R \mid R \in \textbf{Allen}\}\ .$$

$R \in$**Allen**	$s_R \in \mathcal{L}_\pi(\{x,x'\})$	$\chi_R([a,b],[a',b'])$
x = x'	$\boxed{x,x'}$	$a = a',\ \ b = b'$
x s x'	$\boxed{x,x'}\boxed{x'}$	$a = a',\ \ b < b'$
x si x'	$\boxed{x,x'}\boxed{x}$	$a = a',\ \ b' < b$
x f x'	$\boxed{x'}\boxed{x,x'}$	$a' < a,\ \ b = b'$
x fi x'	$\boxed{x}\boxed{x,x'}$	$a < a',\ \ b = b'$
x d x'	$\boxed{x'}\boxed{x,x'}\boxed{x'}$	$a' < a,\ \ b < b'$
x di x'	$\boxed{x}\boxed{x,x'}\boxed{x}$	$a < a',\ \ b' < b$
x o x'	$\boxed{x}\boxed{x,x'}\boxed{x'}$	$a < a' \leq b < b'$
x oi x'	$\boxed{x'}\boxed{x,x'}\boxed{x}$	$a' < a \leq b' < b$
x m x'	$\boxed{x}\boxed{x'}$	--
$x < x'$	$\boxed{x}\boxed{}\boxed{x'}$	$b < a'$
x mi x'	$\boxed{x'}\boxed{x}$	--
$x > x'$	$\boxed{x'}\boxed{}\boxed{x}$	$b' < a$

Table 2: The Allen relations via $\pi_{\{x,x'\}}$

[3] In (Niemi and Koskenniemi, 2009), s_{mo} is represented as the string

[m Jan]m [m Feb]m [m Mar]m ... [m Dec]m

of length 36 over 14 symbols (the 12 months plus the 2 brackets [m and]m) on which finite-state transducers operate. (See the previous footnote.)

[4] Restricted to a finite alphabet, the maps ρ_X, bc, *unpad* and π_X are computable by finite-state transducers (Fernando, 2011).

For example, in the case of the "*f*inish" relation f$\in$**Allen**,

$$s \models x\, \texttt{f} x' \iff \pi_{\{x,x'\}}(s) = \boxed{x'}\boxed{x, x'}$$

provided x and x' are s-intervals. The third column of Table 2 characterizes $R \in$**Allen** as conditions χ_R on pairs $[a, b]$ and $[a', b']$ of real numbers (in $\mathbb{R}$) denoting closed intervals[5] — e.g.,

$$[a, b]\ \texttt{f}\ [a', b'] \iff a' < a \text{ and } b = b' .$$

This brings us to the semantics of TimeML proposed in (Pratt-Hartmann, 2005a). A system $\mathcal{TPL}$ of *Temporal Preposition Logic* is built from an infinite set E of *event-atoms*, and interpreted relative to the family

$$\mathcal{I} \stackrel{\text{def}}{=} \{[a, b] \mid a, b \in \mathbb{R} \text{ and } a \leq b\}$$

of closed, bounded non-empty intervals in $\mathbb{R}$. A $\mathcal{TPL}$*-model* $\mathcal{A}$ is defined to be a finite subset of $\mathcal{I} \times E$. The intuition is that a pair $\langle I, e\rangle$ in $\mathcal{A}$ represents "an occurrence of an event of type e over the interval" I (Pratt-Hartmann, 2005; page 17), reversing the construal in line (2) above of e as a token. Identifying occurrences with events, we can think of $\mathcal{A}$ as a finite set of events, conceived as "intervals cum description" (van Benthem, 1983; page 113). Treating events as fluents, we have

Proposition 1. *For every $\mathcal{TPL}$-model $\mathcal{A}$, there is a unique string $\mathsf{s}(\mathcal{A}) \in \mathcal{L}_\pi(\mathcal{A})$ such that for all $x, x' \in \mathcal{A}$ with $x = \langle I, e\rangle$ and $x' = \langle I', e'\rangle$,*

$$\pi_{\{x,x'\}}(\mathsf{s}(\mathcal{A})) = s_R \iff \chi_R(I, I')$$

*for $R \in$**Allen** and s_R, χ_R specified in Table 2.*

To construct the string $\mathsf{s}(\mathcal{A})$, let *Ends*$(\mathcal{A})$ be the set of endpoints of $\mathcal{A}$

$$Ends(\mathcal{A}) \stackrel{\text{def}}{=} \bigcup_{I \in dom(\mathcal{A})} ends(I)$$

where *dom*$(\mathcal{A})$ is the domain $\{I \mid (\exists e \in E)\ \langle I, e\rangle \in \mathcal{A}\}$ of $\mathcal{A}$, and *ends*$([a, b])$ is the unordered pair $\{a, b\}$. Sorting gives *Ends*$(\mathcal{A}) = \{r_1, r_2, \ldots, r_n\}$ with $r_1 < r_2 < \cdots < r_n$. Breaking $[r_1, r_n]$ up into $2n - 1$ intervals, let

$$\alpha_i \stackrel{\text{def}}{=} \{\langle I, e\rangle \in \mathcal{A} \mid r_i \in I\} \quad \text{for } 1 \leq i \leq n$$

and

$$\beta_i \stackrel{\text{def}}{=} \{\langle I, e\rangle \in \mathcal{A} \mid [r_i, r_{i+1}] \subseteq I\} \quad \text{for } 1 \leq i < n.$$

Interleaving and block-compressing give

$$\mathsf{s}(\mathcal{A}) \stackrel{\text{def}}{=} bc(\alpha_1\beta_1 \cdots \alpha_{n-1}\beta_{n-1}\alpha_n)$$

$x_1 \stackrel{\text{def}}{=} \langle[1, 5], e\rangle$	$r_1 = 1,\ r_2 = 4$
$x_2 \stackrel{\text{def}}{=} \langle[4, 9], e\rangle$	$r_3 = 5,\ r_4 = 9$
$x_3 \stackrel{\text{def}}{=} \langle[9, 50], e'\rangle$	$r_5 = 50$
$\mathcal{A} \stackrel{\text{def}}{=} \{x_1, x_2, x_3\}$	

Table 3: Example $\mathsf{s}(\mathcal{A}) = \boxed{x_1}\boxed{x_1, x_2}\boxed{x_2}\boxed{x_2, x_3}\boxed{x_3}$

(see Table 3 for an example). One may then verify (by induction on the cardinality of the domain of $\mathcal{A}$) that $\mathsf{s}(\mathcal{A})$ is the unique string in $\mathcal{L}_\pi(\mathcal{A})$ satisfying the equivalence in Proposition 1.

But is encoding $\mathcal{A}$ as a string $\mathsf{s}(\mathcal{A})$ adequate for $\mathcal{TPL}$-satisfaction? Let us introduce $\mathcal{TPL}$-formulae through an English example.

(8) During each of John's drives to Boston, he ate a donut.

(8) translates in $\mathcal{TPL}$ to (9), which is interpreted relative to a $\mathcal{TPL}$-model $\mathcal{A}$ and an interval $I \in \mathcal{I}$ according to (10) and (11), with $[e]\varphi$ abbreviating $\neg\langle e\rangle\neg\varphi$ (as usual), $\top$ a tautology (in that $\mathcal{A} \models_I \top$ always) and $\subset$ as strict (irreflexive) subset.

(9) [John-drive-to-Boston] $\langle$John-eat-a-donut$\rangle\top$

(10) $\mathcal{A} \models_I \langle e\rangle\varphi \stackrel{\text{def}}{\iff} (\exists J \subset I \text{ s.t. } \mathcal{A}(J, e))\ \mathcal{A} \models_J \varphi$

(11) $\mathcal{A} \models_I \neg\varphi \stackrel{\text{def}}{\iff} \text{not } \mathcal{A} \models_I \varphi$

Clause (10) shows off a crucial feature of $\mathcal{TPL}$: quantification over intervals is bounded by the domain of $\mathcal{A}$; that is, quantification is restricted to intervals that are paired up with an event-atom by the

[5] Over non-empty closed intervals that include points $[a, a]$, the Allen relations m and mi collapse to o and oi, respectively. Alternatively, we can realize m and mi by trading closed intervals for *open* intervals (required to be non-empty); see the Appendix below.

$\mathcal{TPL}$-model (making $\mathcal{TPL}$ "quasi-guarded"; Pratt-Hartmann, 2005; page 5). This is *not* to say that the only intervals I that may appear in forming $\mathcal{A} \models_I \varphi$ are those in the domain of $\mathcal{A}$. Indeed, for $[a, b] \in \mathcal{I}$ and $[a', b'] \in dom(\mathcal{A})$ such that $[a', b'] \subset [a, b]$, $\mathcal{TPL}$ uses intervals

$$\text{init}([a', b'], [a, b]) \stackrel{\text{def}}{=} [a, a']$$
$$\text{fin}([a', b'], [a, b]) \stackrel{\text{def}}{=} [b', b]$$

to interpret $\{e\}_< \varphi$ and $\{e\}_> \varphi$ according to (12).

(12) $$\mathcal{A} \models_I \{e\}_< \varphi \stackrel{\text{def}}{\iff} (\exists! J \subset I \text{ s.t. } \mathcal{A}(J, e)) \; \mathcal{A} \models_{\text{init}(J,I)} \varphi$$
$$\mathcal{A} \models_I \{e\}_> \varphi \stackrel{\text{def}}{\iff} (\exists! J \subset I \text{ s.t. } \mathcal{A}(J, e)) \; \mathcal{A} \models_{\text{fin}(J,I)} \varphi$$

The bang ! in $\exists! J$ in (12) expresses uniqueness, which means that under the translation of (1) as (13) below, the interval I of evaluation is required to contain a unique event of John talking with Mary.

(1) After $\underbrace{\text{his talk with Mary}}_{\text{p}}$ $\underbrace{\text{John drove to Boston}}_{\text{q}}$.

(13) $\{\text{p}\}_> \langle \text{q} \rangle \top$

For a translation of (1) more faithful to

(2) $p(e) \wedge q(e') \wedge \text{after}(e, e')$

than (13),[6] it suffices to drop ! in (12) for $\langle e \rangle_<$ and $\langle e \rangle_>$ in place of $\{e\}_<$ and $\{e\}_>$ respectively (Fernando, 2011a), and to revise (13) to $\langle \text{p} \rangle_> \langle \text{q} \rangle \top$. Relaxing uniqueness, we can form $[\text{p}]_> \langle \text{q} \rangle \top$ for *after every talk with Mary, John drove to Boston*, as well as $\langle \text{p} \rangle_> \langle \text{p} \rangle \top$ for *after a talk with Mary, John talked with Mary again*. $\mathcal{TPL}$ has further constructs e^f and e^l for the (minimal) first and (minimal) last e-events in an interval.

Returning to the suitability of $\mathsf{s}(\mathcal{A})$ for $\mathcal{TPL}$, consider the question: when do two pairs $\mathcal{A}, I$ and $\mathcal{A}', I'$ of $\mathcal{TPL}$-models $\mathcal{A}, \mathcal{A}'$ and intervals $I, I' \in \mathcal{I}$ satisfy the same $\mathcal{TPL}$-formulae? Some definitions are in order. A bijection $f : A \to B$ between finite sets A and B of real numbers is *order-preserving* if for all $a, a' \in A$,

$$a < a' \iff f(a) < f(a')$$

in which case we write $f : A \cong B$. Given a $\mathcal{TPL}$-model $\mathcal{A}$, and a function $f : Ends(\mathcal{A}) \to \mathbb{R}$, let $\mathcal{A}^f$ be $\mathcal{A}$ with all its intervals renamed by f

$$\mathcal{A}^f \stackrel{\text{def}}{=} \{\langle [f(a), f(b)], e \rangle \mid \langle [a, b], e \rangle \in \mathcal{A}\} .$$

Now, we say $\mathcal{A}$ is *congruent with* $\mathcal{A}'$ and write $\mathcal{A} \cong \mathcal{A}'$ if there is an order-preserving bijection between $Ends(\mathcal{A})$ and $Ends(\mathcal{A}')$ that renames $\mathcal{A}$ to $\mathcal{A}'$

$$\mathcal{A} \cong \mathcal{A}' \stackrel{\text{def}}{\iff} (\exists f : Ends(\mathcal{A}) \cong Ends(\mathcal{A}')) \; \mathcal{A}' = \mathcal{A}^f .$$

Finally, we bring I into the picture by defining the *restriction $\mathcal{A}_I$ of $\mathcal{A}$ to I* to be the subset

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

of $\mathcal{A}$ with intervals strictly contained in I.

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

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

Proposition 2 suggests normalizing a $\mathcal{TPL}$ model $\mathcal{A}$ with endpoints $r_1 < r_2 < \cdots < r_n$ to $\mathsf{nr}(\mathcal{A})$ with r_i renamed to i

$$\mathsf{nr}(\mathcal{A}) \stackrel{\text{def}}{=} \mathcal{A}^f \quad \text{where } f \stackrel{\text{def}}{=} \{\langle r_1, 1 \rangle, \ldots, \langle r_n, n \rangle\}.$$

Assigning every $\mathcal{TPL}$-formula φ the *truth set*

$$\mathcal{T}(\varphi) \stackrel{\text{def}}{=} \{\mathsf{s}(\mathsf{nr}(\mathcal{A}_I)) \mid \mathcal{A} \text{ is a } \mathcal{TPL}\text{-model}, \; I \in \mathcal{I} \text{ and } \mathcal{A} \models_I \varphi\}$$

gives

Proposition 3. *For every $\mathcal{TPL}$-formula φ, $\mathcal{TPL}$-model $\mathcal{A}$, and interval $I \in \mathcal{I}$,*

$$\mathcal{A} \models_I \varphi \iff \mathsf{s}(\mathsf{nr}(\mathcal{A}_I)) \in \mathcal{T}(\varphi) .$$

To bolster the claim that $\mathcal{T}(\varphi)$ encodes $\mathcal{TPL}$-satisfaction, we may construct $\mathcal{T}(\varphi)$ by induction on φ, mimicking the clauses for $\mathcal{TPL}$-satisfaction, as in (14).

(14) $\mathcal{T}(\varphi \wedge \varphi') = \mathcal{T}(\varphi) \cap \mathcal{T}(\varphi')$

Details are provided in the next section, where we consider the finite-state character of the clauses, and may verify Propositions 2 and 3.

[6] Caution: e and e' are tokens in (2), but types in $\mathcal{TPL}$.

3 Regularity and relations behind truth

A consequence of Proposition 3 is that the entailment from φ to φ' given by

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

becomes equivalent to the inclusion $\mathcal{T}(\varphi) \subseteq \mathcal{T}(\varphi')$, or to the unsatisfiability of $\varphi \wedge \neg\varphi'$

$$\varphi \vdash_{\mathcal{I},E} \varphi' \iff \mathcal{T}(\varphi \wedge \neg\varphi') = \emptyset$$

assuming classical interpretations (14) and (15) of conjunction $\wedge$ and negation $\neg$.

(15) $\mathcal{T}(\neg\varphi) \;=\; \Sigma^+ - \mathcal{T}(\varphi)$

Finite-state methods are of interest as regular languages are closed under intersection and complementation. (Context-free languages are not; nor is containment between context-free languages decidable.) The alphabet Σ in (15) is, however, infinite; Σ is the set $\mathit{Fin}(\mathcal{J} \times E)$ of finite subsets of $\mathcal{J} \times E$, where $\mathcal{J}$ is the set

$$\mathcal{J} \stackrel{\text{def}}{=} \{[n,m] \in \mathcal{I} \mid n,m \in \mathbb{Z}_+\}$$

of intervals in $\mathcal{I}$ with endpoints in the set $\mathbb{Z}_+$ of positive integers $1, 2, \ldots$ (containing the domain of a normalized $\mathcal{TPL}$-model). As with π_X, regularity demands restricting Σ to a finite subalphabet — or better: subalphabet*s* given by the set $\mathcal{F}$ of pairs $\langle \mathcal{I}', E' \rangle$ of finite subsets $\mathcal{I}'$ and E' of $\mathcal{J}$ and E respectively, for which

$$\Sigma \;=\; \bigcup_{\langle \mathcal{I}', E' \rangle \in \mathcal{F}} \mathit{Pow}(\mathcal{I}' \times E') \; .$$

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 φ, bounds the size of a minimal model of φ. We get a computable function mapping a $\mathcal{TPL}$-formula φ to a finite subset $\mathcal{I}_\varphi$ of $\mathcal{J}$ just big enough so that if φ is $\mathcal{TPL}$-satisfiable,

$$(\exists \mathcal{A} \in \mathit{Pow}(\mathcal{I}_\varphi \times E_\varphi))(\exists I \in \mathcal{I}_\varphi) \; \mathcal{A} \models_I \varphi$$

where E_φ is the finite subset of E occurring in φ. To minimize notational clutter, we leave out the choice $\langle \mathcal{I}', E' \rangle \in \mathcal{F}$ of a finite alphabet below.

Next, keeping intersection and complementation in (14) and (15) in mind, let us call an operation *regularity-preserving* (rp) if its output is regular whenever all its inputs are regular. To interpret $\mathcal{TPL}$, we construe operations broadly to allow their inputs and output to range over relations between strings (and not just languages), construing a relation to be *regular* if it is computable by a finite-state transducer. For instance, the modal diamond $\langle e \rangle$ labelled by an event-atom $e \in E$ is interpreted via an accessibility relation $\mathcal{R}(e)$ in the usual Kripke semantics

$$\mathcal{T}(\langle e \rangle \varphi) \;=\; \mathcal{R}(e)^{-1} \mathcal{T}(\varphi)$$

of $\langle e \rangle \varphi$ where $R^{-1}L$ is the set $\{s \in \Sigma^* \mid (\exists s' \in L) \; sRs'\}$ of strings related by R to a string in L. The operation that outputs $R^{-1}L$ on inputs R and L is rp. But what is the accessibility relation $\mathcal{R}(e)$?

Three ingredients go into making $\mathcal{R}(e)$:

(i) a notion of strict containment $\sqsupset$ between strings

(ii) the demarcation $s^\bullet$ of a string s

(iii) a set $\mathcal{D}(e)$ of strings representing full occurrences of e.

We take up each in turn, starting with $\sqsupset$, which combines two ways a string can be part of another. To capture strict inclusion $\subset$ between intervals, we say a string s' is a *proper factor of* a string s, and write $s \; \mathit{pfac} \; s'$, if s' is s with some prefix u and suffix v deleted, and uv is non-empty

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

(Dropping the requirement $uv \neq \epsilon$ gives *factors* simpliciter.) The second way a string s' may be part of s applies specifically to strings 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

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

Now, writing $R; R'$ for the *relational composition of* binary relations R and R' in which the output of R is fed as input to R'

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

we compose *pfac* with $\trianglerighteq$ for *strict containment* $\sqsupset$

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

(It is well-known that relational composition ; is rp.) Next, the idea behind demarcating a string s is to mark the beginning and ending of every interval I mentioned in s, with fresh fluents *bgn-I* and *I-end*. The *demarcation* $(\alpha_1\alpha_2\cdots\alpha_n)^\bullet$ of $\alpha_1\alpha_2\cdots\alpha_n$ adds *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 $\langle I, e\rangle \notin \alpha_{i-1}$

and adds *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 $\langle I, e\rangle \notin \alpha_{i+1}$.[7]

For $s = \mathsf{s}(\mathcal{A})$ given by the example in Table 3,

$$s^\bullet = \boxed{x_1, \textit{bgn-}I_1} \boxed{x_1, x_2, I_1\textit{-end}, \textit{bgn-}I_2} \boxed{x_2} \boxed{x_2, x_3, I_2\textit{-end}, \textit{bgn-}I_3} \boxed{x_3, I_3\textit{-end}}$$

We then form the *denotation* $\mathcal{D}_{\mathcal{I}'}(e)$ *of* e *relative to* a finite subset $\mathcal{I}'$ of $\mathcal{I}$ by demarcating every string in $\bigcup_{I\in\mathcal{I}'} \boxed{\langle I, e\rangle}^+$ as in (16).

$$\text{(16)} \quad \mathcal{D}_{\mathcal{I}'}(e) \stackrel{\text{def}}{=} \bigcup_{I\in\mathcal{I}'}\{s^\bullet \mid s \in \boxed{\langle I, e\rangle}^+\}$$

To simplify notation, we suppress the subscript $\mathcal{I}'$ on $\mathcal{D}_{\mathcal{I}'}(e)$. Restricting strict containment $\sqsupset$ to $\mathcal{D}(e)$ gives

$$s \,\mathcal{R}_\circ(e)\, s' \stackrel{\text{def}}{\iff} s \sqsupset s' \text{ and } s' \in \mathcal{D}(e)$$

from which we define $\mathcal{R}(e)$, making adjustments for demarcation

$$s \,\mathcal{R}(e)\, s' \stackrel{\text{def}}{\iff} s^\bullet \,\mathcal{R}_\circ(e)\, s'^\bullet .$$

That is, $\mathcal{R}(e)$ is the composition $\cdot^\bullet; \mathcal{R}_\circ(e); \cdot_\bullet$ where demarcation $\cdot^\bullet$ is inverted by $\cdot_\bullet$. As $\mathcal{TPL}$'s other constructs are shown in §4.1 of (Fernando, 2011a) to be interpretable by rp operations, we have

[7]The markers *bgn-I* and *I-end* are analogous to the brackets [g and]g in (Niemi and Koskenniemi, 2009), an essential difference being that a grain (type) g supports multiple occurrences of [g and]g, in contrast to the (token) interval I.

Proposition 4. *All $\mathcal{TPL}$-connectives can be interpreted by rp operations.*

Beyond $\mathcal{TPL}$, the interval temporal logic $\mathcal{HS}$ of (Halpern and Shoham, 1991) suggests variants of $\langle e\rangle\varphi$ with strict containment $\sqsupset$ in $\mathcal{R}(e)$ replaced by any of Allen's 13 interval relations R.

$$\text{(17)} \quad \mathcal{A} \models_I \langle e\rangle_R \varphi \stackrel{\text{def}}{\iff} (\exists J \text{ s.t. } I \, R \, J) \; \mathcal{A}(J, e) \text{ and } \mathcal{A} \models_J \varphi$$

To emulate (17), we need to mark the evaluation interval I in $\mathcal{A}$ by some $r \notin E$, setting

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

rather than simply forming $\mathcal{A}_I$ (which will do if we can always assume the model's full temporal extent is marked). A string $s = \alpha_1 \cdots \alpha_n$ *r-marks I* if $\langle I, r\rangle \in \bigcup_{i=1}^n \alpha_i$. If that interval is unique, we say s is *r-marked*, and write $\mathsf{I}(s)$ for the interval it r-marks, and s_- for s with the fluent $\langle \mathsf{I}(s), r\rangle$ deleted (so that $\mathsf{s}(\mathcal{A}_r[I])_- = \mathsf{s}(\mathcal{A})$). For any of the relations $R\in$**Allen**, we let $\approx_R$ hold between r-marked strings that are identical except possibly for the intervals they r-mark, which are related by R

$$s \approx_R s' \stackrel{\text{def}}{\iff} s_- = s'_- \text{ and } \mathsf{I}(s) \, R \, \mathsf{I}(s').$$

Next, given an event-atom e, we let $\mathcal{R}(e)_R$ be a binary relation that holds between r-marked strings related by $\approx_R$, the latter of which picks out a factor subsuming some string in $\mathcal{D}(e)$

$$s \,\mathcal{R}(e)_R\, s' \stackrel{\text{def}}{\iff} s \approx_R s' \text{ and } (\exists d \in \mathcal{D}(e)) \; s'_r \trianglerighteq d$$

where s'_r is the factor of s' that begins with *bgn*-$\mathsf{I}(s')$ and ends with $\mathsf{I}(s')$-*end*. Replacing $\mathcal{A}_I$ by $\mathcal{A}_r[I]$ in $\mathcal{T}(\varphi)$ for

$$\mathcal{T}_r(\varphi) \stackrel{\text{def}}{=} \{\mathsf{s}(\mathsf{nr}(\mathcal{A}_r[I])) \mid \mathcal{A} \text{ is a } \mathcal{TPL}\text{-model}, \; I \in \mathcal{I} \text{ and } \mathcal{A} \models_I \varphi\} ,$$

(17) corresponds to

$$\mathcal{T}_r(\langle e\rangle_R \varphi) = \mathcal{R}(e)_R^{-1} \, \mathcal{T}_r(\varphi).$$

4 Conclusion and future work

The key notion behind the analysis above of time in terms of strings is the map π_X, which for X consisting of interval-event pairs $\langle I, e\rangle$, is applied in Proposition 1 to turn a $\mathcal{TPL}$-model $\mathcal{A}$ into a string $\mathsf{s}(\mathcal{A})$. As far as $\mathcal{TPL}$-satisfaction $\mathcal{A} \models_I \varphi$ is concerned, we can normalize the endpoints of the intervals to an initial segment of the positive integers, after restricting $\mathcal{A}$ to intervals contained in the evaluation interval I (Proposition 3). For a finite-state encoding of $\mathcal{TPL}$-satisfaction, it is useful to demarcate the otherwise homogeneous picture $\boxed{\langle I, e\rangle}^+$ of $\langle I, e\rangle$, and to define a notion $\sqsupset$ of proper containment between strings. We close with further finite-state enhancements.

Demarcation is linguistically significant, bearing directly on telicity and the so-called Aristotle-Ryle-Kenny-Vendler classification (Dowty, 1979), illustrated by the contrasts in (18) and (19).

(18) John was driving $\vdash$ John drove

John was driving to L.A. $\not\vdash$ John drove to L.A.

(19) John drove <u>for</u> an hour

John drove to L.A. <u>in</u> an hour

The difference at work in (18) and (19) is that *John driving to L.A.* has a termination condition, *in*(*John, L.A.*), missing from *John driving*. Given a fluent such as *in*(*John, L.A.*), we call a language L *φ-telic* if for every $s \in L$, there is an $n \geq 0$ such that $s \unrhd \boxed{\neg\varphi}^n \boxed{\varphi}$ (which is to say: a string in L ends as soon as φ becomes true). L is *telic* if it is φ-telic, for some φ. Now, the contrasts in (18) and (19) can be put down formally to the language for *John driving to L.A.* being telic, but not that for *John driving* (Fernando, 2008).

The demarcation (via φ) just described does not rely on some set $\mathcal{I}'$ of intervals I from which fluents *bgn-I* and *I-end* are formed (as in $s^\bullet$ from section 3). There are at least two reasons for attempting to avoid $\mathcal{I}'$ when demarcating or, for that matter, building the set $\mathcal{D}(e)$ of denotations of e. The first is that under a definition such as (16), the number of e-events (i.e., events of type e) is bounded by the cardinality of $\mathcal{I}'$.

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

The second is that an interval arguably has little to do with an e-event being an e-event. An interval [4,9] does not, in and of itself, make $\langle[4, 9], e\rangle$ an e-event; $\langle[4, 9], e\rangle$ is an e-event only *in* a $\mathcal{TPL}$-model that says it is. An alternative is to express in strings what holds during an event that makes it an e-event. Consider the event type e of *Pat walking a mile*. Incremental change in an event of that type can be represented through a parametrized fluent $f(r)$ with parameter r ranging over the reals in the unit interval $[0, 1]$, such that $f(r)$ says *Pat has walked $r\cdot$(a mile)*. Let $\mathcal{D}(e)$ be

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

where $f_\uparrow$ abbreviates the fluent

$$(\exists r < 1)\ f(r) \wedge \mathsf{Previous}(\neg f(r)).$$

$\mathsf{Previous}$ is a temporal operator that constrains strings $\alpha_1 \cdots \alpha_n$ so that whenever $\mathsf{Previous}(\varphi)$ belongs to α_{i+1}, φ belongs to α_i; that is,

$$\boxed{}\boxed{\mathsf{Previous}(\varphi)} \Rightarrow \boxed{\varphi}\boxed{}$$

using an rp binary operator $\Rightarrow$ on languages that combines subsumption $\unrhd$ with constraints familiar from finite-state morphology (Beesley and Karttunen, 2003).

The borders and interior of $\langle I, e\rangle$ aside, there is the matter of locating an e-string in a larger string (effected in $\mathcal{TPL}$ through strict inclusion $\supset$, the string-analog of which is proper containment $\sqsupset$). But what larger string? The influential theory of tense and aspect in (Reichenbach, 1947) places e relative not only to the speech S but also to a *reference time* r, differentiating, for instance, the simple past $\boxed{e, r}\boxed{\text{S}}$ from the present perfect $\boxed{e}\boxed{\text{S},r}$, as required by differences in defeasible entailments $\mid\!\sim$, (20), and acceptability, (21).

(20) Pat has left Paris $\mid\!\sim$ Pat is not in Paris

Pat left Paris $\mid\!\not\sim$ Pat is not in Paris

(21) Pat left Paris. ($^?$Pat has left Paris.)

But Pat is back in Paris.

The placement of r provides a bound on the *inertia* applying to the postcondition of Pat's departure

(Fernando, 2008). The extension $\mathcal{A}_r[I]$ proposed in section 3 to the combination $\mathcal{A}_I$ (adequate for $\mathcal{TPL}$, but not $\mathcal{HS}$) explicitly r-marks the evaluation interval I, facilitating an account more intricate than simply $\sqsupset$ of e's occurrence in the larger string. $\mathcal{TPL}$ goes no further than Ramsey in analyzing *That Caesar died* as an ontological claim that an event of certain sort exists (Parsons, 1990), leading to the view of an event as a truthmaker (Davidson, 1967; Mulligan et al., 1984). The idea of an event (in isolation) as some sort of proof runs into serious difficulties, however, as soon as tense and aspect are brought into the picture; complications such as the *Imperfective Paradox* (Dowty, 1979), illustrated in (22), raise tricky questions about what it means for an event to exist and how to ground it in the world (speaking loosely) in which the utterance is made.

(22) John was drawing a circle when he ran out of ink.

But while the burden of proof may be too heavy to be borne by a single pair $\langle I, e\rangle$ of interval I and event-atom e, the larger picture in which the pair is embedded can be strung out, and a temporal statement φ interpreted as a binary relation $\mathbf{R}_\varphi$ between such strings that goes well beyond $\sqsupset$. The inputs to $\mathbf{R}_\varphi$ serve as indices, with those in the domain of $\mathbf{R}_\varphi$ supporting the truth of φ

$$\varphi \text{ is } \textit{true at } s \quad \stackrel{\text{def}}{\iff} \quad (\exists s')\; s\, \mathbf{R}_\varphi\, s'$$

(Fernando, 2011, 2012). In witnessing truth at particular inputs, the outputs of $\mathbf{R}_\varphi$ constitute denotations more informative than truth values, from which indices can be built bottom-up, in harmony with a semantic analysis of text from its parts (to which presumably TimeML is committed). An obvious question is how far finite-state methods will take us. Based on the evidence at hand, we have much further to go.

Acknowledgments

My thanks to Daniel Isemann for useful, sustained discussions. The work is supported by EI Grant # CC-2011-2601B:GRCTC.

References

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

James F. Allen and George Ferguson. Actions and events in interval temporal logic. *J. Logic and Computation*, 4(5):531–579, 1994.

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.

Edmund M. Clarke, Jr., Orna Grumberg and Doron A. Peled. 1999. *Model Checking*. MIT Press.

Donald Davidson. 1967. The logical form of action sentences. In *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press.

David Dowty. 1979. *Word Meaning and Montague Grammar*. Kluwer.

Tim Fernando. 2008. Branching from inertia worlds. *J. Semantics* 25:321–344.

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

Tim Fernando. 2011a. Strings over intervals. *TextInfer 2011 Workshop on Textual Entailment*, pages 50-58, Edinburgh (ACL Archives).

Tim Fernando. 2011b. Finite-state representations embodying temporal relations. In *9th International Workshop FSMNLP 2011*, Blois, pages 12–20. A revised, extended version is in the author's webpage.

Tim Fernando. 2012. Steedman's temporality proposal and finite automata. In *Amsterdam Colloquium 2011*, Sprnger LNCS 7218, pages 301-310.

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

ISO. 2007. ISO Draft International Standard 24617-1 *Semantic annotation framework — Part 1: Time and events*. ISO (Geneva).

Ray Jackendoff. 1996. The proper treatment of measuring out, telicity, and perhaps even quantification in English. *Natural Language and Linguistic Theory* 14:305–354.

Hans Kamp and Uwe Reyle. 1993. *From Disocurse to Logic*. Kluwer.

Kevin Mulligan, Peter Simons and Barry Smith. 1984. Truth-makers. *Philosophy and Phenomenological Research* **44**: 287–321.

Jyrki Niemi and Kimmo Koskenniemi. 2009. Representing and combining calendar information by using finite-state transducers. In *7th International Workshop FSMNLP 2008*, pages 122–33. Amsterdam.

Terry Parsons. 1990. *Events in the Semantics of English*. MIT Press.

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

Ian Pratt-Hartmann. 2005a. From TimeML to TPL*. In *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*, pages 337-353. Tilburg.

Hans Reichenbach. 1947. *Elements of Symbolic Logic*. Macmillan & Co (New York).

Carol Tenny. 1987. Grammaticalizing Aspect and Affectedness. PhD dissertation, Department of Linguistics and Philosophy, MIT.

Wolfgang Thomas. 1997. Languages, automata and logic. In *Handbook of Formal Languages: Beyond Words*, Volume 3, pages 389–455. Springer-Verlag.

Appendix: a case of "less is more"?

Because the set $\mathcal{I}$ of intervals from which a $\mathcal{TPL}$-model $\mathcal{A}$ is constructed includes singleton sets $[a, a] = \{a\}$ (for all real numbers a), there can never be events x and x' in $\mathcal{A}$ such that x meets (or abuts) x', $x \texttt{ m } x'$, according to Table 2 above. It is, however, easy enough to throw out sets $[a, a]$ from $\mathcal{I}$, requiring that for $[a, b] \in \mathcal{I}$, a be strictly less than b. (In doing so, we follow (Allen, 1983) and (Pratt-Hartmann, 2005a), but stray from (Pratt-Hartmann, 2005).) The result is that the overlap at b between $[a, b]$ and $[b, c]$ is deeemed un-observable (effectively re-interpreting closed intervals by their interiors, understood to be non-empty). The third column $\chi_R([a, b], [a', b'])$ in Table 2 is modified to a condition $[a, b]\ R^\circ\ [a', b']$ that differs on the cases where R is one of the four Allen relations `o`, `m`, `oi`, `mi`, splitting the disjunction $a' \leq b$ in `o` with `m`, and $a \leq b'$ in `oi` with `mi`.

$R \in$ **Allen**	$s_R \in \mathcal{L}_\pi(\{e_1, e_2\})$	$[a, b]\ \ R^\circ\ \ [a', b']$
x `o` x'	$\boxed{x}\boxed{x, x'}\boxed{x'}$	$a < a' < b < b'$
x `m` x'	$\boxed{x}\boxed{x'}$	$b = a'$
x `oi` x'	$\boxed{x'}\boxed{x, x'}\boxed{x}$	$a' < a < b' < b$
x `mi` x'	$\boxed{x'}\boxed{x}$	$b' = a$

All other rows in Table 2 are the same for $[a, b]\ R^\circ\ [a', b']$. The somewhat wasteful encoding $s(\mathcal{A})$ in Proposition 1 then becomes $s^\circ(\mathcal{A})$ in

Proposition 1°. *For every $\mathcal{TPL}$-model $\mathcal{A}$ such that $a < b$ for all $[a, b] \in dom(\mathcal{A})$, there is a unique string $\mathsf{s}^\circ(\mathcal{A}) \in \mathcal{L}_\pi(\mathcal{A})$ such that for all $x, x' \in \mathcal{A}$ with $x = \langle I, e\rangle$ and $x' = \langle I', e'\rangle$, and $R \in$***Allen**

$$\pi_{\{x,x'\}}(\mathsf{s}^\circ(\mathcal{A})) = s_R \iff I\ R^\circ I'.$$

The encoding $\mathsf{s}^\circ(\mathcal{A})$ is formed exactly as $s(\mathcal{A})$ is in section 2 above from the endpoints $r_1 < r_2 < \cdots < r_n$ of *dom*$(\mathcal{A})$, except that the α_i's for the endpoints r_i are dropped (these being un-observable), leaving us with the β_i's for $[r_i, r_{i+1}]$

$$\mathsf{s}^\circ(\mathcal{A}) \stackrel{\text{def}}{=} bc(\beta_1 \cdots \beta_{n-1}).$$

Beyond Proposition 1, the arguments above for $\mathsf{s}(\mathcal{A})$ carry over to $\mathsf{s}^\circ(\mathcal{A})$, with the requirement on a $\mathcal{TPL}$-model $\mathcal{A}$ that $a < b$ for all $[a, b] \in dom(\mathcal{A})$. It is noteworthy that (Pratt-Hartmann, 2005a) makes no mention that this requirement is a departure from (Pratt-Hartmann, 2005). Although the restriction $a < b$ rules out $\mathcal{TPL}$-models with points $[a, a]$ in their domain, it also opens $\mathcal{TPL}$ up to strings in which events meet — a trade-off accepted in (Allen and Ferguson, 1994). To properly accommodate points alongside larger intervals, we can introduce a fluent *indiv* marking out boxes corresponding to points $[a, a]$ (as opposed to divisible intervals $[a, b]$ where $a < b$), and re-define π_X to leave boxes with *indiv* in them alone. From this perspective, the restriction $a < b$ is quite compatible with π_X as defined above. But can we justify the notational overhead in introducing *indiv* and complicating π_X? We say no more here.