Linguist and Philos (2007) 30:527–550
DOI 10.1007/s10988-008-9026-1

RESEARCH ARTICLE

Observing events and situations in time

Tim Fernando

Published online: 22 April 2008

Abstract Events and situations are represented by strings of temporally ordered observations, on the basis of which the events and situations are recognized. Allen's basic interval relations are derived from superposing strings that mark interval boundaries, and Kamp's event structures are constructed as projective limits of strings. Observations are generalized to temporal propositions, leading to event-types that classify event-instances. Working with sets of strings built from temporal propositions, we obtain natural notions of bounded entailment from set inclusions. These inclusions are decidable if the sets are accepted by finite automata.

Keywords Event structure · Time

1 Introduction

Consider the phrase "rain from dawn to dusk." Suppose we were to analyze its meaning in terms of primitives rain, dawn and dusk that are true (or not) at particular times. In other words, each $p \in \{\mathsf{rain}, \mathsf{dawn}, \mathsf{dusk}\}$ is a temporal proposition interpreted relative to a set T of times and a function $[\![\cdot]\!]$ that maps p to a set $[\![p]\!] \subseteq T$ of times such that

$$t \in [\![p]\!] \quad \text{is read} \quad \text{“}p \text{ is true at } t\text{”}$$

for all $t \in T$. An obvious candidate for T is the set $\Re$ of real numbers. But is the notion of "dawn" defined so precisely that we can pin down when dawn ends

T. Fernando (✉)
Computer Science Department, Trinity College,
Dublin 2, Ireland
e-mail: tfernand@cs.tcd.ie

(or begins) with the infinite precision of $\Re$? That is, can we choose a real number x such that dawn is true (or false) at x but not at any nearby point $x + \delta$, no matter how small $\delta > 0$ is? Were we to model time by intervals over $\Re$ instead of elements of $\Re$, the problem becomes how to choose an interval I over $\Re$ such that dawn is true at I but not at any extension of I, however minute. For instance, if I were the unit interval

$$[0, 1] = \{x \in \Re \mid 0 \leq x \leq 1\}$$

consisting of real numbers between 0 and 1 (inclusive), can we distinguish I from intervals

$$[0, 1 + \delta] = [0, 1] \cup \{x \in \Re \mid 1 < x \leq 1 + \delta\}$$

or

$$[-\delta, 1] = [0, 1] \cup \{x \in \Re \mid -\delta \leq x < 0\}$$

for all $\delta > 0$? Is there not a sufficiently tiny $\delta > 0$ such that dawn can only be deemed true at $[0, 1]$ if it is also at $[0, 1 + \delta]$ and at $[-\delta, 1]$? Insisting that dawn began (or ended) at 6:03 and not a picosecond earlier (or later) has the smell of false precision.

A common approach to vague predicates is to live with *many* admittedly overdetermined interpretations $[\![\cdot]\!]$, calling them *supervaluations* (e.g. Fine 1975). An alternative pursued in this paper is to work with representations faithful to the bounded precision of observations. An example of such a representation is the string $\boxed{\text{dawn}}\boxed{\text{rain}}\boxed{\text{dusk}}$, representing three "successive" times $t_1, t_2, t_3 \in T$ such that dawn is observed at t_1, rain at t_2, and dusk at t_3. The notion of "successive" times can be made explicit through a binary relation su on T, relative to which we define an *su-chain* to be a finite sequence $t_1 t_2 \cdots t_n$ of times such that $t_i \, su \, t_{i+1}$ for $1 \leq i < n$. In general, a string $a_1 a_2 \cdots a_n$ of sets a_i of temporal propositions *su-represents* an *su*-chain $t_1 t_2 \cdots t_n$ (of the same length n) if for every integer i from 1 to n (inclusive), a_i consists of observations at t_i. See Table 1.[1] We are assuming here a notion of an observation at time t that we can treat as primitive or define in some way. Alternatively, resorting to the functions $[\![\cdot]\!]$ mentioned above, we might say that $a_1 a_2 \cdots a_n$ $(su, [\![\cdot]\!])$-*describes* an *su*-chain $t_1 t_2 \cdots t_n$ if for every i from 1 to n,

$$t_i \in [\![\varphi]\!] \quad \text{for every temporal proposition } \varphi \in a_i.$$

[1] As will become clear below, it will be useful to take a_i to be a set of temporal propositions (rather than a single temporal proposition). We enclose that set by a box (rather than by the usual curly braces $\{\cdot\}$) to reinforce the intuition that $a_1 a_2 \cdots a_n$ is a film-strip made of snapshots a_i. Since a box designates a set whose elements are written inside it, we have

$$\boxed{\varphi, \psi} = \boxed{\psi, \varphi} = \boxed{\varphi, \psi, \varphi}$$

and the empty box $\square$ is the empty set $\emptyset$ (conceived as a snapshot).

Table 1 Strings from temporal propositions

Temporal proposition	φ
Symbol (snapshot)	$a = \boxed{\varphi \ldots}$
String (film strip)	$a_1 a_2 \cdots a_n$

Note that a_i is *not* required to include every temporal proposition that is $[\![\cdot]\!]$-true at t_i. Observations at t are understood to be true at t; however, an arbitrary temporal proposition true at t need not be observed at t.

We can evaluate the string $\boxed{\mathsf{dawn}}\boxed{\mathsf{rain}}\boxed{\mathsf{dusk}}$ relative to various choices of T, su and $[\![\cdot]\!]$. For example, we might let T consist of half-closed real intervals

$$[x,y) \stackrel{\text{def}}{=} \{z \in \Re \mid x \leq z < y\}$$

for $x, y \in \Re$ such that $x < y$, and stipulate that su holds between members of T that border (or "meet") each other in that

$$[x,y)\ su\ [x',y') \stackrel{\text{def}}{\Leftrightarrow} y = x'$$

for all $[x,y)$ and $[x',y') \in T$. Under these definitions, successive intervals have no gaps between them:

$$\text{whenever } [x,y)\ su\ [x',y'), \quad [x,y) \cup [x',y') = [x,y').$$

Thus, if the interpretation $[\![\mathsf{rain}]\!]$ of rain has the property that for all $[x,y) \in T$,

$$[x,y) \in [\![\mathsf{rain}]\!] \iff \text{whenever } x < x' < y, \\ [x,x') \in [\![\mathsf{rain}]\!] \text{ and } [x',y) \in [\![\mathsf{rain}]\!]$$

then

$$\boxed{\mathsf{rain}}\ (su, [\![\cdot]\!])\text{-describes}[x,y) \iff \text{whenever } x < x' < y, \\ \boxed{\mathsf{rain}}\boxed{\mathsf{rain}}\ (su, [\![\cdot]\!])\text{-describes} \\ [x,x')[x',y)$$

so that the difference between

$$\boxed{\mathsf{dawn}}\boxed{\mathsf{rain}}\boxed{\mathsf{dusk}} \text{ and } \boxed{\mathsf{dawn}}\boxed{\mathsf{rain}}^n\boxed{\mathsf{dusk}} \text{ for } n > 1$$

(where s^0 is the null/empty string ϵ and $s^{n+1} = s^n s$) comes down to how many times we chop an interval $[x,y)$. A reason to chop $[x,y)$ up is to accommodate additional observations such as noon, 2pm and warm at sub-intervals of $[x,y)$, leading to strings such as

$$\boxed{\mathsf{dawn}}\boxed{\mathsf{rain}}\boxed{\mathsf{rain, noon}}\boxed{\mathsf{rain}}\boxed{\mathsf{rain, 2pm, warm}}\boxed{\mathsf{rain, warm}}\boxed{\mathsf{rain}}\boxed{\mathsf{dusk}}.$$

Having linked strings to interpretations $[\![\cdot]\!]$, we might ask if these strings really do constitute an alternative to supervaluations. Does the no-gap construal above not re-introduce overdetermination? Not necessarily. The claim that $\boxed{\text{dawn}}\boxed{\text{rain}}\boxed{\text{dusk}}$ $(su, [\![\cdot]\!])$-describes $[0,1)[1,2)[2,3)$ does *not* entail that dawn is not $[\![\cdot]\!]$-true at $[0, 1+\delta)$ for $\delta > 0$, or that the real numbers 0, 1, 2 or 3 are the exact points at which the truth values of certain temporal propositions change. Observations are not exhaustive with respect to truth, so there need not be a hidden assumption here of infinite precision. Furthermore, strings need not be evaluated relative to a set T of times based on the real numbers or a relation *su* that fills in all gaps. Indeed, we might minimize our metaphysical commitments by working *not* with "real" times but with *observation* times, any two of which are discernibly different (Fernando 2006b). What matters (arguably) is not so much what times or events "really" are, but how the events are temporally related and how we recognize events. The strings $\boxed{\text{dawn}}\boxed{\text{rain}}^n\boxed{\text{dusk}}$ (for $n \geq 1$) record observations on the basis of which an event of "rain from dawn to dusk" is recognized.[2] Reasons for forming strings of different lengths $(n+2)$ have already been hinted above, but will be developed at greater length below, as we relate these strings to event structures in the sense of Kamp and Reyle (1993) as well as temporal logics.

1.1 Temporal relations

Event structures in Kamp and Reyle (1993) highlight two binary relations on events, strict temporal precedence $\prec$ and temporal overlap $\bigcirc$. Kamp has shown how to flesh out the notion of time implicit in an event structure through moments, relative to which events stretch over intervals. The relations $\prec$ and $\bigcirc$ can be expressed as disjunctions of Allen's interval relations (e.g. Allen and Ferguson 1994). In Sect. 2, we provide a string-theoretic account of Allen's relations and event structures through suitable constructions on strings (involving superposition, padding, block reduction, inverses and projective limits).

1.2 Types versus instances

Particular events can be related according not only to when they happen but also to what states of affairs they describe. Putting the times at which they happen aside, we might ask when two events are instances of the same type. When does one event contain another? What entailments can we associate with events? We take up these questions in Sect. 3, where we exploit more fully the possibilities afforded by formulating observations as temporal propositions. Rather than appeal to worlds in a Kripke semantics for these propositions, we build strings representing situations that relate different events. Entailments can

[2] We can record rain during parts of dawn and dusk in strings $\boxed{\text{dawn, rain}}\boxed{\text{rain}}^n\boxed{\text{dusk, rain}}$ without requiring rain throughout dawn or dusk. As observations are non-exhaustive, the strings $\boxed{\text{dawn}}\boxed{\text{rain}}^n\boxed{\text{dusk}}$ are not incompatible with these, so I have used them above for simplicity. In any case, I believe what I have to say below applies equally to the more specified choices $\boxed{\text{dawn, rain}}\boxed{\text{rain}}^n\boxed{\text{dusk}}$ or $\boxed{\text{dawn}}\boxed{\text{rain}}^n\boxed{\text{dusk, rain}}$ or $\boxed{\text{dawn, rain}}\boxed{\text{rain}}^n\boxed{\text{dusk, rain}}$.

be read directly off these strings, and can be extended by either increasing string length (described in Table 5 below as "temporal stretch") or adding temporal propositions to the constituent symbols ("descriptive detail" in Table 5). In the concluding section, we discuss some of the wider implications for natural language temporality.

2 Event times related in strings

As concrete particulars, events have temporal projections, the topic of this section. These projections are commonly described as intervals, basic relations between which have been catalogued by James Allen. We reproduce these relations in terms of strings representing event occurrences (Sect. 2.1), and then explain how to extract intervals from these strings by showing that the strings constitute instances of Hans Kamp's event structures (Sect. 2.3).

To isolate the structure relevant to intervals, we shall reduce a string in two ways. These reductions are perhaps most easily motivated by the no-gap construal above of $\fbox{dawn}\fbox{rain}\fbox{dusk}$, but are suited also to the examples of interest in this section. Reducing $\fbox{dawn}\fbox{rain}^n\fbox{dusk}$ for $n \geq 1$ to $\fbox{dawn}\fbox{rain}\fbox{dusk}$, we define the *block compression* bc(s) of an arbitrary string s inductively by

$$\mathsf{bc}(s) \stackrel{\text{def}}{=} \begin{cases} s & \text{if length}(s) \leq 1 \\ \mathsf{bc}(as') & \text{if } s = aas' \\ a\,\mathsf{bc}(a's') & \text{if } s = aa's' \text{ where } a \neq a' \end{cases}.$$

for all symbols a and a'. Compressing a block aa of two a's to one implements the dictum "no time without change" (Kamp and Reyle 1993, p. 674). A second reduction is based on the idea that an empty box $\square$ appearing at the head or tail of a string is uninformative padding. Given a string s, we strip off initial and final $\square$'s as often as they appear, defining

$$\mathsf{unpad}(s) \stackrel{\text{def}}{=} \begin{cases} s & \text{if } s \text{ neither begins nor ends with } \square \\ \mathsf{unpad}(s') & \text{if } s = \square s' \text{ or else if } s = s'\square \end{cases}$$

for any string s. For example,

$$\mathsf{unpad}(\square^n s \square^m) = \mathsf{unpad}(s)$$

for all integers $n, m \geq 0$. Now, we apply the functions bc and unpad in sequence to form the projection $\pi(s)$ of a string s

$$\pi(s) \stackrel{\text{def}}{=} \mathsf{unpad}(\mathsf{bc}(s)) = \mathsf{bc}(\mathsf{unpad}(s))$$

with initial and final $\square$'s deleted, and blocks aa^n compressed to a. For example,

$$\pi(\square^n \fbox{dawn}^i \fbox{rain}^j \fbox{dusk}^k \square^m) = \fbox{dawn}\fbox{rain}\fbox{dusk} \quad (1)$$

for all non-negative integers $n, m \geq 0$ and all positive integers $i, j, k > 0$. A bit more notation will be handy. We write $f^{-1}s$ for the *inverse image of* a string s *under* a function f

$$f^{-1}s \stackrel{\text{def}}{=} \{s' \mid f(s') = s\}$$

and we refer to a set of strings as a language. The *Kleene star* L^* *of* a language L is the smallest set L' containing ϵ such that for all $s \in L$ and $s' \in L'$, $ss' \in L'$. Hence, $\Box^*$ is the set

$$\Box^* = \{\Box^n \mid n \geq 0\}$$

of finite strings of $\Box$, and we can describe example (1) above as

$$\pi^{-1}\boxed{\mathsf{dawn}}\boxed{\mathsf{rain}}\boxed{\mathsf{dusk}} = \Box^*\boxed{\mathsf{dawn}}^+\boxed{\mathsf{rain}}^+\boxed{\mathsf{dusk}}^+\Box^*$$

where $L^+ \stackrel{\text{def}}{=} L^*L$. Throughout this paper, we follow the custom in formal language theory of conflating a string s with the singleton language $\{s\}$ when convenient.

2.1 Allen's interval relations via superposition

Fix two events e and e', and let $o(\mathsf{e})$ and $o(\mathsf{e}')$ be temporal propositions that assert observations of e and e', respectively. Let us consider strings over the alphabet $Pow(\{o(\mathsf{e}), o(\mathsf{e}')\})$ consisting of the four symbols $\Box$, $\boxed{o(\mathsf{e}')}$, $\boxed{o(\mathsf{e}')}$ and $\boxed{o(\mathsf{e}), o(\mathsf{e}')}$. To save on space, let us shorten the last three symbols to $[\mathsf{e}]$, $[\mathsf{e}']$ and $[\mathsf{e}, \mathsf{e}']$, respectively. The string

$$\boxed{o(\mathsf{e}), o(\mathsf{e}')} = [\mathsf{e}, \mathsf{e}']$$

of length 1 describes temporal overlap $\mathsf{e} \bigcirc \mathsf{e}'$ between e and e' insofar as both e and e' are observed at the same box. The string

$$\boxed{o(\mathsf{e})}\boxed{o(\mathsf{e}')} = [\mathsf{e}][\mathsf{e}']$$

"suggests" that e temporally precedes e', $\mathsf{e} \prec \mathsf{e}'$, insofar as e occurs before e' in it. The reason we say "suggests" (rather than "describes," as in the case of $[\mathsf{e}, \mathsf{e}']$) is that the string $[\mathsf{e}][\mathsf{e}']$ might be "part of" the string $[\mathsf{e}, \mathsf{e}'][\mathsf{e}']$ with "subpart" $[\mathsf{e}, \mathsf{e}']$ describing overlap between e and e'. We will make the notions of "part" and "subpart" precise in Sect. 3.2 below, but for now, let us simplify matters by construing strings exhaustively. That is, let us agree that a string $a_1 a_2 \cdots a_n$ over the alphabet $Pow(\{o(\mathsf{e}), o(\mathsf{e}')\})$ *holds* of n successive times $t_1, t_2, \ldots, t_n$ if for $1 \leq i \leq n$,

$$\mathsf{e} \text{ occurs at } t_i \iff o(\mathsf{e}) \in a_i$$

and

$$\text{e}' \text{ occurs at } t_i \iff o(\text{e}') \in a_i.$$

Thus, the occurrences of e form an interval in $a_1 a_2 \cdots a_n$ precisely if

$$\text{whenever } 1 \leq i < j < k \leq n, \quad \text{if } o(\text{e}) \in a_i \text{ and } o(\text{e}) \in k \text{ then } o(\text{e}) \in a_j$$

(and similarly for e′ in place of e). Now, let us consider how e and e′ might be temporally related in a string $a_1 a_2 \cdots a_n$ over $Pow(\{o(\text{e}), o(\text{e}')\})$ assuming e and e′ occur as intervals in $a_1 a_2 \cdots a_n$. If we apply the function π defined above to strings $a_1 a_2 \cdots a_n$ where both e and e′ occur as intervals, it turns out there are 13 different strings we can get, one for each of the 13 Allen interval relations (Allen and Ferguson 1994) (See Table 2).

We can generate the strings in Table 2 naturally through a binary operation on languages, the symbols in which are sets. The operation is called *superposition*, written &, and the alphabet $Pow(\Psi)$ consists of subsets of some fixed set Ψ such as $\{o(\text{e}), o(\text{e}')\}$. Before we define it in general, let us give some examples, recalling that we confuse a string s with the language $\{s\}$ when convenient. On strings $a_1 a_2 \cdots a_n$ and $b_1 b_2 \cdots b_n$ of the same length n, & returns the componentwise union $a_i \cup b_i$ of the strings

$$a_1 a_2 \cdots a_n \;\&\; b_1 b_2 \cdots b_n = (a_1 \cup b_1)(a_2 \cup b_2) \cdots (a_n \cup b_n)$$

so that, for example,

$$\begin{aligned}
[\text{e}, \text{e}'] &= [\text{e}] \;\&\; [\text{e}'] \\
[\text{e}][\text{e}'] &= [\text{e}]\square \;\&\; \square[\text{e}'] \\
[\text{e}][\text{e}, \text{e}'][\text{e}] &= [\text{e}][\text{e}][\text{e}] \;\&\; \square[\text{e}']\square.
\end{aligned}$$

In general, given languages L and L' over the alphabet $Pow(\Psi)$, the *superposition* $L\&L'$ consists of the componentwise union of strings in L and L' of the same length

$$L\&L' \stackrel{\text{def}}{=} \bigcup_{n \geq 0} \{(a_1 \cup b_1) \cdots (a_n \cup b_n) | a_1 \cdots a_n \in L \text{ and } b_1 \cdots b_n \in L'\}$$

Table 2 Allen relations between e and e′ in $s \in \mathcal{L}(\text{e}, \text{e}')$

Allen relation	$\pi(s)$	Allen relation	$\pi(s)$
e before e′	[e]□[e′]	e after e′	[e′]□[e]
e meets e′	[e][e′]	e met-by e′	[e′][e]
e overlaps e′	[e][e, e′][e′]	e overlapped-by e′	[e′][e, e′][e]
e starts e′	[e, e′][e′]	e started-by e′	[e, e′][e]
e during e′	[e′][e, e′][e′]	e contains e′	[e][e, e′][e]
e finishes e′	[e′][e, e′]	e finished-by e′	[e][e, e′]
e equals e′	[e, e′]		

(Fernando 2004). Now, to capture the strings listed in Table 2, let us form the inverse images of [e] and of [e′] under π before superposing. That is, we define the language

$$\mathcal{L}(\mathsf{e},\mathsf{e}') \stackrel{\text{def}}{=} \pi^{-1}[\mathsf{e}] \,\&\, \pi^{-1}[\mathsf{e}'] \\ = \Box^*[\mathsf{e}]^+\Box^* \,\&\, \Box^*[\mathsf{e}']^+\Box^*$$

superposing any positive number of consecutive observations of e with any positive number of consecutive observations of e′, padded to the left and right by □'s. We can partition the language $\mathcal{L}(\mathsf{e},\mathsf{e}')$ into three disjoint sublanguages (where we write + for non-deterministic choice):

(i) $\Box^*[\mathsf{e}]^+\Box^*[\mathsf{e}']^+\Box^*$ in which $\mathsf{e} \prec \mathsf{e}'$
(ii) $\Box^*([\mathsf{e}]^* + [\mathsf{e}']^*)[\mathsf{e},\mathsf{e}']^+([\mathsf{e}]^* + [\mathsf{e}']^*)\Box^*$ in which $\mathsf{e} \bigcirc \mathsf{e}'$, and
(iii) $\Box^*[\mathsf{e}']^+\Box^*[\mathsf{e}]^+\Box^*$ in which $\mathsf{e}' \prec \mathsf{e}$.

Next, if for any function f on strings and any language L, we write $f\langle L\rangle$ for the image $\{f(s) \mid s \in L\}$ of L under f, then it follows that

(i) $\pi\langle\Box^*[\mathsf{e}]^+\Box^*[\mathsf{e}']^+\Box^*\rangle = [\mathsf{e}](\Box + \epsilon)[\mathsf{e}']$
(ii) $\pi\langle\Box^*([\mathsf{e}]^* + [\mathsf{e}']^*)[\mathsf{e},\mathsf{e}']^+([\mathsf{e}]^* + [\mathsf{e}']^*)\Box^*\rangle = ([\mathsf{e}] + [\mathsf{e}'] + \epsilon)[\mathsf{e},\mathsf{e}']([\mathsf{e}] + [\mathsf{e}'] + \epsilon)$
(iii) $\pi\langle\Box^*[\mathsf{e}']^+\Box^*[\mathsf{e}]^+\Box^*\rangle = [\mathsf{e}'](\Box + \epsilon)[\mathsf{e}]$.

For the record,

Theorem 1 $\pi\langle\pi^{-1}[\mathsf{e}] \,\&\, \pi^{-1}[\mathsf{e}']\rangle$ *consists of the thirteen strings*

$$[\mathsf{e}](\Box + \epsilon)[\mathsf{e}'] + ([\mathsf{e}] + [\mathsf{e}'] + \epsilon)[\mathsf{e},\mathsf{e}']([\mathsf{e}] + [\mathsf{e}'] + \epsilon) \\ + [\mathsf{e}'](\Box + \epsilon)[\mathsf{e}]$$

listed in Table 2.

Remark As

$$\pi\langle\pi^{-1}[\mathsf{e}] \,\&\, \pi^{-1}[\mathsf{e}']\rangle \neq \pi\langle\pi^{-1}[\mathsf{e}]\rangle \,\&\, \pi\langle\pi^{-1}[\mathsf{e}']\rangle = [\mathsf{e},\mathsf{e}'],$$

one should be careful to say that the reductions bc and unpad built into π abstract away structure irrelevant to intervals. Some of that structure is essential if we are to capture the Allen relations through superposition. *Some* of that structure, not all. We can form $\pi\langle\mathcal{L}(\mathsf{e},\mathsf{e}')\rangle$ by applying π to

$$\{s \in \pi^{-1}[\mathsf{e}] \mid \mathrm{length}(s) \leq 3\} \,\&\, \{s \in \pi^{-1}[\mathsf{e}'] \mid \mathrm{length}(s) \leq 3\}.$$

Can we make do then with strings of length ≤ 3? Not if we wish to consider more than two events side by side. For n events $\mathsf{e}_1, \ldots, \mathsf{e}_n$, a string in

$$\pi\langle\pi^{-1}[\mathsf{e}_1] \,\&\, \cdots \,\&\, \pi^{-1}[\mathsf{e}_n]\rangle$$

Table 3 Axioms for event structures (Kamp and Reyle 1993, p. 667)

(P_1)	$e < e'$ implies not $e' < e$
(P_2)	$e < e' < e''$ implies $e < e''$
(P_3)	$e \bigcirc e$
(P_4)	$e \bigcirc e'$ implies $e' \bigcirc e$
(P_5)	$e < e'$ implies not $e \bigcirc e'$
(P_6)	$e_1 < e_2 \bigcirc e_3 < e_4$ implies $e_1 < e_4$
(P_7)	$e < e'$ or $e \bigcirc e'$ or $e' < e$

can have up to length $2n - 1$. It is tempting to reduce a block a^k for $k > 1$ to a, if what is important is what is observed (namely a), and not the number (beyond 1) of acts of observation. But the number of acts of observation becomes crucial when a^k is refined differentially by superposition with a string $a_1 \cdots a_k$ of k snapshots, where a_1 or a_k might be $\Box$. Our association of Allen relations with $\pi\langle\mathcal{L}(\mathsf{e}, \mathsf{e}')\rangle$ in Table 2 depends on the understanding that *no* further observations are to be made that bear on e or e'. The structure that π projects away and the complications arising from the partiality of observations are among the main concerns of Sect. 3 below. In the meantime, we will presently embellish the superposition $\pi^{-1}[\mathsf{e}]$ & $\pi^{-1}[\mathsf{e}']$ so as to relate the strings to event structures.

2.2 Moments for Kamp's event structures

An *event structure* $\langle \mathsf{E}, \prec, \bigcirc \rangle$ in Kamp and Reyle (1993) consists of a set E of events and binary relations on E of temporal precedence $\prec$ and temporal overlap $\bigcirc$ satisfying the axioms listed in Table 3.[3] If $\bigcirc$ were equality on E, then (P_1)–(P_7) would say simply that $\prec$ linearly orders E. But as $\bigcirc$ can hold between non-equal events, it will prove useful to collect pairwise overlapping subsets of E in the set

$$O(\bigcirc) \stackrel{\text{def}}{=} \{a \subseteq \mathsf{E} \mid (\forall \mathsf{e}, \mathsf{e}' \in a)\ \mathsf{e} \bigcirc \mathsf{e}'\}$$

and lift $\prec$ to $O(\bigcirc)$ by existential quantification, defining

$$a \prec^{\bigcirc} a' \stackrel{\text{def}}{\Leftrightarrow} (\exists \mathsf{e} \in a)(\exists \mathsf{e}' \in a')\ \mathsf{e} \prec \mathsf{e}'$$

for all $a, a' \in O(\bigcirc)$. The crucial next step is to pick out the $\subseteq$-maximal elements of $O(\bigcirc)$ in

$$M(\bigcirc) \stackrel{\text{def}}{=} \{t \in O(\bigcirc) \mid (\forall a \in O(\bigcirc))t \subseteq a \text{ implies } a = t\}$$

and set

$$\tau(\mathsf{e}) \stackrel{\text{def}}{=} \{t \in M(\bigcirc) \mid \mathsf{e} \in t\}$$

for all $\mathsf{e} \in \mathsf{E}$. It is natural to call the elements of $M(\bigcirc)$ *moments* in view of

[3] The first two postulates are superfluous. (P_1) is derivable from (P_2), (P_3) and (P_5); and (P_2) from (P_3) and (P_6).

Theorem 2 (Kamp) *Given an event structure* $\langle \mathsf{E}, \prec, \bigcirc \rangle$, $\prec^{\bigcirc}$ *linearly orders* $M(\bigcirc)$ *and for every* $\mathsf{e} \in \mathsf{E}$, $\tau(\mathsf{e})$ *is a* $\prec_{\bigcirc}$*-interval of* $M(\bigcirc)$*—i.e.,*

$$(\forall t, t' \in \tau(\mathsf{e}))(\forall t'' \in M(\bigcirc)) \quad t \prec^{\bigcirc} t'' \prec^{\bigcirc} t' \text{ implies } t'' \in \tau(\mathsf{e}).$$

Moreover,

$$\mathsf{e} \prec \mathsf{e}' \iff (\forall t \in \tau(\mathsf{e}))(\forall t' \in \tau(\mathsf{e}')) \quad t \prec^{\bigcirc} t'$$
$$\mathsf{e} \bigcirc \mathsf{e}' \iff (\exists t \in \tau(\mathsf{e})) \quad t \in \tau(\mathsf{e}')$$

for all $\mathsf{e}, \mathsf{e}' \in \mathsf{E}$.

Examples
For $\mathsf{E} = \{\mathsf{e}, \mathsf{e}'\}$, there are three event structures

(i) $\prec_1 = \{(\mathsf{e}, \mathsf{e}')\}$ and $\bigcirc_1 = \emptyset$
(ii) $\prec_2 = \emptyset$ and $\bigcirc_2 = \mathsf{E} \times \mathsf{E}$
(iii) $\prec_3 = \{(\mathsf{e}', \mathsf{e})\}$ and $\bigcirc_3 = \emptyset$

with

(i) $M(\bigcirc_1) = \{\{\mathsf{e}\}, \{\mathsf{e}'\}\}$ and $\prec_1^{\bigcirc_1} = \{(\{\mathsf{e}\}, \{\mathsf{e}'\})\}$
(ii) $M(\bigcirc_2) = \{\mathsf{E}\}$ and $\prec_2^{\bigcirc_2} = \emptyset$
(iii) $M(\bigcirc_3) = \{\{\mathsf{e}\}, \{\mathsf{e}'\}\}$ and $\prec_3^{\bigcirc_3} = \{(\{\mathsf{e}'\}, \{\mathsf{e}\})\}$

corresponding to the strings (i) $[\mathsf{e}][\mathsf{e}']$, (ii) $[\mathsf{e}, \mathsf{e}']$ and (iii) $[\mathsf{e}'][\mathsf{e}]$. The obvious question is what about the 10 other strings in Table 2? These ten strings collapse into one of the three once we delete boxes *not* in $M(\bigcirc)$, namely, $\Box$ and in case $[\mathsf{e}, \mathsf{e}'] \in \mathsf{M}(\bigcirc)$, $[\mathsf{e}]$ and $[\mathsf{e}']$.

More concretely, take the following instance of (ii). Suppose e and e' were the half-open real intervals $[0, 2)$ and $[1, 3)$ respectively so that

(†) e but not e' occurs at $[0, 1)$, both e and e' occur at $[1, 2)$, and e' but not e occurs at $[2, 3)$.

To capture the real numbers and intervals in (†) through $M(\bigcirc)$ and $\prec^{\bigcirc}$, we must enrich the event structure $\langle \{\mathsf{e}, \mathsf{e}'\}, \emptyset, \{\mathsf{e}, \mathsf{e}'\} \times \{\mathsf{e}, \mathsf{e}'\} \rangle$ considerably. We will see how in Sect. 2.3. For now, let us focus on the string representation

$$[\mathsf{e}][\mathsf{e}, \mathsf{e}'][\mathsf{e}'] = \boxed{o(\mathsf{e})}\boxed{o(\mathsf{e}), o(\mathsf{e}')}\boxed{o(\mathsf{e}')}$$

of (†). To get around the requirement of maximality in $M(\bigcirc)$, it suffices to add events e'_l (for the left of e') and e_r (for the right of e) to $\{\mathsf{e}, \mathsf{e}'\}$ for an event structure $\langle \{\mathsf{e}, \mathsf{e}', \mathsf{e}_r, \mathsf{e}'_l\}, \prec_+, \bigcirc_+ \rangle$ where $\prec_+$ and $\bigcirc_+$ extend $\prec$ and $\bigcirc$ so that we can picture $\prec_+{}^{\bigcirc_+}$ on $M(\bigcirc_+)$ as

$$[\mathsf{e}, \mathsf{e}'_l][\mathsf{e}, \mathsf{e}'][\mathsf{e}_r, \mathsf{e}'] = \boxed{o(\mathsf{e}), o(\mathsf{e}'_l)}\boxed{o(\mathsf{e}), o(\mathsf{e}')}\boxed{o(\mathsf{e}_r), o(\mathsf{e}')}.$$

In particular, $\mathsf{e} \bigcirc_+ \mathsf{e}'_l \prec_+ \mathsf{e}'$ and $\mathsf{e} \prec_+ \mathsf{e}_r \bigcirc_+ \mathsf{e}'$. We can extract $[\mathsf{e}][\mathsf{e},\mathsf{e}'][\mathsf{e}']$ from $[\mathsf{e},\mathsf{e}'_l][\mathsf{e},\mathsf{e}'][\mathsf{e}_r,\mathsf{e}']$ by applying the function r that maps a string $a_1a_2\cdots a_n$ to its restriction

$$\mathsf{r}(a_1a_2\cdots a_n) \stackrel{\text{def}}{=} (a_1 \cap [\mathsf{e},\mathsf{e}'])(a_2 \cap [\mathsf{e},\mathsf{e}'])\cdots(a_n \cap [\mathsf{e},\mathsf{e}'])$$

to $[\mathsf{e},\mathsf{e}'] = \{o(\mathsf{e}), o(\mathsf{e}')\}$.

The general idea is that an event e marks not only the time at which it occurs but in case it is bounded to the left, its past e_l, and in case it is bounded to the right, its future e_r.[4] This suggests refining the superposition

$$\pi^{-1}[\mathsf{e}] \;\&\; \pi^{-1}[\mathsf{e}'] = \Box^*[\mathsf{e}]^+\Box^* \;\&\; \Box^*[\mathsf{e}']^+\Box^*$$

to

$$\mathcal{L}^l_r(\mathsf{e},\mathsf{e}') \stackrel{\text{def}}{=} [\mathsf{e}_l]^*[\mathsf{e}]^+[\mathsf{e}_r]^* \;\&\; [\mathsf{e}'_l]^*[\mathsf{e}']^+[\mathsf{e}'_r]^*$$

(with the offending non-maximal set $\Box$ filled in), from which we can derive all thirteen strings in Table 2

$$\pi\langle\pi^{-1}[\mathsf{e}] \;\&\; \pi^{-1}[\mathsf{e}']\rangle = \pi\langle\mathsf{r}\langle\mathcal{L}^{\mathsf{l}}_r(\mathsf{e},\mathsf{e}')\rangle\rangle.$$

The composition $\pi \circ \mathsf{r}$ of functions π and r will be crucial to our construction of event structures as projective limits of strings.

2.3 Event structures as projective limits

For the remainder of this section, we fix a set E of events, and drop the distinction between an event $\mathsf{e} \in \mathsf{E}$ and a temporal proposition $o(\mathsf{e})$ asserting an observation of e. This simplifies the notation, reducing the symbols from which we form strings to subsets of E. Given a set $X \subseteq \mathsf{E}$ of events and a string $s = a_1a_2\cdots a_n \in Pow(\mathsf{E})^*$, we write $\mathsf{r}_X(s)$ for the restriction

$$\mathsf{r}_{\mathsf{X}}(a_1a_2\cdots a_n) \stackrel{\text{def}}{=} (a_1 \cap \mathrm{X})(a_2 \cap \mathrm{X})\cdots(a_n \cap X)$$

to X, and let $\pi_X(s)$ be the result of applying π to $\mathsf{r}_X(s)$

$$\pi_X(s) \stackrel{\text{def}}{=} \pi(\mathsf{r}_X(s)).$$

Notice that

$$\text{whenever } Y \subseteq X, \quad \pi_Y(s) = \pi_Y(\pi_X(s))$$

[4] For an event e bounded to the left and right, the triple $\langle \mathsf{e}_l, \mathsf{e}, \mathsf{e}_r\rangle$ is essentially what chapter 2 of van Lambalgen and Hamm (2005) calls a Walker instant (with e understood to have non-empty temporal extent).

which is to say that the maps $\pi_X : Pow(\mathsf{E})^* \to Pow(X)^*$ induce a projective/inverse limit over the set $Fin(\mathsf{E})$ of finite subsets X of E. More specifically, we define an E-*point* to be a function $\mathsf{p} : Fin(\mathsf{E}) \to \bigcup_{X \in Fin(\mathsf{E})} Pow(X)^*$ such that

$$(\forall X \in Fin(\mathsf{E}))\ (\forall Y \subseteq X) \quad \mathsf{p}(Y) = \pi_Y(\mathsf{p}(X)) \in Pow(Y)^*.$$

For each $s \in Pow(\mathsf{E})^*$, the map p_s sending a finite subset X of E to $\pi_X(s)$ is an E-point. Going beyond these points, we can represent the real line $\langle \Re, < \rangle$ by an E-point $\mathsf{p}^{\Re}$ if $\mathsf{E} \supseteq \Re$ contains the real numbers and

$$\mathsf{p}^{\Re}(\{r_1, r_2, \ldots, r_n\}) \stackrel{\text{def}}{=} \boxed{r_1}\Box\boxed{r_2}\Box \cdots \Box\boxed{r_n}$$

for any finite sequence $r_1 < r_2 < \cdots < r_n$ of real numbers.[5] Returning to the example in the previous subsection of $\mathsf{e} = [0, 2)$ and $\mathsf{e}' = [1, 3)$, we can express

(†) e but not e' occurs at $[0, 1)$, both e and e' occur at $[1, 2)$, and e' but not e occurs at $[2, 3)$.

through the string

$$\boxed{0, \mathsf{e}}\boxed{\mathsf{e}}\boxed{1, \mathsf{e}, \mathsf{e}'}\boxed{\mathsf{e}, \mathsf{e}'}\boxed{2, \mathsf{e}'}\boxed{3}$$

of length 6 over the finite set $\{0, 1, 2, 3, \mathsf{e}, \mathsf{e}'\}$ of events.

In general, given an E-point p, let $\langle \mathsf{E}^{\mathsf{p}}, <^{\mathsf{p}}, \bigcirc^{\mathsf{p}} \rangle$ be the triple consisting of the set

$$\mathsf{Ep} \stackrel{\text{def}}{=} \{\mathsf{e} \in \mathsf{E} \mid \mathsf{p}(\{\mathsf{e}\}) = \boxed{\mathsf{e}}\}$$

of events e that p treats as an interval, and binary relations $<^{\mathsf{p}}$ and $\bigcirc^{\mathsf{p}}$ on E^{p} such that (in accordance with Table 2)

$$\mathsf{e} <^{\mathsf{p}} \mathsf{e}' \stackrel{\text{def}}{\iff} \mathsf{p}(\{\mathsf{e}, \mathsf{e}'\}) \in \boxed{\mathsf{e}}\boxed{\mathsf{e}'} + \boxed{\mathsf{e}}\Box\boxed{\mathsf{e}'}$$

$$\mathsf{e} \bigcirc^{\mathsf{p}} \mathsf{e}' \stackrel{\text{def}}{\iff} \mathsf{p}(\{\mathsf{e}, \mathsf{e}'\}) \in (\boxed{\mathsf{e}} + \boxed{\mathsf{e}'} + \epsilon)\boxed{\mathsf{e}, \mathsf{e}'}(\boxed{\mathsf{e}} + \boxed{\mathsf{e}'} + \epsilon)$$

for all $\mathsf{e}, \mathsf{e}' \in \mathsf{E}$.

Proposition 3 *For every* E-*point* p, $\langle \mathsf{E}^{\mathsf{p}}, <^{\mathsf{p}}, \bigcirc^{\mathsf{p}} \rangle$ *is an event structure. Furthermore, for all* $\mathsf{e}, \mathsf{e}' \in \mathsf{E}$,

$$(\exists \mathsf{e}_1 <^{\mathsf{p}} \mathsf{e}')(\exists \mathsf{e}_2 \bigcirc^{\mathsf{p}} \mathsf{e}_1)\ \mathsf{e} <^{\mathsf{p}} \mathsf{e}_2 \quad \textit{implies} \quad \mathsf{p}(\{\mathsf{e}, \mathsf{e}'\}) = \boxed{\mathsf{e}}\Box\boxed{\mathsf{e}'}$$

[5] Whether a full system of observations can be devised for arbitrarily close real numbers $r < r'$ is (as noted in the introduction) questionable in practice, if not in principle. It is not clear to me, in any case, that natural language semantics depends on such a system (pace van Lambalgen and Hamm 2005).

and whenever $e \bigcirc^p e'$,

$$(\exists e_1 <^p e')\ e \bigcirc^p e_1 \quad \textit{implies} \quad p(\{e,e'\}) \in \boxed{e}\boxed{e,e'}(\boxed{e} + \boxed{e'} + \epsilon)$$
$$(\exists e_1 \bigcirc^p e)\ e' <^p e_1 \quad \textit{implies} \quad p(\{e,e'\}) \in (\boxed{e} + \boxed{e'} + \epsilon)\boxed{e,e'}\boxed{e}.$$

What about the converses of the implications in Proposition 3? Let us build these conditions into the following definition. An E-point p *represents* an event structure $\langle E, \prec, \bigcirc \rangle$ if $\langle E^p, <^p, \bigcirc^p \rangle = \langle E, \prec, \bigcirc \rangle$ and

$$p(\{e,e'\}) = \boxed{e}\Box\boxed{e'} \iff (\exists e'' \prec e') (\exists e''' \bigcirc e'')\ e \prec e''' \tag{2}$$

$$p(\{e,e'\}) \in \boxed{e}\boxed{e,e'}(\boxed{e} + \boxed{e'} + \epsilon) \iff e \bigcirc e' \text{ and } (\exists e'' \prec e')\ e \bigcirc e'' \tag{3}$$

$$p(\{e,e'\}) \in (\boxed{e} + \boxed{e'} + \epsilon)\boxed{e,e'}\boxed{e} \iff e \bigcirc e' \text{ and } (\exists e'' \bigcirc e)\ e' \prec e'' \tag{4}$$

for all $e, e' \in E$. The remainder of this section is devoted to establishing

Theorem 4 *For every event structure* $\langle E, \prec, \bigcirc \rangle$, *there is a unique* E*-point* p *that represents it.*

To prove Theorem 4, let us fix an event structure $\langle E, \prec, \bigcirc \rangle$. We extend it to an event structure $\langle E_+, \prec_+, \bigcirc_+ \rangle$ with canonical witnesses to the existential quantification in the equivalences (2)–(4) as follows. Let

$$E_+ \stackrel{\text{def}}{=} E \ \cup \ \{e_l \mid e \in E \text{ and } (\exists e')\ e' \prec e\} \ \cup \ \{e_r \mid e \in E \text{ and } (\exists e')\ e \prec e'\}$$

where e, e'_l and e''_r are distinct for all $e, e', e'' \in E$. The plan is to extend $\bigcirc$ to $\bigcirc_+$ so that the equivalences (2)–(4) are met by

$$p(\{e,e'\}) = \boxed{e}\Box\boxed{e'} \iff e'_l \bigcirc_+ e_r \tag{5}$$

$$p(\{e,e'\}) \in \boxed{e}\boxed{e,e'}(\boxed{e} + \boxed{e'} + \epsilon) \iff e \bigcirc e' \text{ and } e \bigcirc_+ e'_l \tag{6}$$

$$p(\{e,e'\}) \in (\boxed{e} + \boxed{e'} + \epsilon)\boxed{e,e'}\boxed{e} \iff e \bigcirc e' \text{ and } e'_r \bigcirc_+ e \tag{7}$$

respectively. Accordingly, we set

$$\mathsf{e}'_l \bigcirc_+ \mathsf{e}_r \overset{\text{def}}{\iff} (\exists \mathsf{e}'' \prec \mathsf{e}')(\exists \mathsf{e}''' \succ \mathsf{e})\ \mathsf{e}'' \bigcirc \mathsf{e}'''$$
$$\mathsf{e} \bigcirc_+ \mathsf{e}'_l \overset{\text{def}}{\iff} (\exists \mathsf{e}'' \prec \mathsf{e}')\ \mathsf{e} \bigcirc \mathsf{e}''$$
$$\mathsf{e}'_r \bigcirc_+ \mathsf{e} \overset{\text{def}}{\iff} (\exists \mathsf{e}'' \succ \mathsf{e}')\ \mathsf{e}'' \bigcirc \mathsf{e}$$

and for symmetry,

$$\mathsf{e}_r \bigcirc_+ \mathsf{e}'_l \overset{\text{def}}{\iff} \mathsf{e}'_l \bigcirc_+ \mathsf{e}_r$$
$$\mathsf{e}'_l \bigcirc_+ \mathsf{e} \overset{\text{def}}{\iff} \mathsf{e} \bigcirc_+ \mathsf{e}'_l$$
$$\mathsf{e} \bigcirc_+ \mathsf{e}'_r \overset{\text{def}}{\iff} \mathsf{e}'_r \bigcirc_+ \mathsf{e}.$$

The remaining instances of $\bigcirc_+$ beyond $\bigcirc$ are given by

$$\mathsf{e}_l \bigcirc_+ \mathsf{e}'_l \overset{\text{def}}{\iff} (\exists \mathsf{e}'' \prec \mathsf{e})(\exists \mathsf{e}''' \prec \mathsf{e}')\ \mathsf{e}'' \bigcirc \mathsf{e}'''$$

and

$$\mathsf{e}_r \bigcirc_+ \mathsf{e}'_r \overset{\text{def}}{\iff} (\exists \mathsf{e}'' \succ \mathsf{e})(\exists \mathsf{e}''' \succ \mathsf{e}')\ \mathsf{e}'' \bigcirc \mathsf{e}'''.$$

To ensure $\langle \mathsf{E}_+, \prec_+, \bigcirc_+ \rangle$ is an event structure, we arrange

$$x \bigcirc_+ x' \iff \text{ neither } x \prec_+ x' \text{ nor } x' \prec_+ x$$

or equivalently,

$$x \prec_+ x' \iff \text{ neither } x \bigcirc_+ x' \text{ nor } x' \prec_+ x$$

(for all $x, y \in \mathsf{E}_+$), setting $\prec_+$ to

$$\begin{aligned} \prec \quad & \cup \{(\mathsf{e}_l, \mathsf{e}'_r) \mid (\exists \mathsf{e}'' \prec \mathsf{e})(\exists \mathsf{e}''')\ \mathsf{e}' \prec \mathsf{e}''' \text{ and not } \mathsf{e}_l \bigcirc_+ \mathsf{e}'_r\} \\ & \cup \{(\mathsf{e}, \mathsf{e}'_r) \mid (\exists \mathsf{e}'')\ \mathsf{e}' \prec \mathsf{e}'' \text{ and not } \mathsf{e} \bigcirc_+ \mathsf{e}'_r\} \\ & \cup \{(\mathsf{e}_l, \mathsf{e}') \mid (\exists \mathsf{e}'')\ \mathsf{e}'' \prec \mathsf{e} \text{ and not } \mathsf{e}_l \bigcirc_+ \mathsf{e}'\}. \end{aligned}$$

For every non-empty subset X of E, let X_+ be the part of E_+ carved out by X

$$X_+ \overset{\text{def}}{=} X \cup \{\mathsf{e}_l \mid \mathsf{e} \in X \text{ and } (\exists \mathsf{e}')\ \mathsf{e}' \prec \mathsf{e}\} \cup \{\mathsf{e}_r \mid \mathsf{e} \in X \text{ and } (\exists \mathsf{e}')\ \mathsf{e} \prec \mathsf{e}'\}$$

and let $\prec_+^X$ and $\bigcirc_+^X$ be the restrictions to X_+ of $\prec_+$ and $\bigcirc_+$

$$\prec_+^X \overset{\text{def}}{=} \prec_+ \cap (X_+ \times X_+)$$
$$\bigcirc_+^X \overset{\text{def}}{=} \bigcirc_+ \cap (X_+ \times X_+).$$

For example, $\prec_+^{\mathsf{E}}$ is $\prec_+$, and $\bigcirc_+^{\mathsf{E}}$ is $\bigcirc_+$. But it will be useful below also to consider $X \in Fin(\mathsf{E})$, in view of

Lemma 5 *If* $\langle \mathsf{E}, \prec, \bigcirc \rangle$ *is an event structure and* X *is a non-empty subset of* E, *then* $\langle X_+, \prec_+^X, \bigcirc_+^X \rangle$ *is an event structure.*

Next, we apply Kamp's construction of time (described in Sect. 2.2) to the event structure $\langle X_+, \prec_+^X, \bigcirc_+^X \rangle$, picking out

(a) the $\bigcirc_+^X$-overlapping sets

$$O_X \stackrel{\text{def}}{=} \{a \subseteq X_+ \mid (\forall \mathsf{e}, \mathsf{e}' \in a)\ \mathsf{e} \bigcirc_+^X \mathsf{e}'\}$$

(written $O(\bigcirc_+^X)$ under the conventions of Sect. 2.2)

(b) the moments

$$M_X \stackrel{\text{def}}{=} \{a \in O_X \mid (\forall a' \in O_X)\ a \subseteq a' \text{ implies } a = a'\}$$

(in full, $M(\bigcirc_+^X)$), and

(c) the relation $\prec_X$ with

$$a \prec_X a' \stackrel{\text{def}}{\iff} (\exists x \in a)(\exists x' \in a')\ x \prec_+^X x'$$

for $a, a' \in O_X$.

For finite X, we can appeal to Lemma 5 and Theorem 2 for an enumeration

$$M_X = \{a_1^X, a_2^X, \ldots, a_n^X\}$$

of M_X that is $\prec_X$-increasing

$$a_1^X \prec_X a_2^X \prec_X \cdots \prec_X a_n^X$$

(dropping the subscript X on n for simplicity). Generalizing over $X \in Fin(\mathsf{E})$, we form the map $\hat{\mathsf{p}} : Fin(\mathsf{E}) \to \bigcup_{X \in Fin(\mathsf{E})} Pow(X)^*$ such that $\hat{\mathsf{p}}(\emptyset) = \epsilon$ and for non-empty $X \in Fin(\mathsf{E})$,

$$\hat{\mathsf{p}}(X) \stackrel{\text{def}}{=} \pi_X(a_1^X a_2^X \cdots a_n^X).$$

An argument by induction on the cardinality of finite subsets of E shows that the map $\hat{\mathsf{p}}$ is the unique E-point representing $\langle \mathsf{E}, \prec, \bigcirc \rangle$. So much for Theorem 4.

3 From concrete particulars to types

Having dwelt in the previous section on temporal instantiations of events, we now abstract away from concrete particulars to consider the types exemplified by these occurrences. The distinction here between particulars and types can be explained by analogy with Kripke semantics for modal logic. Recall that a *Kripke model* $\langle N, R, V\rangle$ for a set P of atomic propositions consists of

(a) a *frame* $\langle N, R\rangle$ given by a set N of *nodes* and an *accessibility relation* $R \subseteq N \times N$ on N, and
(b) a *valuation* V: $P \to Pow(N)$ from P to the family of subsets of N.

Given a valuation V, an atomic proposition $p \in P$ can be viewed as a type $V(p) \subseteq N$ of nodes where p is interpreted to be true. Add a binary relation R on N and we can classify further subsets of N through a set $\Phi \supseteq P$ of temporal propositions φ with $[\![\varphi]\!] \subseteq N$ (and $[\![p]\!] = V(p)$ for $p \in P$). But what does this all have to do with events?

The idea is summarized in Table 4, where Kripke frames and event structures are viewed as concrete particulars, in contrast to types given by valuations and languages over the alphabet $Pow(\Phi)$. More precisely, given an event structure $\langle \mathsf{E}, \prec, \bigcirc\rangle$, Theorem 2 says we can extract a linear order $\prec^{\bigcirc}$ on a set $M(\bigcirc)$ of moments such that if we map an arbitrary event $\mathsf{e} \in \mathsf{E}$ to an atomic proposition $o(\mathsf{e})$ that we in turn interpret as the $\prec^{\bigcirc}$-interval

$$\hat{V}(o(\mathsf{e})) \stackrel{\text{def}}{=} \{t \in M(\bigcirc) \mid \mathsf{e} \in t\}$$

then we obtain a Kripke model $\langle M(\bigcirc), \prec^{\bigcirc}, \hat{V}\rangle$ for $\{o(\mathsf{e}) \mid \mathsf{e} \in \mathsf{E}\}$. In Sect. 2.3, we suppressed the distinction between $o(\mathsf{e})$ and e to lighten the notation. We now restore that distinction, but mainly in order to shift the focus from the particulars $\mathsf{e} \in \mathsf{E}$ to the temporal propositions $\varphi \in \Phi$. Henceforth, we adopt the term *fluent* for temporal proposition, following the custom in AI since McCarthy and hayes (1969) (as well as more recent work such as van Lambalgen and Hamm 2005). That is, we refer hereafter to the elements of Φ as fluents.

Deriving a fluent $o(\mathsf{e}) \in \Phi$ from an event e that stretches over an interval raises the possibility of interpreting fluents over intervals. That is, instead of the frame $\langle M(\bigcirc), \prec^{\bigcirc}\rangle$, why not build a Kripke model over $\langle \mathsf{E}, \prec, \bigcirc\rangle$ or perhaps $\langle O(\bigcirc) - \{\emptyset\}, \prec^{\bigcirc}, \hat{\bigcirc}\rangle$ where

$$a \;\hat{\bigcirc}\; a' \stackrel{\text{def}}{\iff} a \cap a' \neq \emptyset$$

for all $a, a' \in O(\bigcirc)$? In Venema (1990), for instance, fluents are interpreted over closed intervals of a strict partial order. Here, we take a different approach. Instead of interval-based fluents, we assemble finite strings $a_1 a_2 \cdots a_n \in Fin(\Phi)^*$

Table 4 Concrete particulars *versus* information units

	Particulars	Types (information units)
Kripke model	Kripke frame $\langle N, R\rangle$	Valuation $V: P \to Pow(N)$
Here	Event structure $\langle \mathsf{E}, \prec, \bigcirc\rangle$	Languages over $Fin(\Phi)$

of finite sets $a_i \subseteq \Phi$ of fluents that are interpreted in the usual (point-based) way over linear orders. To link up with interval-based event structures $\langle \mathsf{E}, \prec, \bigcirc \rangle$, we appeal to Kamp's construction of moments (Sect. 2.2) and projective limits (Sect. 2.3).

Implicit in the identification of e with $o(\mathsf{e})$ in Sect. 2.3 is the assumption that the fluent $o(\mathsf{e})$ is used to describe only the event e. Generalizing from instances to types, we work in this section also with fluents such as dawn that may occur at scattered (not necessarily successive) positions in a string $a_1 a_2 \cdots a_n \in Fin(\Phi)^*$, forming part of a description of a variety of event types, anyone of which may have several instances. (It may rain at dawn one day, be clear the next, and then rain again the following dawn.) A string $s \in Fin(\Phi)^*$ represents not so much an event instance as an event type (with any number of instances); instances are represented by string occurrences. For example, the string $s\hat{s}s$ containing two occurrences of s represents an event type formed by two instances of the type represented by s, separated by a string $\hat{s}$ of observations.

In Sect. 3.1, we provide examples of fluents and event types they describe. The event types can be refined incrementally; we show how in Sect. 3.2, building up to situations.

3.1 Some fluents and event types

We start with the simple case of an interval described by $o(\mathsf{e})$. Over a finite linear order $<$ (such as the X-components of the projective limit in Sect. 2.3), a non-empty interval I has $<$-least and $<$-greatest elements t_I and t'_I, which is to say it is closed

$$I = \{x \mid t_I \leq x \leq t'_I\}$$

(where $\leq$ is the union of $<$ with $=$). Suppose we name t_I and t'_I by fluents b_I and e_I respectively, requiring of a Kripke model $\langle N, R, V \rangle$ that

$$V(b_I) = \{t_I\} \text{ and } V(e_I) = \{t'_I\}$$

(so that b_I and e_I are *nominals* in the sense of Hybrid Logic (Blackburn 2000)). Suppose also that P is the past operator and F is the future operator; that is, over a Kripke model $\langle N, R, V \rangle$, $\mathsf{P}\varphi$ is true iff φ is true at an R-earlier time

$$x \models \mathsf{P}\varphi \iff (\exists y \mathsf{R} x)\ y \models \varphi$$

and $\mathsf{F}\varphi$ is true iff φ is true at an R-later time

$$x \models \mathsf{F}\varphi \iff (\exists y \mathsf{R}^{-1} x)\ y \models \varphi.$$

We can then express $o(\mathsf{e})$ in terms of b_I and e_I as the fluent

$$b_I \vee e_I \vee (\mathsf{P} b_I \wedge \mathsf{F} e_I)$$

where, as usual, $\wedge$ is conjunction and $\vee$ disjunction

$$x \models \varphi \wedge \psi \iff x \models \varphi \text{ and } x \models \psi$$
$$x \models \varphi \vee \psi \iff x \models \varphi \text{ or } x \models \psi$$

for all fluents φ, ψ. To picture e bounded to the left and right, we have a choice

$$\boxed{\mathsf{F}o(\mathsf{e})}\boxed{o(\mathsf{e})}^{+}\boxed{\mathsf{P}o(\mathsf{e})} = \left\{\boxed{\mathsf{F}o(\mathsf{e})}\boxed{o(\mathsf{e})}^{n}\boxed{\mathsf{P}o(\mathsf{e})} \mid n \geq 1\right\}$$

of strings sandwiching $o(\mathsf{e})$ by $\mathsf{F}o(\mathsf{e})$ and $\mathsf{P}o(\mathsf{e})$ just as in the previous section, e_l and e_r delimit e. We can strip off the first and last positions of these strings (reducing their lengths by 2) if we peer inside $o(\mathsf{e})$ and unwind the disjunction $b_I \vee e_I \vee (\mathsf{P}b_I \wedge \mathsf{F}e_I)$, transforming $\boxed{\mathsf{F}o(\mathsf{e})}\boxed{o(\mathsf{e})}^{+}\boxed{\mathsf{P}o(\mathsf{e})}$ to

$$\boxed{b_I, e_I} + \boxed{b_I}\square^{*}\boxed{e_I}. \tag{8}$$

To generalize from a particular interval I to multiple occurrences of a type, let us abstract away the subscript I on b_I and e_I for fluents b and e that may have any number of scattered occurrences over time (e.g. dawn, rain). Writing ψ for e, and $\neg\psi$ for the negation of ψ,

$$x \models \neg\psi \iff \text{ not } x \models \psi,$$

we have two extreme examples:

Case 1 $b = \psi$, turning (8) to $\boxed{\psi} + \boxed{\psi}\square^{*}\boxed{\psi}$, and

Case 2 $b = \neg\psi$, turning (8) to $\boxed{\neg\psi, \psi} + \boxed{\neg\psi}\square^{*}\boxed{\psi}$ which reduces to $\boxed{\neg\psi}\square^{*}\boxed{\psi}$ after we discard the spurious possibility $\boxed{\neg\psi, \psi}$.

In Case 1, we might expect that ψ holds from start to finish and accordingly transform $\boxed{\psi} + \boxed{\psi}\square^{*}\boxed{\psi}$ to $\boxed{\psi}^{+}$ on the basis of the inertial principle that

(‡) ψ persists unless a force is applied on it

(and no force is expected unless specifically mentioned). Applying (‡) to the instance of non-persistence in Case 2, we must postulate a force on ψ to explain the transition from $\neg\psi$ to ψ. One way to express this is through a fluent $\mathsf{f}\psi$ saying a force is applied to ψ, which we add to the string $\boxed{\neg\psi}\boxed{\psi}$ of length two to get

$$\boxed{\neg\psi, \mathsf{f}\psi}\boxed{\psi}.$$

What about the other strings (in $\boxed{\neg\psi}\square^{+}\boxed{\psi}$) for Case 2? Suppose we can measure the degree to which ψ holds by an element of (say) the unit interval $[0, 1]$, and can form a fluent ψ-deg(x), read "ψ holds to the degree x," with

$$\psi \quad \text{equivalent to} \quad \psi\text{-deg}(1).$$

Let $\psi_\uparrow$ be the fluent

$$(\exists x < 1)\ \psi\text{-deg}(x)\ \wedge\ (\exists y < x)\ \mathsf{previous}(\psi\text{-deg}(y))$$

saying that the degree of ψ is short of 1 but is greater than at the previous time.[6] We transform the strings in $\boxed{\neg\psi}\Box^{+}\boxed{\psi}$ to the strings in

$$\boxed{\psi\text{- deg}(0)}\boxed{\psi_\uparrow}^{+}\boxed{\psi} \tag{9}$$

to describe a monotone incremental transition from ψ-deg(0) to ψ. The change described in (9) is idealized in van Lambalgen and Hamm (2005) through a trajectory predicate over $[0, 1]$, concerning which van Lambalgen and Hamm acknowledge that

> One cannot simply assume that we have a dense set of events in memory to derive from this that cognitive (and not just physical) time may be assumed to be continuous. It is much more reasonable to assume that density arises in the limit of adding more and more events, and that, at each stage, memory contains only finitely many events. (van Lambalgen and Hamm 2006, p. 12)

Finite strings arguably reflect the selectivity and partiality of cognition more faithfully than the use of real numbers. At any rate, there are two transitions in (9) that (according to (‡)) call for forces: the transition from ψ-deg(0) to $\psi_\uparrow$ and the transition from $\psi_\uparrow$ to ψ. Adding these forces to (9), we get

$$\boxed{\psi\text{-deg}(0), \mathsf{f}\psi_\uparrow}\boxed{\psi_\uparrow}^{*}\boxed{\psi_\uparrow, \mathsf{f}\psi}\boxed{\psi}$$

which we can obtain from the superposition

$$\underbrace{\boxed{\psi\text{-deg}(0), \mathsf{f}\psi_\uparrow}\boxed{\psi_\uparrow^{+}}}_{\text{activity}}\Box\ \&\ \Box^{+}\underbrace{\boxed{\psi_\uparrow, \mathsf{f}\psi}\boxed{\psi}}_{\text{achievement}}.$$

In terms of the well-known Vendler classes (Vendler 1967), the forces behind $\mathsf{f}\psi_\uparrow$ and $\mathsf{f}\psi$ correspond *roughly* to an activity and an achievement that, as suggested in Dowty (1979), combine to form an accomplishment. The correspondences here are rough as particular examples of accomplishments, such as *Pat walk from Dublin to Belfast*, might (or might not) be analyzed by variants of (9), with say, ψ-deg(0) replaced by $in(Pat, Dublin)$, $\psi_\uparrow$ by $walk(Pat)$, and ψ by $in(Pat, Belfast)$. Even then, however, we would have the superposition

---

[6] That is, given a Kripke model $\langle N, R, V\rangle$ and $x \in N$,

$$x \models \mathsf{previous}(\varphi) \iff y \models \varphi \text{ for some } yR_\circ x$$

where $R_\circ$ is the successor subrelation of R

$$yR_\circ x \overset{\text{def}}{\iff} yRx \text{ and } (\forall zRx)\ z = y \text{ or } zRy.$$

$$\boxed{in(Pat, Dublin), \mathsf{f}walk(Pat)}\boxed{walk(Pat)}^{+}\Box \;\&$$
$$\Box^{+}\boxed{walk(Pat), \mathsf{f}in(Pat, Belfast)}\boxed{in(Pat, Belfast)}$$

of an activity (concatenated with $\Box$) and an achievement (prefixed by $\Box^{+}$). By contrast, Case 1 represents a state insofar as no force is applied to ψ, and so the inertial principle (‡) yields $\boxed{\psi}^{+}$. We shall see how to make (‡) precise shortly.

3.2 Refinements by constraints and entailments

An event type given by a language $L \subseteq Fin(\Phi)^{*}$ over finite sets of fluents must often be refined to another $L' \subseteq Fin(\Phi)^{*}$. One way to make the notion of refinement precise is through the notion of subsumption $\trianglerighteq$. We say L' *subsumes* L and write $L' \trianglerighteq L$ if the superposition of L and L' contains L'

$$L' \trianglerighteq L \overset{\text{def}}{\iff} L' \subseteq L\&L'$$

(Fernando 2004). Over strings s identified with languages $\{s\}$, $\trianglerighteq$ reduces to componentwise containment between strings of the same length

$$a_1 \cdots a_n \trianglerighteq b_1 \cdots b_m \iff n = m \text{ and } b_i \subseteq a_i \text{ for } 1 \leq i \leq n$$

so that over languages, $\trianglerighteq$ holds whenever every string in the first subsumes one in the second

$$L' \trianglerighteq L \iff (\forall s' \in L')(\exists s \in L)\; s' \trianglerighteq s.$$

The intuition is that a language L is essentially a type with instances $s \in L$. In other words, the information in L amounts to a disjunction $\bigvee_{s \in L} s$ of conjunctions s. Under this construal, $L' \trianglerighteq L$ roughly means L' is at least as informative as L. A notion of containment between strings orthogonal to $\trianglerighteq$ is the following. A *factor of* s is a string s' such that $s = us'v$ for some (possibly empty) strings u, v.

We can now associate the inertial principle (‡) with the requirement that every factor of a string that subsumes $\Box\boxed{\varphi}$ also subsumes $\boxed{\varphi}\Box$ or $\boxed{\mathsf{f}\varphi}\Box$. We write

$$\Box\boxed{\varphi} \Rightarrow \boxed{\varphi}\Box + \boxed{\mathsf{f}\varphi}\Box$$

for the set of strings in $Fin(\Phi)^{*}$ that meet this requirement.[7] In general, given two languages L and L' over the alphabet $Fin(\Phi)$, we define the *constraint* $L \Rightarrow L'$ to consist of all strings $s \in Fin(\Phi)^{*}$ such that for every factor s' of s,

[7] A more detailed analysis of inertia and force is presented in Fernando (2006a), with constraints

$$\boxed{\varphi}\Box \Rightarrow \Box\boxed{\varphi} + \boxed{\mathsf{f}\neg\varphi}\Box$$

(regulating persistence to the right) and

$$\boxed{\mathsf{f}\varphi}\Box \Rightarrow \Box\boxed{\varphi} + \boxed{\mathsf{f}\neg\varphi}\Box$$

(saying intuitively, succeed unless opposed) for inertial fluents φ.

$$s' \trianglerighteq L \text{ implies } s' \trianglerighteq L'.$$

The special case of $L' = \emptyset$ reduces to the set

$$L \Rightarrow \emptyset = \{s \in \mathit{Fin}(\Phi)^* \mid \text{not } s \trianglerighteq \Box^* L \Box^*\}$$

of strings that do not subsume $\Box^* L \Box^*$. For $L = \boxed{\neg\psi, \psi}$, this removes the spurious possibility in Case 2 of Sect. 3.1.

But how exactly are constraints applied to refine a language? To answer this question, we revise subsumption $\trianglerighteq$ to a relation that is insensitive to $\Box$-padding. Let $L_\Box$ be L with any number of $\Box$'s inserted or deleted at the beginning and end of L

$$L_\Box \stackrel{\text{def}}{=} \Box^* \mathsf{unpad}\langle L \rangle \Box^*.$$

If L' subsumes $L_\Box$, we write $L' \blacktrianglerighteq L$, pronounced L' *weakly subsumes* L

$$L' \blacktrianglerighteq L \stackrel{\text{def}}{\iff} L' \trianglerighteq L_\Box.$$

Table 5 compares $\blacktrianglerighteq$ to other notions of containment above, including equality. We can strip off $\Box$-padding when evaluating weak subsumption

$$L' \blacktrianglerighteq L \iff \mathsf{unpad}\langle L' \rangle \blacktrianglerighteq \mathsf{unpad}\langle L \rangle.$$

Next, let us agree that L *C-entails* L', and write $L \vdash_C L'$, if every string in C that weakly subsumes L also weakly subsumes L'

$$L \vdash_C L' \stackrel{\text{def}}{\iff} (\forall s \in C)\ s \blacktrianglerighteq L \text{ implies } s \blacktrianglerighteq L'.$$

Under this definition, C is the set of possible situations against which to evaluate the implication from L to L'. If we write $L^{\blacktrianglerighteq}$ for the set of strings that weakly subsume L

$$L^{\blacktrianglerighteq} \stackrel{\text{def}}{=} \{s \in \mathit{Fin}(\Phi)^* \mid s \blacktrianglerighteq L\}$$

then $L \vdash_C L'$ can be put more concisely as: the intersection of $L^{\blacktrianglerighteq}$ with C weakly subsumes L'

Table 5 Notions of containment between strings

	Temporal stretch	Descriptive detail
Equality $=$	Fixed	Fixed
Subsumption $\trianglerighteq$	Fixed	Variable
Factor	Variable	Fixed
Weak subsumption $\blacktrianglerighteq$	Variable	Variable

$$L \vdash_C L' \iff (L^{\blacktriangleright} \cap C) \ \underline{\blacktriangleright}\ L'.$$

Constraints are introduced incrementally by intersection with C. The smaller C becomes, the more $\vdash_C$-entailments we get. The entailment $L \vdash_C L'$ reduces to an inclusion

$$L \vdash_C L' \iff (L^{\underline{\blacktriangleright}} \cap C) \subseteq L'^{\underline{\blacktriangleright}}$$

between languages $L^{\underline{\blacktriangleright}} \cap C$ and $L'^{\underline{\blacktriangleright}}$ that are regular provided L, C and L' are. Finding finite automata that accept L, C and L' pays off as it is well-known that inclusions are decidable between regular languages (but not between certain context-free languages). In this connection, it is worth noting that we can adapt a construction due to Koskenniemi (e.g. Beesley and Karttunen 2003) to show that the constraint $L \Rightarrow L'$ is regular if L and L' are. If we collect the strings that subsume L in $L^{\trianglerighteq}$

$$L^{\trianglerighteq} \stackrel{\text{def}}{=} \{s \in Fin(\Phi)^* \mid s \trianglerighteq L\}$$

and the strings not in L in the *complement* $\overline{L}$ *of* L

$$\overline{L} \stackrel{\text{def}}{=} \{s \in Fin(\Phi)^* \mid s \notin L\}$$

then we can express $L \Rightarrow L'$ as the complement of the set

$$Fin(\Phi)^*(L^{\trianglerighteq} \cap \overline{L'^{\trianglerighteq}})\ Fin(\Phi)^*$$

of strings with a factor belonging to $L^{\trianglerighteq}$ but not to $L'^{\trianglerighteq}$. That is, we have the Koskenniemi-esque equation

$$L \Rightarrow L' = \overline{Fin(\Phi)^*\ (L^{\trianglerighteq} \cap \overline{L'^{\trianglerighteq}})\ Fin(\Phi)^*}.$$

By the closure properties of regular languages, it follows that $L \Rightarrow L'$ is regular if L and L' are. (More in Fernando (2007))

4 Conclusion

The basic idea developed in this paper is to represent the temporal structure of events and situations as strings. To understand these representations, we applied a function π on strings for two ends in Sect. 2. First, the Allen interval relations were derived from superposition of sets of strings. Second, event structures were formed as projective limits of strings. In Sect. 3, we abstracted away from event occurrences to event types based on fluents with scattered temporal instantiations. The step from event occurrences to event types given by fluents is compatible with the claim in Steedman (2000) that there is more to natural language temporality than time (or the event occurrences from which Kamp derives moments of time). Rather than treat an event structure $\langle E, \prec, \bigcirc \rangle$ as a Kripke frame to interpret fluents (for an interval-based temporal logic), we chopped fluents of interval granularity to fluents with the granularity of Kamp's moments $M(\bigcirc)$. Finite sets of these fluents were then strung together to describe intervals. These strings finitely approximate Kripke models over linear orders, matching the bounded precision of natural language descriptions. We can sharpen these approximations by working with languages and not simply strings in isolation. But the approximations will each be discrete, mirroring the step-by-step operation of computer programs (and instructions, in general), each step of which is decomposable into finer steps.

The link with computation merits close scrutiny, as cognition presumably bears some relation to computation. It turns out that the simplest computational devices, finite automata and finite-state transducers, provide a rich supply of string sets for analyzing time, events and situations. Moreover, regular languages induce bounded but decidable notions of entailment, subject to incremental refinement by constraints. The use of finite strings is a move away from the totality of possible worlds, intended to improve the fit between bounded natural language descriptions and the semantic entities against which they are interpreted. The underlying claim is that by assigning events and situations very explicit representations, we can read entailments directly off these representations, without having to bring in models. Until this claim is accepted, however, models are indispensable for understanding what these representations might be about (with the bounded precision of language abstracted away).

References

Allen, J. F., & Ferguson, G. (1994). Actions and events in interval temporal logic. *Journal of Logic and Computation, 4*(5), 531–579.

Beesley, K. R., & Karttunen, L. (2003). *Finite state morphology*. Stanford: CSLI Publications.

Blackburn, P. (2000). Representation, reasoning, and relational structures: A hybrid logic manifesto. *Logic Journal of the IGPL, 8*(3), 339–365.

Dowty, D. R. (1979). *Word meaning and Montague grammar*. Dordrecht: Reidel.

Fernando, T. (2004). A finite-state approach to events in natural language semantics. *Journal of Logic and Computation, 14*(1), 79–92.

Fernando, T. (2006a). Finite-state temporal projection. In *Proc. 11th International Conference on Implementation and Application of Automata*, LNCS 4094, pp. 230–241. Springer-Verlag.

Fernando, T. (2006b). Representing events and discourse: Comments on Hamm, Kamp and van Lambalgen. *Theoretical Linguistics, 32*(1), 57–64.

Fernando, T. (2007). Temporal propositions as regular languages. In M. Aurnague et al (Eds.), *Language, representation and reasoning: Memorial volume to Isabel Gomez Txurruka* (pp. 167–187). University of the Basque Country Press.

Fine, K. (1975). Vagueness, truth and logic. *Synthese, 30*, 265–300.

Kamp, H., & Reyle, U. (1993). *From discourse to logic*. Dordrecht: Kluwer Academic Publishers.

McCarthy, J., & Hayes, P. (1969). Some philosophical problems from the standpoint of artificial intelligence. In M. Meltzer & D. Michie (Eds.), *Machine intelligence* (Vol. 4, pp. 463–502). Edinburgh University Press.

Steedman, M. (2000). *The productions of time*. Draft, ftp://ftp.cogsci.ed.ac.uk/pub/steedman/temporality/temporality.ps.gz, July 2000. Subsumes 91 'Temporality'. In J. van Benthem & A. ter Meulen (Eds.), *Handbook of logic and language* (pp. 895–935). Elsevier North Holland.

van Lambalgen, M., & Hamm, F. (2005). *The proper treatment of events*. Blackwell.
van Lambalgen, M., & Hamm, F. (2006) Additional material for *The proper treatment of events*, http://staff.science.uva.nl/michiell/docs/ProofsTime.pdf, accessed June 2006.
Vendler, Z. (1967). *Linguistics in philosophy*. Ithaca: Cornell University Press.
Venema, Y. (1990). Expressiveness and completeness of an interval tense logic. *Notre Dame Journal of Formal Logic, 31*(4), 529–547.