

# Tense, Temporal Reference and Tense Logic

Patrick Blackburn

## Abstract

This paper examines extensions of Priorian tense logic in which reference to times is possible. The key technical idea is to *sort* the atomic symbols of Prior's language and to impose different interpretational restrictions on the different sorts. Among the sorts introduced are nominals (which permit Reichenbachian analyses of tense and tense-in-texts to be reconstructed in tense logic) and sorts which mimic temporal indexicals and calendar terms. The possibilities raised by sorting richer systems are briefly discussed.

## 1 Introduction

Few formal semanticists look with much favour upon Priorian tense logic. For a start, natural language tenses usually have referential import — but reference to times is not possible in Priorian languages. Secondly, the focus of research on tense has shifted in recent years. The anaphoric properties of tense, and the role temporal constructions have to play in discourse, are now centre stage. Such issues seem alien to tense logic.

These shortcomings are real, nonetheless there is a great deal of value in Prior's approach. Prior insisted on the primacy of the *internal* view of time. This view situates the speaker firmly inside the temporal flow: the speech-time centred ordering of *past*, *present* and *future*, rather than the absolute *earlier-than*/*later-than* relation, is considered central to the analysis of tensed talk. This is natural: we live *in* time, and the internal perspective is imprinted on natural language in many ways.

Prior's insistence on the internal perspective led him to develop his temporal calculi as modal logics, not classical logics. Classical logic, with its explicit variables and binding, offers a Gods-eye-view of temporal structure; modal logic, on the other hand, uses operators to quantify over structure 'from the inside'. Moreover, because modal logic rejects the complexities of variables and binding in favour of operators, the resulting systems are very simple.

Prior's decision to work with modal languages has interesting logical consequences. For example, as we now know from *modal correspondence theory* (see van Benthem [1][2]), to work with a modal language is essentially to work with a very restricted fragment of classical logic, a fragment with many special properties. Thus if a problem can be modeled in a modal language, we already know something important: the full power of classical logic is not needed. Other logical aspects of Prior's system are also well understood. For example completeness, decidability and complexity results for interesting classes of temporal structures have been obtained, and there is a growing interest in developing good theorem provers. Given the increasing interplay between formal semantics and disciplines such as knowledge representation, and the consequent emphasis on the role of inference, this logical insight is valuable.

Thus there are many good reasons for attempting to extend Prior's approach rather than simply abandoning it. This paper introduces a number of extensions and shows that they

are adequate for dealing with the problems mentioned above, as well as a number of other difficulties.

Now, when extending Prior's system we must take care not to destroy those properties that made it attractive in the first place. The extensions proposed in this paper are all particularly simple and revolve around one central strategy: *sorting* the atomic symbols of the language. Different sorts of propositional symbols are introduced, and their interpretation is constrained in various ways to achieve the desired forms of temporal reference. This strategy preserves the simplicity of the Priorean model (both the syntactic and semantic changes introduced are confined to the atomic level) and does not tamper with the 'internal perspective' characteristic of modal logic. Moreover, the logical results known for systems of the kind considered here confirm that the strategy really is straightforward: completeness, complexity and correspondence results have been given for such systems (see, for example, Gargov and Goranko [11] or Blackburn [4][5]) and these results typically extend the standard results for Priorean languages in a direct manner. Thus the proposed extensions preserve the desirable properties of Prior's system: it remains to be seen, however, whether they are of interest in natural language semantics. The purpose of the present paper is to show that they are.

We proceed as follows. After a *résumé* of Priorean tense logic, we introduce our first sorted language: *nominal tense logic*. In this extension a second sort of propositional variable — the *nominal* — is introduced. Nominals are constrained to be true at exactly one time, thus they act as a name for the time they are true at. This simple referential mechanism enables us to combine the ideas of Prior and Reichenbach in a natural way. It also allows us to deal with certain anaphoric phenomena; the key idea here is to use nominals as if they were discourse markers of temporal DRT.

With the sorting strategy thus established, we turn to indexicals. Nominal tense logic is enriched with three new propositional symbols: *yesterday*, *today*, and *tomorrow*, and these are interpreted using the semantic machinery of the California theory of reference. As we shall see, the exploitation of this machinery by means of sorted atomic symbols yields a clean account of some basic interactions between tense and temporal adverbials. Following this we introduce calendar terms, and then discuss the possibility of sorting richer systems.

Some historical remarks are in order. The idea of sorting intensional languages can be traced back to work done in the late 1960s by Arthur Prior [23][24] and Robert Bull [7]. This work seems to have been largely overlooked. The Sofia school of modal logic independently developed similar ideas in the 1980s, in the context of both Propositional Dynamic Logic [20][21], and modal logic [10][11]. Slightly later Blackburn [3][4] considered sorting in the setting of temporal logic. With the exception of Prior's discussion of the indexical 'now', most of this literature is technical and does not consider applications of sorting to natural language semantics.

## 2 Priorean Tense Logic

The language of (propositional) Priorean tense logic is constructed out of the following primitive symbols. First we have a countably infinite collection VAR of propositional variables, which we officially write as  $p$ ,  $q$ , and  $r$ , possibly subscripted; however when discussing examples drawn from natural language we shall frequently use more mnemonic symbols such as *John run*, *Fido bark* and *The third door on the left be brown*. In addition we have some truth functionally adequate collection of Boolean connectives (in this paper we shall choose  $\neg$  and  $\wedge$ ), and the tense operators  $P$  and  $F$ .

We form the well formed formulas (or sentences) of Priorean tense logic from these

symbols as follows. First we define ATOM, the set of atomic symbols of the language, to be precisely VAR, the set of propositional variables. We then stipulate that WFF, the set of well formed formulas of the language, is to be the smallest set containing ATOM that is closed under negation, conjunction, and the application of tense operators. We make free use of all the usual defined symbols. In particular, we define the other Boolean connectives  $\rightarrow$ ,  $\vee$  and  $\leftrightarrow$  in the usual manner, and in addition define  $G\phi$  to be  $\neg F\neg\phi$ , and  $H\phi$  to be  $\neg P\neg\phi$ , for all wffs  $\phi$ .

All this will probably be familiar to the reader, with one (perhaps puzzling) exception: why did we bother introducing the set ATOM? This is clearly redundant: we could have defined WFF directly in terms of VAR. In fact the introduction of ATOM is an anticipation of the later sorted languages we shall consider. Syntactically, sorting merely amounts to introducing different kinds of atomic symbol, and alterations of the definition of ATOM are the only syntactic changes made in what follows. Thus all the languages discussed in this paper have the same formation rules, those stated above.

Prior's language is interpreted using *frames* and *Kripke models*. A frame  $\mathbf{T}$  is a pair  $\langle T, < \rangle$  where  $T$  is a non-empty set, and  $<$  is a binary relation on  $T$ . The elements of  $T$  are often called *times* or *points*, and  $<$  is usually called the *precedence relation*. In order to make this simple model more realistic we shall place some constraints on allowable precedence relations. In what follows we shall assume that  $<$  is always a *strict total order*; that is, a relation that is transitive, irreflexive, and trichotomous. In short, we are assuming a linear flow of time. This assumption will suffice for the first half of the paper; thereafter we will need to be a little more demanding.

A Kripke model  $\mathbf{M}$  is a pair  $\langle \mathbf{T}, V \rangle$  where  $\mathbf{T}$  ( $= \langle T, < \rangle$ ) is a frame and  $V$  is a *valuation* on  $\mathbf{T}$ . A valuation on  $\mathbf{T}$  assigns a subset of  $\mathbf{T}$  to each atom; that is, such a valuation is a function from ATOM to  $Pow(T)$ . Valuations thus impose 'information distributions' on frames, and Kripke models can be thought of as time flows decorated with information about what happens when.

Given a Kripke model  $\mathbf{M}$  ( $= \langle T, <, V \rangle$ ), a wff  $\phi$ , and a point  $t \in T$  the fundamental thing we are concerned with is whether  $\mathbf{M}$  *satisfies*  $\phi$  at  $t$ , or in symbolic form, whether  $\mathbf{M} \models \phi[t]$ . This relation is defined as follows:

$$\begin{array}{lll} \mathbf{M} \models a[t] & \text{iff} & t \in V(a), \text{ for all } a \in \text{ATOM} \\ \mathbf{M} \models \neg\phi[t] & \text{iff} & \text{not } \mathbf{M} \models \phi[t] \\ \mathbf{M} \models \phi \wedge \psi[t] & \text{iff} & \mathbf{M} \models \phi[t] \text{ and } \mathbf{M} \models \psi[t] \\ \mathbf{M} \models F\phi[t] & \text{iff} & \exists t' (t < t' \text{ and } \mathbf{M} \models \phi[t']) \\ \mathbf{M} \models P\phi[t] & \text{iff} & \exists t' (t' < t \text{ and } \mathbf{M} \models \phi[t']) \end{array}$$

Note the way this satisfaction definition captures the essence of the internal perspective. We evaluate formulas at points *inside* models (the evaluation point can be thought of as speech-time) and the operators F and P scan forward and backwards relative to this deictic centre.

In the work that follows we occasionally make use of the idea of *logical equivalence*. This is defined as follows. Two wffs  $\phi$  and  $\psi$  are logically equivalent iff for all models  $\mathbf{M}$  and all points  $t$  in  $\mathbf{M}$ :

$$\mathbf{M} \models \phi[t] \text{ iff } \mathbf{M} \models \psi[t].$$

One final remark is worth making. Note that the base case of the satisfaction definition is parameterised in terms of ATOM. As we proceed we are going to impose a syntactic sortal structure on ATOM. This syntactic sortal structure will have a semantic correlate: we are going to place constraints on how the different sorts may be interpreted. But such atomic level constraints are the only semantic changes we shall make; the satisfaction definition

given above will not need changing when we consider *nominal tense logic* in the following section.

### 3 Nominal Tense Logic

As Priorean tense logic contains no mechanism for referring to times, its use as a model of tense in natural language has severe limitations. Tenses aren't solely, or even primarily, quantificational in nature: they usually have referential import. For example, an utterance of "John ran" doesn't mean that John ran at some completely unspecified past time, but that he ran at some *particular* (contextually determined) past time. The natural representation of this sentence in Priorean tense logic,  $P(\textit{John run})$ , fails to mirror the referential nature of the English original.

Our solution to this difficulty is very simple. We make Priorean tense logic referential by means of the following extension. We add a new type of propositional symbol, called *nominals*, to our language. Nominals will typically be written as  $i, j, k$  and so on. They will be wffs, and we will be able to freely combine them with other wffs in the usual fashion to form new wffs. The point of introducing nominals lies in their interpretation: *we'll insist that in any model nominals are to be true at exactly one time*. Nominals thus 'name' the unique time they are true at and our problem is solved. For example we will now have a simple representation for "John ran" that captures its referential force:  $P(i \wedge \textit{John run})$  suffices, for in order for this to be true John must run at the past time picked out by  $i$ .

That's all there is to it. Let's go through it in more detail and see where it leads.

Syntactically we obtain the language of nominal tense logic  $\text{NTL}$  from the language of Priorean tense logic by adding a denumerably infinite set  $\text{NOM}$  of fresh atomic symbols to the language. The elements of  $\text{NOM}$  are denoted by  $i, j, k \dots$ , possibly subscripted, and they are called nominals. These symbols are 'fresh' in the sense that  $\text{NOM} \cap \text{VAR} = \emptyset$ . That is, nominals and propositional variables are distinct. We define the set of atoms  $\text{ATOM}$  of the language of  $\text{NTL}$  to be  $\text{NOM} \cup \text{VAR}$ , and we make the wffs of this enriched language using exactly the same formation rules as for the Priorean language:  $\text{WFF}$  is the smallest set containing  $\text{ATOM}$  that is closed under negation, conjunction, and application of tense operators. Thus the only syntactic difference is that  $\text{ATOM}$  is now  $\text{VAR} \cup \text{NOM}$ , instead of simply  $\text{VAR}$ .

As with the Priorean language, we interpret  $\text{NTL}$  in Kripke models  $\mathbf{M} = \langle \mathbf{T}, V \rangle$ . As before  $\mathbf{T}$  is a frame  $\langle T, < \rangle$  and  $V$  is a function from  $\text{ATOM}$  to  $\text{Pow}(T)$ . However we now make a crucial demand: *for all  $i \in \text{NOM}$ ,  $V(i)$  is a singleton subset of  $T$* . Functions  $V$  that don't obey this sortal restriction aren't valuations. As with the syntax, this atomic level change is the only change we make. Wffs of  $\text{NTL}$  are interpreted in Kripke models in exactly the manner familiar from Priorean tense logic.

As the representation of "John ran" as  $P(i \wedge \textit{John run})$  shows, the utility of nominals stems from the fact that they are a referential mechanism. Because  $i$  is true at one and only one point in any model, this representation makes a far stronger claim than  $P(\textit{John run})$ . In order for  $P(i \wedge \textit{John run})$  to be true at some time of utterance  $t$ , not only must there be a time  $t'$  preceding  $t$  at which *John run* is true, but in fact the unique time picked out by  $i$  must precede  $t$  and it must be true *at this particular time* that *John run*. In short, the enriched language gives us the power to refer — and because temporal reference is such a pervasive phenomenon in natural language, this will enable us to model some interesting phenomena. For example,  $\text{NTL}$  provides a framework in which the ideas of Reichenbach and Prior can be naturally amalgamated.

Reichenbach claimed that understanding tensed expressions involved understanding the

temporal relations that held between *three* special points of time: the *point of speech*, the *point of event* and the *point of reference*. The first two points have the obvious meanings: the point of speech is the utterance time, and the point of event is the time at which the event being spoken of occurred. Note that Priorean tense logic, with its internal view of time, gives a clear account of these two points: the point of speech is the point in the model at which the utterance is evaluated, while the point of event is the point in the model where the event being spoken of is verified.

But what of Reichenbach’s novelty, *point of reference*? This concept is best introduced by example. Consider the past perfect sentence “John had run”. Note the way it works. Our attention is not directed immediately to the time at which John runs, rather we are first referred to some point  $t'$  preceding the point of speech and told that at some point prior to this intermediate point John ran. Such an intermediate point is called a point of reference.

Reichenbach accounted for the variety of tenses found in natural language in terms of the different patterns of temporal precedence and coincidence these three points can exhibit. For example, in the past perfect we have that the point of event  $E$  precedes the point of reference  $R$ , which in turn precedes the point of speech  $S$ , or, to use Reichenbach’s diagrammatic notation,  $E - R - S$ . To give a second example, Reichenbach proposed that the function of simple past tense was to locate the point of event and point of reference at the same point in the past, thus his diagram for the simple past was  $E, R - S$ . Now this concept of a special intermediate point picked out by tensed sentences simply isn’t present in the Priorean framework. Equally clearly, once we have nominals in our language to refer to times, we can mirror this idea. Let’s consider the matter systematically. Reichenbach’s idea admits of thirteen possible patterns of precedence and coincidence. The following table, the first three columns of which are taken from Comrie [8, page 25], lists with examples the possibilities admitted by Reichenbach’s analysis. The fourth column gives an NTL representation.

Structure	Name	English example	Representation
E-R-S	Pluperfect	I had seen	$P(i \wedge P\phi)$
E,R-S	Simple past	I saw	$P(i \wedge \phi)$
R-E-S	Future-in-the-past	I would see	$P(i \wedge F\phi)$
R-S,E	Future-in-the-past	I would see	$P(i \wedge F\phi)$
R-S-E	Future-in-the-past	I would see	$P(i \wedge F\phi)$
E-S,R	Present perfect	I have seen	$P\phi$
S,R,E	Present	I see	$\phi$
S,R-E	Prospective	I am going to see	$F\phi$
S-E-R	Future perfect	I will have seen	$F(i \wedge P\phi)$
S,E-R	Future perfect	I will have seen	$F(i \wedge P\phi)$
E-S-R	Future perfect	I will have seen	$F(i \wedge P\phi)$
S-R,E	Future	I will see	$F(i \wedge \phi)$
S-R-E	Future-in-the-future	(Latin: abiturus ero)	$F(i \wedge F\phi)$

We could sum up these entries by saying that we have factored the English tenses into a ‘shift’ (performed by the tense operators) and a ‘refer’ (performed by the nominals). The ‘shift’ component is Prior’s contribution, the ‘refer’ Reichenbach’s.

Some further comments on these representations are called for. First note that all three logically possible permutations of  $R$ ,  $E$  and  $S$  for both the Future-in-the-past and the future perfect are represented by a single formula. Such a mode of representation seems to be preferred by linguists. Comrie, for example, argues that such non-committal representations are essential [8, pages 26 – 27]; indeed he criticises Reichenbach’s diagrams for their specificity. His argument is that while one of the three possibilities may be favoured, and one largely

ruled out, these preferences are merely Gricean implicatures: all three usages are possible, and a uniform representation is required.

Second, note that the present perfect and the simple past are represented differently. The identical representations these two forms receive in standard tense logic is a source of dissatisfaction with the use of tense logic in the analysis of natural language. However, though it is pleasant that there is a distinction between the representation of these two tenses in NTL, note how it arises. It has been achieved by adding something — namely a Reichenbachian component — to Prior’s account of the past tense. On the other hand we have added nothing to the (unsatisfactory) Priorean account of the perfect. The perfect poses difficult problems for formal semantics, and the machinery introduced in this paper does not solve them.

Actually, even for English, the above table does not exhaust all the possibilities; for, as Prior observed [23, page 13], certain sentences seem to require more than one point of reference. Consider Prior’s example, “I shall have been going to see John”. This seems to require two points of reference,  $R1$  and  $R2$ . The pragmatically most plausible pattern they will exhibit is  $S - R2 - E - R1$ . The semantic possibilities are captured in NTL as follows:

$$F(i \wedge P(j \wedge F(I \text{ see } John))).$$

Comrie [8][9, pages 122-130] develops this idea to its logical conclusion. He proposes a modification of Reichenbach’s system which essentially amounts to allowing an arbitrary number of reference points. (Comrie also replaces Reichenbach’s simultaneity relation with a relation of intervalic overlap. This idea is easy to model in the sorted interval based languages discussed later in this paper.) The possible tenses are then defined to be linkages between the point of speech and the point of event mediated by a (possibly null) sequence of reference points. This idea isn’t made completely precise, but the following seems to capture Comrie’s intention. Let  $\phi$  and  $\psi$  be metavariables over wffs and  $n$  a metavariable over nominals. We take as primitive tensed forms all instances of  $P\phi$ ,  $\phi$ , and  $F\phi$ . We then say that if  $\psi$  is a tensed form then so are  $P(n \wedge \psi)$ ,  $(n \wedge \psi)$ , and  $F(n \wedge \psi)$ . Everything in the previous table falls under this definition, and according to Comrie, so do most of the tense forms exhibited in natural languages.

## 4 Intersentential anaphora

In this section we use nominals to model temporal intersentential anaphora. We approach this subject via Partee’s paper “Nominal and Temporal Anaphora” [22]. Partee’s discussion, which draws on earlier work by Hinrichs [12], is couched in the language of temporal DRT [14][15]. In the remainder of this section will see how Partee’s examples can be dealt with in NTL. The key idea is to use nominals as discourse markers — or, to put a more Reichenbachian slant on it, to think of points of reference (as Reichenbach himself did) from the standpoint of intersentential anaphora.

Consider the discourse “The shutters were closed. It was dark.” As Partee points out, the tense in the second sentence (which is a *state* sentence) anaphorically picks out the time referred to by the tense of the first sentence (which is also a state sentence). We can represent this discourse in NTL as follows:

$$P(i \wedge \textit{The shutters be closed}) \wedge P(i \wedge \textit{It be dark}).$$

Note that we have reused the  $i$  nominal provided by the first sentence in the representation of the second. This really does give us what is desired, namely reference to a past

time that is both a shutters-being-closed-time and an it-being-dark time. Although the two occurrences of the  $i$  nominal are in different conjuncts and embedded within the scopes of different past tense operators, we achieve the desired semantic effect, as it follows from the semantics of NTL that the above representation is logically equivalent to:

$$P(i \wedge \textit{The shutters be closed} \wedge \textit{It be dark}).$$

That is, because nominals are ‘hardwired’ to a unique time, no matter how deeply they are embedded within the structure of our representations they are always available for reuse.

This ability of nominals to ‘float out of the scope’ of tense operators means we can build up discourse representations online. Consider the above discourse once more. Imagine that we are processing this discourse, building up its representation as we go. First we process “The shutters were closed”. The parser goes to work and builds  $P(i \wedge \textit{The shutters be closed})$ . The second sentence is then analysed. Because it is a sentence in the simple past tense a representation of the form  $P(n \wedge \textit{It dark})$  must be built, where  $n$  is to be instantiated with some nominal. But which nominal should be used? Because the second sentence is a state sentence that follows hot on the heels of another, the parser reuses the  $i$  of the previous representation. As we have just seen, this natural strategy is semantically correct.

Matters are more interesting when we have a sequence of two or more *event* sentences. Consider the discourse: “Mary woke up sometime during the night. She turned on the light.” Representing this in NTL by:

$$P(i \wedge \textit{Mary wake up sometime during the night}) \wedge P(i \wedge \textit{She turn on the light})$$

is inadequate: clearly the illumination should follow the awakening. The event verb of the second sentence has advanced the referential focus. This is a typical feature of event sentences. Whereas state sentences elaborate on the time in focus, one of the functions of event sentences is to move the story along. (Of course, this is a simplification of a complex phenomena. However as my aim is simply to show that Partee’s account can be reconstructed in NTL, and I shall adopt her assumptions without further discussion.) In fact it is simple to advance the reference time in the required fashion: we need merely introduce a new reference marker (that is, a new nominal) and insist that it names a more recent time than did the old:

$$P(i \wedge \textit{Mary wake up sometime during the night}) \wedge P(j \wedge Pi \wedge \textit{She turn on the light}).$$

Note how this works. The reference time of the first sentence is picked out by  $i$ . The second sentence then advances the reference time, and we capture the effect of this as follows. Instead of simply introducing a new nominal  $j$  we introduce the conjunction  $j \wedge Pi$ . The role of  $j$  is to pick out the new reference time, and the conjunct  $Pi$  guarantees that the old reference time lies in the past.

As a third example consider the following discourse which mixes stative and event verbs: “John got up, went to the window, and raised the blind. It was light out. He pulled the blind down and went back to bed. He wasn’t ready to face the day. He was too depressed.” We can represent this as follows:

$$\begin{aligned} & P(i \wedge \textit{John get up}) \\ \wedge & P(j \wedge Pi \wedge \textit{go to the window}) \\ \wedge & P(k \wedge Pj \wedge \textit{raise the blinds}) \\ \wedge & P(k \wedge \textit{It be light out}) \\ \wedge & P(i_1 \wedge Pk \wedge \textit{He pull blind down}) \\ \wedge & P(j_1 \wedge Pi_1 \wedge \textit{go back to bed}) \\ \wedge & P(j_1 \wedge \textit{He be not ready to face the day}) \\ \wedge & P(j_1 \wedge \textit{He be too depressed}) \end{aligned}$$

In short, we can build the desired representations using a strategy that boils down to ‘shift and refer’ coupled with ‘advance the reference time when an event verb is encountered’.

## 5 Indexicals

In this section we use sorting to model the indexicals ‘yesterday’, ‘today’ and ‘tomorrow’. We add three new symbols — *yesterday*, *today* and *tomorrow* — to NTL. These new symbols are atomic wffs and may be freely combined with other wffs to make more complex wffs. They will be used to represent English sentences containing indexicals. For example, we will be able to represent “John climbed yesterday” as

$$P(\textit{yesterday} \wedge \textit{John climb}),$$

and this representation will correctly capture the fact that “John ran yesterday” is true iff John did in fact run at some point during the day immediately preceding the day of utterance.

But how are we to specify the semantics of these new items so that they correctly mimic the behaviour of the English originals? An elegant answer lies to hand: we shall make use of certain semantic machinery developed by Montague [18], Kamp [13] and Kaplan [16][17], namely the machinery underlying the California theory of reference. In spite of its age this theory remains one of the most detailed responses to the problems posed by indexicals. As we shall see, the main ideas of the Californian approach mesh smoothly with the techniques of the present paper, and in fact the major difference between this paper and the earlier work is not conceptual but technical. Hitherto the semantic machinery of the California theory of reference was exploited by means of additional modal operators. We exploit the same machinery here, but not with operators: instead we sort. This leads to a cleaner account of the basic interactions between tense and indexical reference.

Let’s begin with the simplest task, defining the syntax of our language. This is NTL augmented with three new atomic symbols: *yesterday*, *today* and *tomorrow*. The set of atomic symbols of this language is defined as follows:

$$\text{ATOM} = \text{VAR} \cup \text{NOM} \cup \{\textit{yesterday}, \textit{today}, \textit{tomorrow}\}.$$

We make wffs from this stock of symbols exactly as before. That is, WFF is defined to be the smallest set containing ATOM that is closed under negation, conjunction and the application of tense operators.

Now for the task of defining the semantics. First of all, we need to pay more attention to our choice of temporal ontology. These indexicals all presuppose the concept of dayhood, thus it is sensible to work with temporal structures rich enough to support a reasonably realistic picture of this assumed day structure. As there’s no obvious single best choice, let’s agree for the purpose of the present paper to work with a particularly simple option:  $\mathbf{Q}$  ( $= \langle Q, < \rangle$ ), the rational numbers in their usual order. From now on we’ll only consider models based on this frame. We can impose a reasonable looking day structure on  $\mathbf{Q}$  as follows. Define the set of *Days* on  $\mathbf{Q}$  to be  $\{[z, z + 1) : z \in \mathbf{Z}(\subseteq \mathbf{Q})\}$ . Here  $\mathbf{Z}$  ( $= \langle Z, < \rangle$ ) is the set of integers, and we’re using them as a simple (and arbitrary) way to mark off the division between successive days. We also make use of the following three functions. The function *day* maps elements  $q$  of  $Q$  to the unique element of *Days* that contains  $q$ . That is, given a point of time, *day* answers the question ‘What day does this point belong to?’. The functions *next* and *prev* are both maps from *Days* to *Days*. In particular, for all integers  $z$ ,  $\textit{next}([z, z + 1)) = [z + 1, z + 2)$  and  $\textit{prev}([z, z + 1)) = [z - 1, z)$ , so these functions take



as input a day and return, respectively, the next day and the previous day. This is all we'll need to assume in the way of day structure for the purposes of this paper.

With these preliminaries established, we are now ready to introduce the apparatus of the California theory of reference. The key idea of the California theory of reference is that context must be taken into account when evaluating the truth of utterances. This simple idea is made precise as follows: contexts are added to Kripke models, and when wffs are evaluated they are evaluated not only at a certain time, but also at (or *in*) a certain context of utterance.

Let's make this precise. Fix some non-empty set  $C$ , the set of *contexts*, or *contexts of utterance*, and specify a function  $g : C \rightarrow Q$ . The function  $g$  is to be thought of as specifying the *utterance time* of each context of utterance. We call the quadruple  $\langle Q, <, C, g \rangle$  a *contextualisation of  $\mathbf{Q}$* . Clearly this is an extremely meagre notion of context, nonetheless it suffices for the phenomena we're going to consider.

We are now ready to interpret our language. As always with sorted systems, the key point is to say what we mean by a valuation. Here is the definition. A valuation is a function  $V : ATOM \times C \rightarrow Pow(Q)$  that respects the following constraints. First, for each nominal  $i$  and each context  $c$ ,  $V(i, c)$  must be a singleton. Second, for each context  $c$ :

$$\begin{aligned} V(today, c) &= day(g(c)) \\ V(tomorrow, c) &= next(day(g(c))) \\ V(yesterday, c) &= prev(day(g(c))) \end{aligned}$$

The intuition behind these constraints should be clear. In any context  $c$  the *today* atom is to be true at all points in the day containing the utterance time and at no others. That is, *today* picks out precisely those points belonging to the 'day of utterance'. Next, *yesterday* and *tomorrow* are to be true at, respectively, precisely those points in the day preceding, and the day succeeding, the day of utterance. Finally, note that within a fixed context of utterance  $c$ , nominals (as before) pick out a unique time. This definition allows the possibility that in different contexts of utterance the same nominal  $i$  may be used to name different times.

In giving these atomic level stipulations we have essentially completed our task. The following clauses explain how to interpret arbitrary wffs, but as the reader familiar with the Californian theory will observe, they are standard: everything of importance has already taken place. So, define model  $\mathbf{M}$  to be a pair  $\langle \langle Q, <, C, g \rangle, V \rangle$  where  $\langle Q, <, C, g \rangle$  is a contextualisation of  $Q$  and  $V$  is a valuation. Then for all times  $t \in Q$  and all contexts  $c \in C$  we define:

$$\begin{aligned} \mathbf{M} \models a[t, c] &\text{ iff } t \in V(a, c), \text{ for all } a \in ATOM \\ \mathbf{M} \models \neg\phi[t, c] &\text{ iff not } \mathbf{M} \models \phi[t, c] \\ \mathbf{M} \models \phi \wedge \psi[t, c] &\text{ iff } \mathbf{M} \models \phi[t, c] \text{ and } \mathbf{M} \models \psi[t, c] \\ \mathbf{M} \models F\phi[t, c] &\text{ iff } \exists t'(t < t' \text{ and } \mathbf{M} \models \phi[t', c]) \\ \mathbf{M} \models P\phi[t, c] &\text{ iff } \exists t'(t' < t \text{ and } \mathbf{M} \models \phi[t', c]) \end{aligned}$$

Let's put these enrichments to work. Consider the sentence "John ran yesterday". We could represent this as,  $P(yesterday \wedge John \text{ run})$ . This representation captures the temporal force that the 'yesterday' adds to "John ran". For suppose we're in some context of utterance  $c$ , and suppose we evaluate  $P(yesterday \wedge John \text{ run})$  at the pair  $[g(c), c]$ . That is, we're asking ourselves what happens if we 'utter this wff' in the context  $c$  at the utterance time for  $c$ . Then it follows from the above truth definition that this wff is true iff there is a time  $t'$  in the day preceding the utterance day such that *John run* is true at  $[t', c]$ . In short, this wff is true iff John did run during the day classified as the yesterday of the context of occurrence.

Now this is precisely what is required. Moreover, as many temporal semanticists have pointed out, it's something that multiple operator approaches have difficulty achieving. To take a recent example, Oversteegen [19, page 3] complains that:

If temporal adverbs like **yesterday** are treated as operators, there is a scope problem: both sequences, **yesterday** in the scope of the tense operator and the tense operator in the scope of **yesterday**, are obviously wrong.

Such scoping problems simply don't arise with the sorting strategy. The reason is simple. Our indexicals don't in any sense quantify, rather they act as a range restrictors on the quantification performed by the familiar temporal operators. For example the role of *tomorrow* in the wff  $F(\textit{tomorrow} \wedge \textit{John run})$  is to ensure that only those times classified as being 'tomorrow' are relevant to the truth or falsity of the wff. The familiar problems resulting from operator interactions are bypassed.

This talk of restricting the range of temporal quantification bears on the work of the previous section. There we talked of factoring tenses into a 'shift' and a 'refer': the tense operators pointed us in the desired direction, and nominals filled the 'referential slot' giving us the desired specificity. But it is clear that the mechanism underlying both  $F(i \wedge \textit{John run})$  and  $F(\textit{tomorrow} \wedge \textit{John run})$  is essentially the same. Not only do these wffs share a common syntactic form, *i* and *tomorrow* share the common semantic role of restricting the times of relevance to the *F* operator. 'Shift and refer' is a special case of 'shift and restrict'.

Thinking in this more general fashion in terms of quantifications and restrictions seems a useful way of thinking about temporal representations. This is particularly apparent when it comes to building up discourse representations. Consider the following discourse: "John went running yesterday. It was wet and cold and nasty." This can be represented as follows:

$$P(i \wedge \textit{yesterday} \wedge \textit{John run}) \\ \wedge P(i \wedge \textit{It be wet and cold and nasty})$$

Note that the representation of the first sentence contains *two* restrictors: the nominal *i* and *yesterday*. As described in the previous section, the nominal is contributed by the simple past tense of 'went'. It gives us a name for the reference point. The *yesterday* is contributed by the word 'yesterday'. Its function is to constrain when the running took place, namely to 'within yesterday'. Together the two restrictors enable the temporal information encapsulated in "John went running yesterday", to be captured, and crucially part of this representation (the *i*) is reused to correctly anchor the second sentence.

Our discussion has got rather general. Let's return to specifics. Another pleasant feature of these restricting representations is that they handle certain semantically anomalous sentences rather nicely, namely those sentences where the shift demanded by the tense clashes with the reference demanded by the indexical. For example, both

\*John will run yesterday

and

\*John ran tomorrow

are semantically anomalous for this reason. Let's consider what happens when we give the obvious 'shift and restrict' representations to these sentences, namely:

$$F(\textit{yesterday} \wedge \textit{John run})$$

and

$$P(\textit{tomorrow} \wedge \textit{John run}).$$

Just as we would hope, neither wff is satisfiable in any model. That is, no matter what model we choose, no matter what pair  $[t, c]$  we evaluate at, both wffs evaluate to false. Our sorted language ‘sees’ the semantic anomaly; to some extent it is mimicking the way tense and temporal reference interact in English.

## 6 Calendar terms

This section sketches another application of the sorting strategy; modeling calendar terms such as ‘January’, ‘Friday’, ‘March’ and ‘1984’. We first extend the language by adding five further sorts:

$$\begin{aligned} \text{DAY} &= \{ \textit{Sunday}, \dots, \textit{Saturday} \} \\ \text{DATE} &= \{ \mathbf{1}, \dots, \mathbf{31} \} \\ \text{MONTH} &= \{ \textit{January}, \dots, \textit{December} \} \\ \text{YEAR} &= \{ \mathbf{1}, \mathbf{2}, \mathbf{3}, \dots, \mathbf{1989}, \mathbf{1990}, \dots \} \\ \text{ERA} &= \{ \textit{BC}, \textit{AD} \} \end{aligned}$$

We define  $\text{CAL}$  to be  $\text{DAY} \cup \text{DATE} \cup \text{MONTH} \cup \text{YEAR} \cup \text{ERA}$  — we assume that all these sets are mutually disjoint from each other and from all the sorts we already have — and then define:

$$\text{ATOM} = \text{VAR} \cup \text{NOM} \cup \{ \textit{yesterday}, \textit{today}, \textit{tomorrow} \} \cup \text{CAL}.$$

The wffs are made from  $\text{ATOM}$  in the usual fashion. As an example of how we are going to use this language, the sentence “John ran on Monday 4th December 1989” will be represented by:

$$P(\textit{Monday} \wedge \mathbf{4} \wedge \textit{December} \wedge \mathbf{1989} \wedge \textit{AD} \wedge \textit{John run}).$$

I’ll now sketch the semantic definitions that ensure that this wff does its work properly. Recall from the previous section that we are only considering models based on the frame  $\mathbf{Q}$ , and that we have defined what we mean by a day structure over  $\mathbf{Q}$ . Thus it is relatively straightforward — if a little tedious — to define the needed *calendar structure* to interpret our new atoms. The required calendar structure  $\textit{Cal}$  is a 4-tuple:

$$\textit{Cal} = \langle \textit{Weeks}, \textit{Months}, \textit{Years}, \textit{Eras} \rangle.$$

Each of its four components, *Weeks*, *Months*, *Years* and *Eras*, is a partition of  $Q$ . (A partition of  $Q$  is a subset  $P$  of  $\text{Pow}(Q)$  such that  $\bigcup P = Q$ , each  $p \in P$  is non-empty, and for all  $p, p' \in P$ ,  $p \neq p'$  implies  $p \cap p' = \emptyset$ .) We assume without further ado the natural induced ordering of *Weeks*, *Months*, *Years* and *Eras*. For example, if  $m, m' \in \textit{Months}$  we say that  $m < m'$  iff  $\forall t \in m \forall t' \in m' (t < t')$ .

We want to place the obvious conditions on each of these partitions so that  $\textit{Cal}$  really does look like a calendar. Consider the case of *Years*, for example. First, define a *year* to be the set of points contained in 365 or 366 consecutive elements of *Days*. (Recall from the previous section that *Days* is the set of days in the day structure, and further recall that both successor and predecessor functions on days are defined as part of the day structure. Thus ‘consecutive days’ are defined in the obvious way using this apparatus.) We then define the set *Years* in the calendar structure to be any ‘leap year structured’ partition of  $Q$  such that every element of the partition is a *year*. By ‘leap year structured’ is meant that every fourth element in *Years* is a *year* containing 366 days, while the remaining three out of four *years* contain only 365 days.

We construct *Weeks*, *Months* and *Eras* in similar fashion. Every *week* in *Weeks* is made up of the points in seven consecutive days; every *month* in *Months* is made up of the

points in either 28, 29, 30, or 31 consecutive days; and *Eras* partitions  $Q$  into two. However as well as ensuring that each of these four sets *Weeks*, *Months*, *Years* and *Eras* is correctly structured, we must also ensure that they ‘hang together’ correctly. For example, the end of the first era should coincide with the end of some year; the start of a year should coincide with the start of a month containing 31 days, while the second month in any year should contain 28 days, unless the year is a leap year when it must contain 29; and so on. Stating these details completely would be tedious; I’ll leave the details to the reader’s knowledge of calendars and proceed.

We must now describe what we mean by valuations. These are maps from

$$V : \text{ATOM} \times C \longrightarrow \text{Pow}(T),$$

which satisfy the constraints of the previous section, and which in addition satisfy the needed constraints concerning the sorts in CAL. These constraints, though obvious, are wearisome to spell out in detail, so I’ll only give them for the simplest case, the two atoms  $BC$  and  $AD$  in ERA. Recall that *Era* is a binary partition of  $Q$ ; let us write this partition as  $\{bc, ad\}$ , where  $bc < ad$ . If  $V$  is to qualify as a valuation it must satisfy the demands that:

$$\begin{aligned} V(BC, c) &= bc \\ V(AD, c) &= ad, \end{aligned}$$

for all  $c \in C$ . I leave the task of stipulating the constraints needed for the other new sorts to the reader.

We now have a language with a fairly wide referential repertoire. Further extensions in a similar vein are possible, and relatively routine to define. For example, we could add ‘clock terms’ to the language, and define ‘clock structures’ in a manner that would allow “John reached the summit of Ruapehu at three o’clock” to be represented by

$$P(\text{three o'clock} \wedge \text{John reach the summit of Ruapehu}).$$

## 7 Concluding remarks

In this paper we have taken a rudimentary temporal language (namely Priorean tense logic) and, by progressively imposing sortal distinctions at the atomic level, have modeled various forms of temporal reference in natural language. The resulting systems are still very limited. For example, they throw no light on progressive aspect, adverbial modification or the role of the present tense in English. Nonetheless, we have made substantial progress using very simple tools, and it natural to ask whether sorting over a richer ontology could prove fruitful. To close the paper we consider the matter.

The most straightforward development of the present approach would be to sort an *interval based* system. A (fairly standard) choice of intervalic ontology is the following: an interval structure  $\mathbf{I}$  is a 4-tuple  $\langle T, <, \sqsubseteq, \circ \rangle$ , where  $T$  is a non-empty set, the set of intervals, and  $<$ ,  $\sqsubseteq$ , and  $\circ$  are all binary relations on  $T$ . These relations are thought of as the *precedence*, *subinterval* and *overlap* relations respectively. Of course it is necessary to specify further properties of these relations and their mutual interactions if these interval structures are to look temporally realistic. We won’t do so here but refer the reader to van Benthem [2] for a thorough discussion of the various options.

To talk about such interval structure we might use the following language. As primitive symbols we have a set VAR of atomic symbols, written  $p$ ,  $q$ ,  $r$  and so on, and the logical symbols  $\neg$ ,  $\wedge$ ,  $F$ ,  $P$ ,  $\downarrow$  and  $O$ . We make wffs from this collection of symbols in the obvious

way. To interpret this language we again use Kripke models. In the present setting a Kripke model  $\mathbf{M}$  is a pair  $\langle \mathbf{I}, V \rangle$  where  $\mathbf{I} (= \langle T, <, \sqsubseteq, \circ \rangle)$  is an interval structure and  $V$  is a valuation that assigns subsets of  $T$  to atomic symbols.

We interpret wffs in models in the expected fashion. The clauses for the atomic symbols, and for  $\neg$ ,  $\wedge$   $F$  and  $P$  are precisely as for Priorean languages. The two new clauses are as follows:

$$\begin{aligned} \mathbf{M} \models \downarrow\phi[t] & \quad \text{iff} \quad \exists t' (t' \sqsubseteq t \text{ and } \mathbf{M} \models \phi[t']) \\ \mathbf{M} \models O\phi[t] & \quad \text{iff} \quad \exists t' (t' \circ t \text{ and } \mathbf{M} \models \phi[t']) \end{aligned}$$

That is,  $\downarrow\phi$  is true at an interval  $t$  iff  $\phi$  is true at a subinterval  $t'$  of  $t$ ; and  $O\phi$  is true at an interval  $t$  iff  $\phi$  is true at an interval  $t'$  that overlaps  $t$ .

Now this new language is certainly more expressive than the Priorean language, but it also suffers from a familiar weakness: there is no mechanism for temporal reference. However (as should now be apparent) this weakness is easily removed. For example, let's add a second sort INOM, the sort of interval nominals, to the language. We'll write these new symbols as  $e$ ,  $d$ ,  $c$  and so on and insist that in any valuation these items are to be true at precisely one interval. Thus we could represent "John ran" by  $P(e \wedge \textit{John run})$ . This asserts that *John run* is true at some past interval, namely the one picked out by  $e$ . Note that it's a somewhat better representation than that offered by NTL: the use of  $e$  — which can be true over an *extended* interval — captures the fact that the speaker was referring to an extended period of time. Among other things this enables Comrie's modifications of Reichenbach's system to be accurately mirrored. More generally, the richer repertoire of interval operators, in combination with the various referential sorts, give rise to finer grained analyses of tense and temporal reference, and it becomes possible to give a (not wholly risible) analysis of progressive aspect.

Nonetheless, the improvements obtainable in the interval based setting seem to be, by and large, fairly obvious refinements of what has been achieved in the point based setting. A more exciting approach is to work with ontologies containing *both* interval and event structure. For example Blackburn, Gardent and de Rijke [6] work with *back and forth structures*. These are Kripke models consisting of an interval structure homomorphically linked to an event occurrence structure. From the point of view of natural language semantics such two sorted models are natural: the interval structure provides a realistic temporal ontology, the event structure provides a domain in which aspectual distinctions can be drawn, and the morphic link systematically correlates temporal information with event information. Moreover, from the point of view of abstract modal logic, the extension involved is fairly straightforward. There are obvious modal languages for dealing with such structures (essentially interval languages with additional operators) and their logical theory can be developed in the standard manner.

This work is still being developed, but already two points seem to be emerging. First, this ontology makes it possible to do interesting work on tense, aspect, adverbial modification and their interaction in a modal setting. Second, the natural way to exploit the ontological richness seems to be by working with languages of *low* complexity. That is, rather than working with languages with powerful collections of operators, it becomes natural to push ever more information into 'structured atomic symbols' which are sensitive to sortal information. In a sense one takes the sorting strategy to its logical conclusion: the sorting does the bulk of the work, the operators relatively little.

Only further work can determine just how far this approach can be pushed. In the meantime, I'll simply note that (as we have seen in this paper) a surprising amount of analysis can be done even within the point based setting. Although much criticised, Prior's work contains a kernel of technical and intuitive insights that is hard to dismiss. These insights

can be extended in a natural manner to many current concerns of temporal semantics.

**Acknowledgements** I would like to thank Inge Bethke, Claire Gardent, Hans Kamp, Alex Lascarides and Jerry Seligman for their advice, criticism and (not least) their patience in discussing the ideas of this paper. The comments of the two anonymous referees on the penultimate version were also very helpful.

*Department of Philosophy,  
Utrecht University*

## References

- [1] Benthem, J. v.: 1983, Correspondence Theory, in *Handbook of Philosophical Logic 2*, edited by D. Gabbay and F. Guentner, Kluwer Academic Publishers.
- [2] Benthem, J. v.: 1991, *The Logic of Time*, 2nd edition, Kluwer Academic Publishers.
- [3] Blackburn, P.: 1990, *Nominal Tense Logic and other Sorted Intensional Frameworks*, PhD Thesis, Centre for Cognitive Science, University of Edinburgh, Scotland.
- [4] Blackburn, P.: 1993, Nominal Tense Logic, *Notre Dame Journal of Formal Logic*, **34**, pages 56–83.
- [5] Blackburn, P.: 1993, Fine Grained Theories of Time, in *Working Papers of the Fourth International Workshop on Space, Time and Movement, Chateau de Bonas*, edited by Aurnague, Borillo, Borillo, and Bras-Grivart
- [6] Blackburn, P., Gardent, C., de Rijke, M.: 1994, Back and Forth through Time and Events (Extended Abstract), to appear in *Proceedings of the Ninth Amsterdam Colloquium*, ILLC, Amsterdam.
- [7] Bull, R.: 1970, An Approach to Tense Logic, *Theoria*, **12**, pages 171–182.
- [8] Comrie, B.: 1981, On Reichenbach’s Approach to Tense, in *Papers from the Seventeenth Regional Meeting, Chicago Linguistics Society*, edited by Hendrick *et al.*, pages 24–30.
- [9] Comrie, B.: 1985, *Tense*, Cambridge University Press.
- [10] Gargov, G., Passy, S., Tinchev, T.: 1987, Modal Environment for Boolean Speculations (preliminary report), in *Mathematical Logic and its Applications*, edited by D. Skordev, Plenum Press.
- [11] Gargov, G., Goranko, V.: 1993, Modal Logic with Names, *Journal of Philosophical Logic*, **22**, pages 607–636.
- [12] Hinrichs, E.: 1982, Temporal Anaphora in Discourses of English, paper presented at the Ohio State University Conference on the Semantics of Tense and Aspect in Discourse.
- [13] Kamp, H.: 1971, Formal properties of ‘now’, *Theoria*, **37**, pages 227–273.
- [14] Kamp, H.: 1981, A Theory of Truth and Semantic Representation, in *Formal Methods in the Study of Language, Part 1*, edited by J. Groenendijk, T. Janssen, and M. Stokhof, Mathematisch Centrum, Amsterdam, pages 277–322.

- [15] Kamp, H., Rohrer, C.: 1983, Tense in Texts, in *Meaning, Use, and Interpretation of Language*, edited by Bäuerle, Schwarze, and von Stechow, Walter de Gruyter, pages 250–269.
- [16] Kaplan, D.: 1977, Demonstratives: An Essay on the Semantics, Logic, Metaphysics and Epistemology of Demonstratives and Other Indexicals, Draft No 2, manuscript, UCLA.
- [17] Kaplan, D.: 1979, On the Logic of Demonstratives, in *Contemporary Perspectives in the Philosophy of Language*, edited by French, Uehling and Wettstein, pages 401–412, Minneapolis: University of Minnesota Press.
- [18] Montague, R.: 1970, Universal Grammar, in *Formal Philosophy. Selected Papers of Richard Montague*, edited by Richmond Thomason.
- [19] Oversteegen, E.: 1989, *Tracking Time. A Proposal for the representation of Temporal Expressions in Dutch*. Proefschrift, Rijksuniversiteit te Utrecht.
- [20] Passy, S., Tinchev, T.: 1985, PDL with Data Constants. *Information Processing Letters* **20**, pages 35–41.
- [21] Passy, S., Tinchev, T.: 1991, An Essay in Combinatory Dynamic Logic. *Information and Computation*, **93**, pages 263–332.
- [22] Partee, B.: 1984, Nominal and Temporal Anaphora, *Linguistics and Philosophy*, **7**, pages 243–286.
- [23] Prior, A.: 1967, *Past, Present and Future*, Oxford University Press.
- [24] Prior, A.: 1968, “Now”, *Nous*, **2**, pages 101–119.
- [25] Reichenbach, H.: 1947, *Elements of Symbolic Logic*. Random House: New York.