A Dynamic Modal Arrow Logic for the Analysis of Aspectual Phenomena in Natural Language

Ralf Naumann[1]/Rainer Osswald[2]
Seminar für Allgemeine Sprachwissenschaft[1]
University of Düsseldorf
Applied Computer Science[2]
University of Hagen

Abstract

In Naumann (1998b,1999a) and Naumann/Mori (1998) a *Dynamic Event Semantics* (DES) for the analysis of aspectual phenomena was developed that is basically a many-sorted type theory in the sense of Gallin. DES is based on the intuition that non-stative verbs express changes. A change can be conceived of either as an object (event, action) or as a transformation of state. The first perspective is captured in Event Semantics and in Arrow Logic, whereas the second perspective underlies Dynamic Logic. In models for DES the first perspective of changes as objects is accounted for by an eventuality (sub-)structure **E** whereas the second perspective is captured by a transition (sub-)structure **S**.

As noted in the above mentioned articles, the aspectual properties of expressions only depend on the properties the execution-sequences of events have with respect to the result they bring about. This raises the question of whether it is possible to express the aspectual properties of linguistic expressions in weaker logics. In this paper two logics L and L^h are defined. L combines (fragments of) Arrow Logic and Dynamic Modal Logic. It is two-sorted: besides s-formulas there are e-procedures. The latter are primary because verbs and their projections are translated as e-procedures. In L it is not possible to express the dependency of results that are brought about by an event e on particular objects that participate in e. This is possible in the hybrid extension L^h of L. The resulting theory is applied to modification of VPs with directional PPs and the temporal conjunction 'until', the interpretations of which are sensitive to the aspectual properties of verbs.

1 Data and Evidence

Modifiability with *in*-, *for*- and *at*-adverbials of an expression depends on the underlying verb, witness the data in (1).

(1) a. John ate an apple in ten minutes/*for ten minutes/*at three.
b. Mary pushed the cart *in ten minutes/for ten minutes/*at three.
c. Peter was ill *in ten minutes/for two weeks/at three.
d. Susan reached the station in ten minutes/*for ten minutes/at three.

These differences are the basis for the Vendler-classification of verbs: Accomplishments ('eat'), Activities ('push'), stative verbs ('be ill') and Achievements ('reach'). This classification is not exhaustive as the examples in (2) show.

(2) a. Bill knocked at the door *in ten minutes/*for ten minutes/at three.

b. Mary painted the wall in ten minutes/for ten minutes/*at three.

c. Mary gave Bill the book in ten minutes/*for ten minutes/at three.

Whereas a verb like 'knock' admits neither of modification with 'in' nor of that with 'for'-adverbials (the latter yields an iterative interpretation), for a verb like 'paint' both adverbials are admissible. In Naumann (1999a,c) 'knock' and 'paint' are classified as Point- and Proc-Acco-verbs, respectively. In contrast to Achievement-verbs, a resultative reading of the (present) perfect, a so-called Perfect of Result, is not possible for Point-verbs: 'John has knocked at the door' can only be interpreted as an experiential perfect, i.e. in the sense of John knocked at the door at least once in the past.[1] Transfer-verbs differ from Accomplishment-verbs with respect to the progressive. Whereas the inference from 'John was eating an apple' to 'John ate part of the apple' (= there was an event of eating) is valid, this does not hold for 'give': Mary was giving Bill the book' does not imply that there was an event of giving because from this it would already follow that Bill got the book from Mary, i.e. the inference from 'Mary was giving Bill the book' to 'Mary gave Bill the book' would be valid, contrary to the evidence. Similarly to Accomplishment-verbs, modification with an 'in'-adverbial of expressions containing a Transfer-verb is not (quasi)-synonymous with a corresponding sentence using 'after' instead of 'in': Mary gave Bill the book in ten minutes $\neq$ Mary gave Bill the book after ten minutes; John ate an apple in ten minutes $\neq$ John ate an apple after ten minutes. For Achievement-verbs, on the other hand, the two modifications are quasi-synonymous: Susan reached the station in ten minutes = Susan reached the station after ten minutes.

The behaviour with respect to temporal adverbials can be changed by various types of modifying expressions like directional PPs, (3a), secondary predication, (3b), and 'for'-adverbials, (3c), (4)-(5).

(3) a. Mary pushed the cart to the station in ten minutes/*for ten minutes/*at three.

b. Mary painted the wall blue in ten minutes/*for ten minutes/*at three.

c. John pushed a cart for twenty minutes.

(4) a. Mary was pushing the cart $\Rightarrow$ Mary pushed the cart

b. Mary was pushing the cart for twenty minutes $\not\Rightarrow$ Mary pushed the cart for twenty minutes.

(5) a. Mary almost pushed the cart $\Rightarrow$ Mary did not push the cart (but, say, had the intention to do it)

b. Mary almost pushed the cart for twenty minutes

$\Rightarrow$ Mary did not push the cart (but, say, had the intention to do it)

[1] For an analysis of the Present Perfect in English see Naumann (1999b).

⇒ Mary pushed the cart, say, for eighteen minutes

Whereas (1b) is an Activity-expression, (3a) is an Accomplishment-expression. A similar argument applies to (2b) and (3b): depictive adjectives change the Proc-Acco-expression (2b) into an Accomplishment-expression, (3b). The examples in (3c-5) show that modification with a 'for'-adverbial changes the aspectual behaviour too. For the modified expression one gets both a so-called imperfective paradox, (4), and two readings for modification with 'almost', (5). Examples like these show that the aspectual behaviour of an expression is only partly determined by the underlying verb. Another determining factor are modifying expressions. Furthermore, certain expressions impose aspectual restrictions on their use. An example is the temporal conjunction 'until'. It imposes an aspectual restriction on the expression in the main clause: only stative- and Activity-expressions are admitted, (6a). Both Accomplishment- and Achievement-expressions are excluded, (6b).

(6) a. John was ill/worked on the article until Mary arrived.

b. *Bill wrote the article/reached the station until Mary arrived.

The task, then, consists in finding interpretations of verbs in the lexicon on the basis of which (i) the differences in aspectual behaviour at the lexical level and (ii) the process of aspectual composition (changes in aspectual behaviour triggered by modifying expressions) can be explained.

2 Dynamic Event Semantics

Changes as Objects and Changes as Transformations of States Dynamic Event Semantics (DES), Naumann (1998,1999) and Naumann/Mori (1998), is based on the intuition that non-stative verbs like 'eat' express changes. The intuitive notion of a change comprises at least two aspects that are complementary to each other: (i) something (an object: action, event) which brings about the change; (ii) something (a result) which is brought about by the change and which did not hold before the change occurred. In (i) 'change' is understood as the result that is brought about, i.e. in the sense that is captured by (ii), whereas in (ii) 'change' is meant as the object that brings about the result. The second aspect can be described as a transformation of state (TS). Before the change occurred, the world was in a particular state, say s, at which some result Q did not hold, whereas after the change has occurred, the world is in a state s' at which Q does hold. E.g., the eating of an apple is an event of type eating if conceived of as an object. On the perspective of a change as a TS one gets: a state s at which there is a complete apple is transformed into a state s' where the apple no longer exists ($Q =$ the apple does not exist). For the pushing of a cart the change as an object is an event of type pushing whereas the transformation can be described as 'a state s is transformed into a state s' such that relative to s the cart traversed a non-empty path ($= Q$)'. The first perspective, changes as objects, is captured in Event Semantics as well as in Arrow Logic, whereas the second perspective, changes as TS, is captured in Dynamic Logic (or Temporal Logic).

Verb Classification in Dynamic Event Semantics. The double perspective on the intuitive notion of a change either as an object or as a transformation of state that brings about a result is modeled in DES by having both an eventuality-structure **E** with an underlying domain E of events and a transition-structure **S** with an underlying domain S of states. The aspect of a change as an object is captured at the level of E whereas the aspect as a transformation of state is captured at the level of S. The elements of S are basic objects without structure that are ordered by a strict, linear ordering $<_S$. The domain E is related to the domain S by two functions $\alpha : E \to S$ and $\omega : E \to S$ that assign to each $e \in E$ its beginning-point $\alpha(e)$ and end-point $\omega(e)$, respectively. Together, α and ω determine the execution-sequence $\tau(e)$ of e : $\tau(e) = \{s \mid \alpha(e) \leq s \leq \omega(e)\} = (\alpha(e), \omega(e))$. The domain E is structured by a part-of relation $\leq_E$ in terms of which a composition-relation C is defined that is required to be associative.

(7) $C = \{(e, e_1, e_2) \mid e_1 \leq_E e \wedge e_2 \leq_E e \wedge \omega(e_1) = \alpha(e_2) \wedge \alpha(e) = \alpha(e_1) \wedge \omega(e) = \omega(e_2)\}$

C corresponds to the operation of (sequential) composition in Dynamic Logic (DL). In terms of C the following relations can be defined.

(8) a. $R_B = \{(e, e') \mid \exists e'' : Ce', ee''\}$
b. $R_{B^*} = \{(e, e') \mid Be, e' \wedge e \neq e'\}$
c. $R_{E'} = \{(e, e') \mid \exists e'' : Ce, e''e'\}$

The *interior* $\tau^*(e)$ *of* $\tau(e)$ is defined as $\tau^*(e) = \tau(e) - \{\alpha(e), \omega(e)\}$. A sort of *diagonal* on E is defined as $I = \{e \mid \alpha(e) = \omega(e)\}$. The domain E is sorted by the label set $VERB$: for $v \in VERB$, $P_v \subseteq E$ is the set of all events of type v, e.g. the set of all eating events if $v = eat$. The elements of the set $\{P_v \mid v \in VERB\}$ are called *basic* event-types. A verb $v \in VERB$ is always interpreted with respect to P_v, i.e., P_v is the event-type *corresponding* to v. Results Q are subsets of S. A TS is a pair $< Q, < s, s' >>$ with (s, s') a finite sequence of states and Q a result such that $s \notin Q$ and $s' \in Q$. An event-type P_v determines for each of its elements e a set $Res(P_v, e)$ of results that e can possibly bring about. The relationship between an event e and TSs is therefore in general not one-one but one-many: e can bring about more than one result. Different types of results can be distinguished by the way they are evaluated on the execution sequence of events $e \in P_v$. Basically, three different types of results are distinguished: (i) a result Q is *s-minimal* just in case it holds at all states of the execution sequence of e in between e's beginning point and a state s of the execution sequence at which Q holds; (ii) a result Q is *w-minimal* if it holds at all states of the execution sequence of e that are end points of initial stages e' of e that are of type P_v and (iii) a result Q is *maximal* if it only holds at the end point of e (if it holds at all on e's execution sequence). For an event of type 'John eat a fish' examples for the three types of results are: s-minimal: the results brought about by the initial actions by John (e.g. John's opening and closing his mouth, putting a piece of the fish into his mouth)[2]; w-minimal: the results brought about by swallowing part of the fish: partial decrease in the

[2]Note that these actions do not necessarily involve the fish.

mass of the fish due to the swallowing; maximal: total decrease of the mass of the fish: its mass is zero. The set $Res(P_v, e)$ is temporally ordered: the s-minimal results are brought about first, followed by the w-minimal results and the maximal results are brought about last. It is therefore possible to define an ordering $\leq_v$ on $Res(P_v, e)$ based on the relation 'not before': $Q \leq_v Q'$ if Q' is brought about not before Q on the execution sequence of e. Intuitively, Q' is brought about not before Q (relative to an event-type P_v) just in case whenever Q' holds at some point s of the execution sequence $\tau(e)$ of an event $e \in P_v$, then there is a point $s' \in \tau(e)$ with $s' \leq s$ at which Q holds, Latrouite/Naumann 1999a,b.

Verbs v are classified on the basis of (i) the *types* of results that are determined by the corresponding event-types P_v, (ii) the *sort* to which the event-types P_v belong and (iii) the *sort* of the results that are determined and which are maximal elements of $Res(P_v, e)$ with respect to the ordering $\leq_v$ for $e \in P_v$. E.g., Accomplishment-verbs like 'eat' determine all three types of results whereas Activity-verbs like 'push' do not determine a maximal result but only an s- and a w-minimal one. Transfer-verbs determine two maximal results, whereas Point- and Achievement-verbs determine only one. Three different sorts of event-types are distinguished: an event-type P_v is *P-atomic* if no proper initial stage (prefix) e' of an event e belonging to P_v is of this type too, (9a); an event-type P_v is *instantaneous* if each of its elements has an execution sequence that consists of a single state (i.e., the beginning point is identical to the end point), (9b), Naumann 1999b. Instantaneous and P-atomic event-types together form the *atomic* event-types. Finally, an event-type P_v is non-atomic if it is not P-atomic and if the execution sequence of each of its elements is not a singleton. Event-types of sort Accomplishment and Activity are non-atomic, (9c).

(9) a. $\forall P[P - Atomic(P) \Leftrightarrow \forall e[e \in P \rightarrow \neg\exists e'[prefix(e', e) \wedge e' \in P]]]$
b. $\forall P[Instant(P) \Leftrightarrow \forall e[e \in P \rightarrow \alpha(e) = \omega(e)]]$
c. $\forall P[Non - Atomic(P) \Leftrightarrow \neg P - Atomic(P) \wedge \forall e[e \in P \rightarrow \alpha(e) <_S \omega(e)]]$

Examples for P-atomic event-types are those corresponding to Transfer-verbs like 'give' and 'buy'. Point- and Achievement-verbs like 'hit' and 'reach', respectively, correspond to instantaneous event-types. Events belonging to an instantaneous event-type presuppose other events of which they are right boundaries (for details see Naumann (1999a)).[3]

Results that are of the same type can differ with respect to the *sort* to which they belong. Consider events of type 'John eat an apple' and 'Mary sing a song', respectively. Both expressions are of type Accomplishment and therefore define a maximal result which can be paraphrased as: 'the mass of the apple is zero' and 'relative to the beginning-point of the event the partition of the song was went through completely'. The difference between the two results is that the former but not the latter continues to hold after the event terminated. The former result is therefore *state-related* whereas the latter is *event-related*. A result is state-related if it continues to hold after the end of an event until it is undone (by some other (non-stative) event). A result is event-related if it is true

[3]Furthermore, for events belonging to an instantaneous event-type as well as for events that are denoted by unaccusative forms like 'The bottle emptied' the results are evaluated with respect to a presupposed event.

only during the the executions of (non-stative) events (for details see Naumann (1999a)). The classification based on the three criteria is given in Table 1 ((i) = type of maximal elements of $Res(P_v, e)$ relative to $\leq_v$, (ii) = sort of event-type and (iii) = sort of (i)).

Table 1	(i)	(ii)	(iii)
Acco.	maximal	non-atomic	state- or event-related
Act.	w-minimal	non-atomic	event-related
Transfer	maximal	P-atomic	state-related
Point	maximal	instantaneous	event-related
Achievement	maximal	instantaneous	state-related
stative verbs	s-minimal	non-atomic	state-related

2.1 The Relationship between Changes as Objects and Changes as Transformations of States

Recall that $Res(P_v, e)$ is the set of results assigned to e by P_v. In general, e is not required to bring about each element from this set. The question of which results must be realized by e must be split into the following two.

(i) What (types of) results must be brought about by an event e in order to be of type P_v, i.e., what requirements on the results are imposed by P_v?

(ii) What (types of) results must be brought about by an event $e \in P_v$ that is denoted by the verb v in the lexicon, i.e., what requirements on the results are imposed by (the interpretation of) v in the lexicon?

Consider an event of type eating with John as Actor and an apple as stuff that is eaten. If only the s-minimal result is brought about, e.g., John opened his mouth, the corresponding event is not of type 'eat', i.e., it does not belong to P_{eat}. What is minimally required is that John swallow at least part of the apple. This result corresponds to the w-minimal one, i.e. to a partial decrease in the mass of the apple. P_{eat} does not impose any further requirements, in particular it it not required that the maximal result (the mass of the fish is zero) be brought about. From this it follows that an event $e \in P_{eat}$ need not bring about all results from $Res(P_v, e)$. On the other hand, the w-minimal result is not sufficient for e to be an element of the set of events of type eating denoted by 'eat' in the lexicon. 'John ate an apple' is true only if John ate the apple completely and not only an arbitrary part of it. Thus, for 'eat' the requirement imposed on the results that must be brought about is stronger than the one imposed by P_{eat}: P_{eat} : w-minimal result; (interpretation of) 'eat': maximal result.

For an event of type push with Mary as Actor and a cart as object that is pushed, the minimal requirement imposed by P_{push} is the same as that imposed by P_{eat}: the w-minimal result must be brought about (the cart traversed a non-empty path). The s-minimal results corresponding to the initial actions by John only count as attempts at pushing the cart but do not in themselves constitute an event of this type. In this case the interpretation of 'push' in the lexicon cannot strengthen the condition imposed by P_{push} because this event-type does not determine a maximal result, i.e., the maximum of $Res(P_{push}, e)$

with respect to $\leq_{push}$ is the w-minimal result. Consequently, for Activity-verbs v like 'push' the requirement imposed by P_v is identical to that imposed by the interpretation of v.

What is common to the requirement imposed in English by Accomplishment- and Activity-verbs in the lexicon is that the maximum of $Res(P_v, e)$ with respect to the ordering $\leq_v$ must be brought about. This condition is equivalent to the requirement that each element from $Res(P_v, e)$ must be brought about. A similar argument applies to the other classes of verbs like Transfer- or Point-verbs discussed above. In each case the interpretation of a verb v belonging to one of these classes requires an event $e \in P_v$ to bring about the maximum of $Res(P_v, e)$ with respect to $\leq_v$ and therefore each element from $Res(P_v, e)$. The type of the maximal elements of $Res(P_v, e)$ depends on the aspectual class to which v belongs. For verbs of type Accomplishment, Transfer, Point and Achievement this is the type of maximal results whereas for Activity-verbs and stative verbs the maximum is w-minimal and s-maximal, respectively.

3 The Analysis

3.1 Dynamic Nucleus Structures and Dynamic Modes

3.1.1 Dynamic Modes

In section (2) it was shown how various aspectual classes can be distinguished in terms of four criteria that are all related to the dynamic-temporal structure of events of the types belonging to one of these classes. A first criterion is the types of results that are determined by P_v. Important for the classification is the type of the maximal elements of $Res(P_v, e)$ with respect to the ordering $\leq_v$. Types of results differ with respect to the way they are evaluated on the execution-sequences of events. E.g., maximal results only hold at the end-point (if at all) whereas w-minimal results hold at the end-points of all initial stages e' of e that are of type P_v. As maximal elements of $Res(P_v, e)$ are required to hold at $\omega(e)$ by the interpretation of v in the lexicon, it follows that they are evaluated on $\tau(e)$ in the way determined by their type if they belong to the subset of P_v denoted by v. This way corresponds to a dynamic mode from Dynamic Modal Logic (DML), de Rijke (1993). DML is a two-sorted logic that is based on a two-level architecture of formulas and procedures (programs), together with two types of operations modeling the interaction between the two sorts of expressions: dynamic modes (DM) map formulas to procedures and static projections map procedures to formulas. A DM is interpreted as an operation from $\wp(S)$, the power set of the underlying domain of states, to $Re(S)$, the set of binary relations on S: $\lambda Q \lambda s s' [\![DM]\!](Q)(s)(s')$. Two examples for dynamic modes are given in (10).

(10) a. $R_{Min-BEC}(Q) = \{(s, s') \in S \times S \mid s <_S s' \wedge \neg Q(s) \wedge Q(s') \wedge \forall s''[s <_S s'' <_S s' \rightarrow \neg Q(s'')]\}$

b. $R_{Con-BEC}(Q) = \{(s, s') \in S \times S \mid s <_S s' \wedge \neg Q(s) \wedge Q(s') \wedge \forall s''[s <_S s'' <_S s' \rightarrow Q(s'')]\}$

The crucial observation is that dynamic modes exactly fit the description of how changes as transformations of states (TS) are brought about given above in section (2). There a TS was defined as a pair $< Q, < s, s' >>$ consisting of a result $Q \subseteq S$ and an element from $S \times S$ such that Q is false at s and true at s'. For a given dynamic mode DM, $< Q, < s, s' >>$ is either an element of its denotation or not. If Q is a result that must be brought about by an event $e \in P_v$, in particular if $Q \in Res_{\max}(P_v, e) = \{Q \mid Q \in Res(P_v, e) \wedge \forall Q'[Q' \in Res(P_v, e) \rightarrow Q' \leq_v Q]\}$, and if Q is of type i, for i one of the three basic types, Q is evaluated on $\tau(e)$ in the way determined by i. If this way is expressed by the dynamic mode DM and if $< s, s' > = \tau(e)$, one gets: $< Q, < s, s' >> \in [\![DM]\!]$. Thus, Q corresponds to the result brought about, whereas $[\![DM]\!]$ (or $[\![DM]\!](Q)$) corresponds to the way the result is brought about. E.g., for the type of change expressed by an Accomplishment-verb v, a maximal result Q assigned to an $e \in P_v$ is mapped to a binary relation such that Q only holds at the output-state s' and at no other state of the execution-sequence $\tau(e)$.

From the present perspective, the disadvantage of DML consists in the lack of a separate domain E of events (actions). DML-models are one sorted: there is only one domain S of states. As a consequence, transitions are not basic objects but are interpreted as binary relations on S such that the aspect of a change as an object cannot be modeled. Models for Arrow Logic, on the other hand, are two sorted: transitions are not interpreted as (elements of) binary relations but as elements of a separate domain of arrows. In models for DES defined in section (2) above both perspectives are combined because there is both a domain E of events and a domain S of states that are systematically related to each other.

The relationship between dynamic modes and the domain E of events can be defined as follows. Recall that each event $e \in E$ belongs to an event-type $P_v \subseteq E$ that is an element of the basic set $\{P_v \mid v \in VERB\}$ for $VERB$ a subset of the verbs in English. Each P_v induces the binary relation R_v on $S \times S$.

(11) $R_v = \{(s, s') \in S \times S \mid \exists e[e \in P_v \wedge \tau(e) = (s, s')]\}$

Compared to DL, R_v can be interpreted as corresponding to the binary relation R_π denoted by a (basic) program letter π in the following sense: each $e \in P_v$ corresponds to an element $(s, s') \in R_v$ and vice versa. On this perspective, e is the object (change) that brings about the transition from s to s'. Each P_v can therefore be seen as a kind of accessibility-relation corresponding to R_v.

If $(s, s') \in R_v$, the minimal requirement on the result imposed by P_v is satisfied. As was shown above, this requirement can be weaker than the one imposed by the interpretation of v. The relation corresponding to this interpretation is R_v^*.

(12) $R_v^* = \{(s, s') \in S \times S \mid \exists e \exists Q[e \in P_v \wedge \tau(e) = (s, s') \wedge Q \in Res_{\max}(P_v, e) \wedge [\![DM]\!](Q)(s)(s')]\}$

An element of R_v belongs to R_v^* if an element Q of the set of maximal elements of $Res(P_v, e)$ with respect to $\leq_v$ is brought about, i.e. holds at $\omega(e)$. DM is the way Q is evaluated on $\tau(e)$.

As the elements (s, s') of $[\![DM]\!](Q)$ one is interested in are the execution sequences of events, a dynamic mode can be interpreted as an operation from $\wp(S)$ to $\wp(E)$. Yet, this is in general too simple because a type of result can impose restrictions that depend on a basic event-type P_v. This is the case for w-minimal results. In DES dynamic modes are therefore interpreted as operations from $\wp(E) \times \wp(S)$ to $\wp(E)$. Three examples are given in (13).

(13) a. $R_{Min-BEC_{<_S}} = \lambda P \lambda Q \lambda e[e \in P \wedge \alpha(e) \notin Q \wedge \omega(e) \in Q \wedge \forall s[\alpha(e) <_S s <_S \omega(e) \rightarrow s \notin Q]]$

b. $R_{Con-BEC_{<_S}} = \lambda P \lambda Q \lambda e[e \in P \wedge \alpha(e) \notin Q \wedge \omega(e) \in Q \wedge \forall s[\alpha(e) <_S s <_S \omega(e) \rightarrow s \in Q]]$

c. $R_{Con-BEC_v} = \lambda P \lambda Q \lambda e[e \in P \wedge \alpha(e) \notin Q \wedge \omega(e) \in Q \wedge \forall e' \forall s[\alpha(e) <_S s <_S \omega(e) \wedge prefix(e', e) \wedge \omega(e') = s \wedge e' \in P \rightarrow s \notin Q]]$

Note that all three modes require that Q be evaluated differently at the beginning- and the end-point of e, in accordance with the definition of TS given above. From this it follows that they cannot be used for the interpretation of verbs v whose corresponding event-types P_v are instantaneous because the execution sequences of events $e \in P_v$ are singletons such that no TS corresponds to $\tau(e)$. This possibility is related to the fact that the three basic types of results do not exclude each other. What types are indistinguishable for an event-type P_v depends on the sort to which it belongs. For instantaneous event-types, all three types coincide. For P-atomic event-types, the distinction between w-minimal and maximal results collapses: each w-minimal result is maximal and vice-versa. This property distinguishes the (non s-minimal) results assigned to verbs of this class from those assigned to verbs belonging to the class of Accomplishments or Activities where neither w-minimal results are maximal nor maximal results w-minimal. Maximal results that are not w-minimal are called *strongly maximal* (*s-maximal*) whereas maximal results that are also w-minimal are called *weakly maximal* (*w-maximal*). Results that are w-minimal but not maximal are called *w*-minimal*, Latrouite/Naumann (1999a,b).

The problem of defining dynamic modes for instantaneous event-types can be solved by distinguishing two subtypes of s-minimal results: those that are false at $\alpha(e)$ and those which are true at $\alpha(e)$. The mode corresponding to the former, s-minimal$_1$, is $R_{Con-BEC_{<_S}}$, whereas the mode corresponding to the latter, s-minimal$_2$, is $R_{HOLD_{<_S}}$, defined in (14).

(14) $R_{HOLD_{<_S}} = \lambda P \lambda Q \lambda e[e \in P \wedge \forall s[\alpha(e) \leq_S s \leq_S \omega(e) \rightarrow s \in Q]]$

Results that are s-minimal$_2$ characterize both Point-, Achievement- and stative verbs. For the two former classes these results are, in addition, s-minimal, w-minimal and maximal, whereas for the latter class these results are s-minimal and w-minimal but not maximal.

The correspondence between types of results and dynamic modes is given in Table 2.

s-minimal$_1$	$R_{Con-BEC_{<_S}}$
s-minimal$_2$	$R_{HOLD_{<_S}}$
w*-minimal	$R_{Con-BEC_v}$
w-maximal	$R_{Min-BEC_{<_S}}$
s-maximal	$R_{Min-BEC_{<_S}}$

Table 2

From this correspondence between types of results and dynamic modes the following partial characterization of the aspectual classes can be derived.

Accomplishment	$R_{Min-BEC_{<_S}}$	
Activity	$R_{Con-BEC_v}$	
Point	$R_{HOLD_{<_S}}$	
Achievement	$R_{HOLD_{<_S}}$	
Transfer	$R_{Min-BEC_{<_S}}$	
stative verbs	$R_{HOLD_{<_S}}$	Table 3

At this level there is no distinction between (i) Accomplishments- and Transfer-verbs and (ii) between stative verbs, Points and Achievements. These classes can be distinguished if in addition the following two further criteria discussed in section (2) are used: (a) sort of event-type and (b) sort of result. Although Accomplishment- and Transfer-verbs do not differ with respect to the way a $Q \in Res_{\max}(P_v, e)$ is evaluated on the execution sequences of events $e \in P_v$, they differ with respect to the property of being strongly non-closed under (proper) prefixes of the same event-type. For event-types P_v, v of type Transfer, (15a) holds, whereas it fails to hold for v of type Accomplishment.

(15) a. $P_v \cap cl(P_v) = \emptyset$

b. $cl(P_v) = \{e \mid \exists e'[e' \in P_v \wedge prefix(e', e) \wedge e' \in P_v]\}$

(16a) is captured by the dynamic mode $R_{Min-BEC^*_{<_S}}$.

(16) $R_{Min-BEC^*_{<_S}} = \lambda P \lambda Q \lambda e[R_{Min-BEC_{<_S}}(P)(Q)(e) \wedge \forall e'[prefix(e', e) \rightarrow e' \notin P]]$

The execution sequences of events belonging to an event-type P_v with v of type stative verb are non-singletons whereas those of events $e \in P_v$ with v either of type Point or of type Achievement are singletons. This difference can be captured by distinguishing the two $HOLD$-modes R_{HOLD^1} and R_{HOLD^2}.

(17) a. $R_{HOLD^1} = \lambda P \lambda Q \lambda e[R_{HOLD_{<_S}}(P)(Q)(e) \wedge \alpha(e) = \omega(e)]$

b. $R_{HOLD^2} = \lambda P \lambda Q \lambda e[R_{HOLD_{<_S}}(P)(Q)(e) \wedge \alpha(e) <_S \omega(e)]$

As was shown in sections (1) and (2), Achievement-verbs differ from Point-verbs in the sort of the maximal element from $Res(P_v, e)$ that is defined. For Achievement-verbs this result is state-related, i.e., it continues to hold after the end of the event until it is undone. For Point-verbs, on the other hand, this result is event-related, i.e., it is false after the end of the event. This difference shows up at the linguistic level in the possibility of a resultative reading for the Perfect: 'John has reached the station' has a resultative reading, implying that he is still at the station at speech-time. For 'John has knocked at the door' there is no such reading because the knocking does not bring about any result that holds beyond the end of the event. This distinction can be captured by defining variants of dynamic modes which require the result Q to hold an indefinite time after the end of the event. The general scheme of this variant is defined in (18).

(18) $DM^h = \lambda P \lambda Q \lambda e \exists e' \exists e''[DM(P)(Q)(e) \wedge HOLD(E)(Q)(e') \wedge Ce'', ee']$

The aspectual classes from section (1) can now be characterized in terms of dynamic modes as given in Table 4.

Accomplishment	$R_{Min-BEC_{<S}}$
Activity	$R_{Con-BEC_v}$
Point	R_{OCCUR}
Achievement	R_{END}
Transfer	$(R_{Min-BEC^*_{<S}})^h$
stative verbs	R_{HOLD^2}

Table 4

The dynamic modes R_{OCCUR} and R_{END} are defined in (19). They make explicit that an event $e \in P_v$ for v either of type Point or of type Achievement is the right bounary of another (presupposed) event e'. This is defined by the mode $HOLD^{1+}$.

(19) a. $R_{OCCUR} = R_{HOLD^{1+}}$

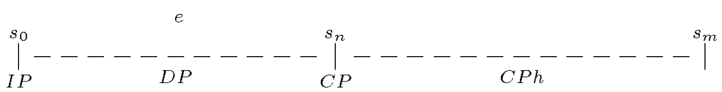
b. $R_{END} = (R_{HOLD^{1+}})^h$

c. $R_{HOLD^{1+}} = \lambda P \lambda Q \lambda e \exists e' \exists e''$
$[R_{HOLD^1}(P)(Q)(e) \wedge C e', e'', e \wedge R_{Min-BEC^*_{<S}}(E)(Q)(e')]$

3.1.2 Dynamic Nucleus-Structures

In the last section aspectual classes were defined in terms of the dynamic-temporal structure of events. The dynamic-temporal structure of an event can be depicted in terms of a *Dynamic Nucleus Structure* (DNS), Moens/Steedman 1988. A DNS consists of four parts: IP (Inception-Point), DP (Development-Portion), CP (Culmination-Point) and CPh (Consequent-Phase).

Dynamic Nucleus Structure (Moens/Steedman 1988)



The DNS characterizing an aspectual class AC corresponds to the execution of an event $e \in P_v$, with $v \in AC$, that brings about all results from $Res(P_v, e)$. The IP and the CP can therefore be identified with $\alpha(e)$ and $\omega(e)$, respectively. The DNS of an aspectual class is determined by (i) the way a result $Q \in Res(P_v, e)$ is evaluated on the four parts and (ii) the way proper prefixes e' of e are related to P_v, i.e., whether they can be of type P_v or not. All DNS agree on the way Q is evaluated at the CP: it is true at this point. For the other parts one gets:

(i) if Q is true at the IP, Q is strongly s-minimal (Point, Achievement, stative verb)

a. if Q is false on the DP, it is maximal (Accomplishment, Transfer)

b. if Q is true and false on the DP, the DP is empty (Point, Achievement)

(ii) if Q is true on the CPh, Q is state-related (Transfer, Achievement)

(iii) if Q is false on the CPh, Q is event-related (Activity, Point, stative verbs)

If P_v is P-atomic and therefore strongly non-closed under prefixes of type, the answer to (ii) is *no*, otherwise *yes*.

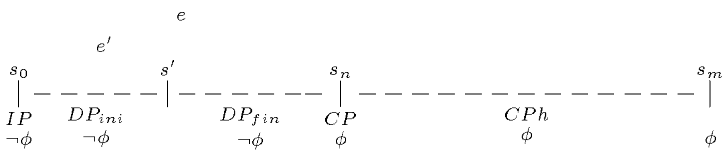
Table 5 summarizes the results.

	IP	DP	CPh	$e' \in P_v$
Acco	false	false	false/true	yes
Act	false	true	false	yes
Transfer	false	false	true	no
Point	true	false$\wedge$true	false	yes
Ach	true	false$\wedge$true	true	yes
stative verb	true	true	false	yes

Table 5

Below the DNS of the aspectual classes discussed in this paper are given.

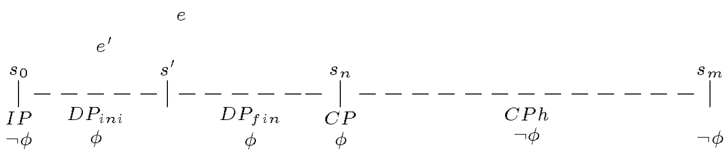
Dynamic Nucleus Structure of Accomplishment-verbs ('eat')



$\Rightarrow$ ϕ expresses the maximal result (the mass of the object denoted by the internal argument is zero)

$\Rightarrow$ $e' \in P_v$, $\omega(e')$ $(= s') \notin Q_\phi$

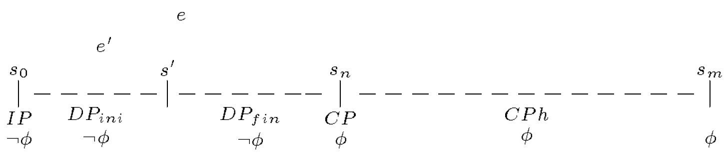
Dynamic Nucleus Structure of Activity-verbs ('push')



$\Rightarrow$ ϕ expresses the w^*-minimal result (the mass of the object denoted by the internal argument is zero)

$\Rightarrow$ $e' \in P_v$, $\omega(e')$ $(= s') \in Q_\phi$

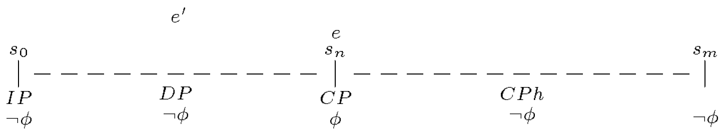
Dynamic Nucleus Structure of Transfer-verbs ('give')



$\Rightarrow$ ϕ expresses the maximal result (the object denoted by the oblique argument has the object denoted by the internal argument)

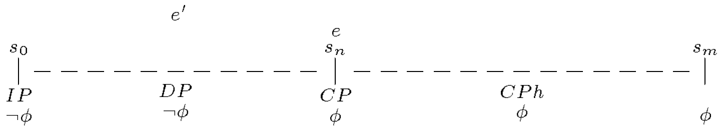
$\Rightarrow$ $e' \notin P_v$, $\omega(e')$ $(= s') \notin Q_\phi$

Dynamic Structure of Point-verbs ('knock')



$\Rightarrow e \in P_v$; $e' =$ presupposed event

Dynamic Structure of Achievement-verbs ('reach')



$\Rightarrow e \in P_v$; $e' =$ presupposed event

3.2 Dynamic Modal Arrow Logic

In this section a logic L, Dynamic Modal Arrow Logic (DMAL), is defined in which it is possible to express the characterizing properties of aspectual classes from the last section. L combines elements from Dynamic Modal Logic (DML), de Rijke (1993), and Arrow Logic (AL), van Benthem (1996). It is two-sorted: there are both s-formulas that are evaluated at elements from S and e-procedures that are evaluated relative to elements from the domain E of events. s-formulas are used to make assertions about what is true or false at particular points of the execution sequence $\tau(e)$ of an event. They are therefore related to the level of changes as transformation of states; e-procedures, on the other hand, are related to the level of changes as objects. They admit to make assertions about properties of events. The link between e-procedures and s-formulas is established by basic e-procedures of the form $R\phi$, $L\phi$ and $D\phi$, which make assertions about the beginning-point $\alpha(e)$, the end-point $\omega(e)$ and an intermediate point of the exeecution sequence of an event e, respectively. The interpretation of s-formulas is standard. The interpretation of e-formulas is given in (20).

(20)
a. $M, e \models \sqsubseteq$ iff $e \in E$
b. $M, e \models \sqsubseteq_v$ iff $e \in P_v$
c. $M, e \models \pi \cap \pi'$ iff $M, e \models \pi \wedge M, e \models \pi'$
d. $M, e \models \pi \cup \pi'$ iff $M, e \models \pi \vee M, e \models \pi'$
e. $M, e \models \pi \bullet \pi'$ iff $\exists e_1, e_2[Ce, e_1 e_2 \wedge M, e_1 \models \pi \wedge M, e_2 \models \pi']$
f. $M, e \models \sim \pi$ iff *not* $M, e \models \pi$
g. $M, e \models \delta$ iff $\alpha(e) = \omega(e)$
h. $M, e \models R\phi$ iff $M, \omega(e) \models \phi]$
i. $M, e \models L\phi$ iff $M, \alpha(e) \models \phi]$
j. $M, e \models D\phi$ iff $\exists s[DP(e, s) \wedge M, s \models \phi]$
$DP = \{(e, s) \mid s \in \tau(e) - \{\alpha(e), \omega(e)\}\}$
k. $M, e \models [X]\pi$ iff $\forall e_1[Xe_1, e \rightarrow M, e_1 \models \pi]$ $\quad X \in \{B, B^*, E'\}$

Semantically, the function of $\sqsubseteq_v$ is similar to that of the program letters π in DL. They admit to impose a sortal restriction on an event e. In DMAL dynamic modes are interpreted as operations from $\wp(E) \times \wp(S)$ to $\wp(E)$. As said in the previous section, this change of interpretation relative to DML reflects, first, that transitions qua events are basic objects in DMAL and, second, that a transformation of state is relativized to an event-type P_v. The e-procedures of the form $DM(\pi, \phi)$ used in the translation of the aspectual classes are defined in (21).

(21)
a. $Min-BEC_v(\sqsubseteq_v, \phi) =_{def.} L\neg\phi \cap \sqsubseteq_v \cap R\phi \cap \sim [(\sqsubseteq_v \cap R\phi) \bullet (\sqsubseteq_v)]$
b. $Min-BEC_{<_S}(\sqsubseteq_v, \phi) =_{def.} L\neg\phi \cap \sqsubseteq_v \cap R\phi \cap \sim D\phi$
c. $Min-BEC^*_{<_S}(\sqsubseteq_v, \phi) =_{def.} Min-BEC_{<_S}(\sqsubseteq_v, \phi) \cap \sim [\sqsubseteq_v \bullet \sqsubseteq_v]$
d. $Con-BEC_v(\sqsubseteq_v, \phi) =_{def.} L\neg\phi \cap \sqsubseteq_v \cap R\phi \cap \sim [(\sqsubseteq_v \cap R\neg\phi) \bullet (\sqsubseteq_v)]$
e. $Con-BEC_{<_S}(\sqsubseteq_v, \phi) =_{def.} L\neg\phi \cap \sqsubseteq_v \cap R\phi \cap \sim D\neg\phi$
f. $BEC^*(\sqsubseteq_v, \phi) =_{def.} \sqsubseteq_v \cap R\phi$
g. $HOLD_{<_S}(\sqsubseteq_v, \phi) =_{def.} L\phi \cap \sqsubseteq_v \cap R\phi \cap \sim D\neg\phi$
h. $HOLD^1(\sqsubseteq_v, \phi) =_{def.} HOLD_{<_S}(\sqsubseteq_v, \phi) \cap \delta$
i. $HOLD^1(\sqsubseteq_v, \phi) =_{def.} HOLD_{<_S}(\sqsubseteq_v, \phi) \cap \sim \delta$
j. $END(\sqsubseteq_v, \phi) =_{def.} HOLD^1(\sqsubseteq_v, \phi) \cap \langle E' \rangle Min-BEC_{<_S}(\sqsubseteq, \phi) \cap$ $< B > HOLD(\sqsubseteq, \phi)$
k. $OCCUR(\sqsubseteq_v, \phi) =_{def.} HOLD^1(\sqsubseteq_v, \phi) \cap \langle E' \rangle Min-BEC_{<_S}(\sqsubseteq, \phi)$

Analyzing Accomplishment- and Activity verbs in the way suggested above solves two problems that event semantics faces. First, the $Con-BEC_v$ mode can be taken to express a weakened form of the property of divisivity which requires an event-predicate to be true of all subevents e' of an event e that satisfies the predicate. As a partial characterization of Activity-expressions this property is too strong as the example of waltzing shows. As it takes three steps to waltz, there are initial-stages e' of an event of waltzing that are not of this type. The $Con-BEC_v$ mode, on the other hand, requires the condition expressed by its second argument to only hold at those subevents e' that are of type P_v, thereby restricting the relevant set of subevents to those satisfying a particular condition (see Naumann 1999a for details). Second, the $Min-BEC$ mode expresses a weakened form of the property of being quantized that requires an event predicate to be false of each subevent e' of an event e satisfying the predicate. This property cannot characterize Accomplishment-expressions as the example of 'walk to the station' shows. Each (non-minimal) final stage e' of an event e of this type belongs to this type too such that the predicate is not quantized. The $Min-BEC_v$ mode only requires that its second argument ϕ is false at the end-point of all proper initial-stages of e. In the case of Accomplishment-verbs (as well as Transfer-verbs) even something stronger holds: ϕ is false for all states of e's execution-sequence, except at $\omega(e)$. The corresponding dynamic mode is $Min-BEC_{<_S}$, (21b). The translation of a verb v (or a sentence containing that verb) is an e-procedure of the form $DM(\sqsubseteq_v, \phi)$ such that DM is the dynamic mode defining the aspectual class to which v belongs. The second argument ϕ is not uniquely determined because it depends on an object that undergoes a change (e.g. an apple that is eaten) such that the translation is a translation-scheme. For different objects one gets different ϕ. In (22a) the general scheme and in (22b,c) two examples are given.

(22) a. $v \rightsquigarrow DM(\sqsubseteq_v, \phi)$

b. eat $\rightsquigarrow$ $Min - BEC_{<_S}(\sqsubseteq_{eat}, \phi')$

c. push $\rightsquigarrow$ $Con - BEC_v(\sqsubseteq_{push}, \psi)$

3.3 The Interpretation of Modifying Expressions like directional PPs

In section (2) it was shown (i) that each event-type P_v imposes a minimal requirement on the result that must be brought about and (ii) that this requirement can be strengthened by the interpretation of v in the lexicon. E.g., whereas P_{eat} requires only that the w^*-minimal result be brought about, the interpretation of 'eat' imposes the stronger condition that the s-maximal result be realized. The semantic function of modifying expressions like directional PPs and secondary predications is analyzed in an analogous way. They impose a further condition that an event e must satisfy. For instance, the PP 'to the station' requires the object with respect to which e brings about the change to be at (or in) the station at the end-point of e. Expressed in terms of the path that is traversed, the condition amounts to the requirement that the end of the path is at (or inside) the station. Consequently, both the verb, e.g. 'run', and the PP impose a condition on the path that is traversed by the object undergoing the change. The condition imposed by the former is a strengthening of that imposed by the latter such that the change of aspectual properties is monotone. Furthermore, the condition that the object undergoing the change be at the station at the end-point of the event e must be satisfied only at that point and at no other point of the execution sequence of e such that it is s-maximal. This second requirement imposes a condition on the *way* the result determined by the PP must be brought about. Thus, a directional PP imposes both a result and a condition on how this result is brought about. It can therefore be translated as an e-procedure of the form $DM(\pi, \phi)$. The e-procedure π will express some accessibility-relation. As a directional PP like 'to the station' can be combined with expressions belonging to different event-types, π must be $\sqsubseteq$, which expresses the most general accessibility-relation (corresponding to some special element from $VERB$). The translation of 'to the station' is (23).

(23) to the station $\rightsquigarrow$ $Min - BEC_{<_S}(\sqsubseteq, \psi_{at\_the\_station})$

(23) is a translation-scheme because the second argument of $Min - BEC_{<_S}$ depends on a particular object that undergoes a change such that for different objects one gets different ψ. The translation of 'run to the station' is (24).

(24) run to the station $\rightsquigarrow$ $Con{-}BEC_v(\sqsubseteq_{run}, \phi) \cap Min{-}BEC_{<_S}(\sqsubseteq, \psi_{at\_the\_station})$

(24) is an instance of (25).

(25) $DM(\pi, \phi) \cap DM'(\pi', \phi')$

(25) represents a general scheme of how changes (or, more generally, additions) of aspectual properties, triggered e.g. by modifying expressions, are accounted for in DMAL. The properties the execution sequence $\tau(e)$ of an event e satisfying $DM(\pi, \phi)$ has with respect to the property expressed by ϕ will in

general be different from those $\tau(e)$ has relative to the property expressed by ϕ and ϕ'. These properties can be calculated from implications like (26).[4]

(26) $Min - BEC_{<_S}(\pi, \phi) \cap Con - BEC_v(\pi, \psi) \rightarrow Min - BEC_{<_S}(\pi, \phi \wedge \psi)$

3.4 The Interpretation of 'Until'

'Until' requires an initial-stage e_1 of the event e denoted by the MC to go on until a particular point of the event e' denoted by the SC. If the procedure π' in the SC is of the form $Min - BEC_{<_S}(\sqsubseteq_v, \phi)$, i.e., if the event e is strongly non-closed under initial-stages with respect to this procedure, the SC determines two points: either the beginning-point $\alpha(e')$ of e' or its end-point $\omega(e')$ as possible values for $\omega(e_1)$. If π' is of the form $Con - BEC_v(\sqsubseteq_v, \phi))$ or if the procedure requires e' to be an instantaneous event, it is required that $\omega(e_1) = \alpha(e')$. For instantaneous events e', this is equivalent to $\omega(e_1) = \alpha(e')$ because $\alpha(e') = \omega(e')$. This difference with respect to the number of possible end-points for e_1 is a consequence of a difference at the level of the dynamic structure. For an event-type that is characterized by the $Min - BEC_{<_S}$ mode it is possible that after the execution of a finite number of events of this type the result determined in the lexicon does not hold. For events belonging to an event-type that is characterized by another mode this is not possible (see Naumann (1998a,b), Naumann/Osswald (1999) for details). The aspectual restriction follows from the fact that the run-time $\tau(e_1)$ of e_1 can be arbitrarily restricted by the event e' denoted by the SC. What must be guaranteed is that for any choice of $\alpha(e')$ or $\omega(e')$ e_1 satisfies π, i.e. the procedure in the MC. This only holds if the interpretation of π ($= DM(\sqsubseteq_v, \phi)$) is closed under initial-stages modulo P_v and events satisfying π are not required to be instantaneous, i.e., $\alpha(e) = \omega(e)$ must not hold. The interpretation of 'Until' is given in (27).

(27) $M, e \models U(\pi, \pi')$ iff there are e_1, e' s.t.

- (i) $Be_1, e \wedge M, e_1 \models \pi$
- (ii) if $\forall e[M, e \models \pi \Leftrightarrow M, e \models \pi \cap [B^*] \sim \pi]$, then $\omega(e_1) = \omega(e') \vee \omega(e_1) = a(e')$ else $\omega(e_1) = \alpha(e')$
- (iii) $M, e' \models \pi'$

3.5 The Hybrid Language L^h

The analysis developed so far faces the following two, related problems. First, in $DM(\pi, \phi)$ ϕ can be an arbitrary s-formula. It need not be a result that e can bring about, in particular it need not be a maximal element of $Res(P_v, e)$ with respect to $\leq_v$. Second, ϕ is brought about with respect to a particular participant d of e. This aspect is not captured either. This leads to problems in the case of modification. E.g., for an event of pushing a cart to the station, it is the cart which must be at the station at the end of the event and not some arbitrary object. These problems can be solved as follows. Recall that each basic event type P_v determines for each of its elements e a set $Res(e, P_v)$ of results that e can possibly bring about. Each element of $Res(e, P_v)$ is brought

[4] An interpretation of secondary predication in a slightly different framework is presented in Naumann (1999c).

about with respect to at least one object that participates in e. E.g., if e is of type 'John eat a fish', the results are brought about with respect to John and the fish. John is assigned both the s-minimal result (e.g. his mouth is open) and a w^*-minimal one (part of the fish is in his stomach) whereas the fish is assigned a w^*-minimal result (its mass partly decreased) and the s-maximal one (its mass is zero). In the case of an event e of type 'Bill push the cart' Bill is assigned the s-minimal result (his actions towards the cart) and possibly a w^*-minimal one (Bill traverses a non-empty path) whereas the cart is assigned only a w^*-minimal result (the cart traverses a non-empty path). The relationship between an event $e \in P_v$, an object d participating in e and a result $Q \in Res(P_v, e)$ that e can possibly bring about with respect to d is captured by a relation Δ_v on $E \times O \times \wp(S)$; see Latrouite/Naumann 1999b for details.

The result that is determined by the interpretation of verbs as well as of modifying expressions is brought about with respect to a particular participant d of e that can be defined in terms of the results that are assigned to it, together with a temporal maximality condition: (i) d is assigned all results that are maximal with respect to $\leq_v$ and (ii) d is that participant of e that is involved last with respect to the objects that satisfy condition (i), for details see Latrouite/Naumann (1999b). This relationship is captured by a (functional) relation Ω_v: $\Omega_v(e) = d$ just in case d is the object satisfying (i) and (ii). In terms of the Δ_v the sets $\Delta^*_v(e)(d) = \{Q \mid \Delta_v(e)(d)(Q) \wedge \forall Q'[\Delta_v(e)(d)(Q') \rightarrow Q' \leq_v Q]\}$ and $\Delta^*(e)(d) = \cup_{v \in VERB} \Delta^*_v(e)(d)$ are defined.

At the formal level L is extended in two ways to a hybrid language L^h. First, for each $v \in VERB$ two operators Σ and Σ^* are added to L that bound variables which take their values in $E \cup O$. In L^h e-procedures are evaluated at elements from E relative to a variable assignment g. Second, besides e-procedures and s-formulas there are e,d-formulas that are evaluated at elements from $E \times O$. At the syntactic level, the following clauses are added: if ϕ is an atomic s-formula, $\Sigma_{v_x}\phi$ and $\Sigma^*_{v_x}\phi$ are e-procedures. Each s-formulas is an e,d-formula (and nothing else is an e,d-formula). The relevant semantic clauses are given in (28) (note that the clauses in (20) above must be made dependent on a variable assignment too).

(28) a. $M, e \models_g \Sigma^*_{v_x}\phi$ iff $V(\phi) \in \Delta_v(e)(g(x))$

b. $M, e \models_g \Sigma_{v_x}\phi$ iff $\Omega_v(e)(g(x))$ and $M, e, g(x) \models \phi$

c. $M, e, d \models p$ iff $V(p) \in \Delta^*(e)(d)$ for p an atomic e,d-formula

d. $M, e, d \models \neg\phi$ iff not $M, e, d \models \phi$

e. $M, e, d \models \phi \wedge \psi$ iff $M, e, d \models \phi$ and $M, e, d \models \psi$

$\Sigma_{v_x}\phi$ is true relative to some $e \in E$ (and relative to g) only if (i) $e \in P_v$, (ii) $g(x)$ is the object that is assigned all results that are maximal with respect to $Res(P_v, e)$ relative to $\leq_v$and (iii) Q_ϕ is a maximal element of $Res(P_{v'}, e)$ (relative to $\leq_{v'}$) for some $P_{v'}$ to which e belongs.

The intuition behind letting each s-formula be an e,d-formula and vice versa is the following: s-formulas are used to make assertions about what holds at particular points of the execution sequence $\tau(e)$ of an event e (or, more generally, about the corresponding Dynamic Nucleus-Structure). Yet one is only interested in the truth or falsity of these s-formulas that express properties which are directly related to the particular event e and not on those that express properties which happen to hold or fail to hold during the execution of e due to the execution of other events. This second aspect, which is not accounted for by s-formulas, is captured by the e, d-formula that corresponds to a given s-formula.

The dependency of the second argument of $DM(\sqsubseteq_v, \phi)$ on an object $d \in O$ in the translation of the verb v can be made explicit by adding the clause $\Sigma^*_{v_x}\phi$, yielding (29a). Like (22a), (29a) is a translation-scheme. In contrast to (22a), (29a) is false if ϕ expresses a property that is evaluated according to DM on $\tau(e)$ but which is not brought about by e (with respect to $g(x)$). In (29b) the translation of 'eat' is given. Similarly, to the translation of a modifying expression the clause $\Sigma_{v_x}\psi$ is added. This yields (29c). In (29d) the translation of 'to the station' is given and in (29e) that of 'run to the station'.

(29)
a. $v \leadsto DM(\sqsubseteq_v, \phi) \cap \Sigma^*_{v_x}\phi$
b. eat $\leadsto Min - BEC_{<_S}(\sqsubseteq_{eat}, \phi) \cap \Sigma^*_{eat_x}\phi$
c. $MOD(EXP_\psi) \leadsto TR(EXP_\psi) \cap Min - BEC_{<_S}(\sqsubseteq, \psi) \cap \Sigma_{v_c}\psi$
d. to the station $\leadsto Min - BEC_{<_S}(\sqsubseteq, \psi_{at\_the\_station}) \cap \Sigma_{v_c}\psi_{at\_the\_station}$
e. run to the station $\leadsto$ $Con - BEC_v(\sqsubseteq_{run}, \phi) \cap \Sigma^*_{run_x}\phi \cap$ $Min - BEC_{<_S}(\sqsubseteq, \psi_{at\_the\_station}) \cap \Sigma_{v_y}\psi_{at\_the\_station}$

References

[1] van Benthem, J. (1996). *Exploring Logical Dynamics*, Stanford, CA, CSLI Publications

[2] Latrouite, A./Naumann, R. (1999a). An Interpretation of Tagalog Voice-Affixes in a Dynamic Event Semantics, to appear in *Proceedings of the 6th Annual Meeting of the Austronesian Formal Linguistics Association, Toronto*

[3] Latrouite, A./Naumann, R. (1999b). An Interpretation of Nominative Arguments, Voice-Affixes and Verb Stems in Tagalog, this volume

[4] Moens, M./Steedman, M: (1988). Temporal Ontology and Temporal Reference, *Computational Linguistics*, vol. 14, pp.15-28

[5] Naumann, R. (1998a). A Hybrid Language for the Analysis of Aspectual and Temporal Phenomena in Natural Language, *Proc. FroCoS'98*, University of Amsterdam, to appear with Kluwer

[6] Naumann, R. (1998b). A Dynamic Logic of Events and States for the Analysis of the Interaction between Verb Aspect and Plural Quantification in Natural Language, *Proc. 5th WoLLIC*, Sao Paulo, Brazil, extended version to appear in *Journal of the IGPL*

[7] Naumann, R. (1999a). A Dynamic Approach to Aspect: Verbs as Programs, to appear in *Journal of Semantics*

[8] Naumann, R. (1999b). A Dynamic Approach to the Present Perfect in English, *Theoretical Linguistics*

[9] Naumann, R. (1999c). A Dynamic Modal Logic for the Interpretation of Three Types of Modification in Natural Language, *Proc. TALN 99*

[10] Naumann, R./Mori, Y. (1998). An Analysis of Definite and Indefinite Japanese Aspectualities in a Dynamic Decompositional Event Semantics, *Proc. 5th Symposium on Logic and Language*, Budapest

[11] Naumann, R./Osswald, R. (1999). Aspects of 'Until', *Proceedings IWCS*, Tilburg

[12] de Rijke, M. (1993). *Extending Modal Logic*, PhD. Thesis, University of Amsterdam

Request PDF on ResearchGate | A Dynamic Temporal Logic for Aspectual Phenomena in Natural Language | It is usually thought that Dowtyâ€™s decompositional approach to aspect and that of event-semantics are incompatible with each other. In this paper this is shown to be wrong by presenting a theory that combines both approaches.Â The Aspectual Analysis of English Verbs in a Dynamic Decom positional Event Semantics, manuscript. R Naumann. A Theory of Aspectuality.Â When beholding the natural world from which the language of the elements emerges, one [Show full abstract] cannot help but admire the inexhaustible wealth of forms and modes of being therein, which could be interpreted simply as other means of expression that are based on the elements.