Progression and Iteration in Event Semantics - An LTAG Analysis Using Hybrid Logic and Frame Semantics

Laura Kallmeyer, Rainer Osswald, Sylvain Pogodalla

Progression and Iteration in Event Semantics – An LTAG Analysis Using Hybrid Logic and Frame Semantics*

Laura Kallmeyer and **Rainer Osswald**
Heinrich Heine Universität, Düsseldorf, Germany

Sylvain Pogodalla
Heinrich Heine Universität, Düsseldorf, Germany
INRIA, Villers-lès-Nancy, F-54600, France
Université de Lorraine, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France
CNRS, LORIA, UMR 7503, Vandœuvre-lès-Nancy, F-54500, France

Abstract

In this paper, we propose to use Hybrid Logic (HL) as a means to combine frame-based lexical semantics with quantification. We integrate this into an LTAG syntax-semantics interface and show that this architecture allows a fine-grained description of event structures by quantifying for instance over subevents. As a case study we provide an analysis of iteration and progression in combination with *for*-adverbials. With the HL approach and with standard techniques of underspecification we can account for the behaviour of these adverbials without the assumption of an additional iteration operator on events. This is due to the fact that frame semantics allows to express general constraints on event types that require an event to be an iteration in certain contexts.

1 Introduction

An important topic for theories of aspectual composition and coercion is the interaction of lexical aspect (*Aktionsart*) and temporal adverbials. On the one hand, *in*- and *for*-adverbials have been in use since Vendler (1957) as indicators for distinguishing between activities and accomplishments, among others. On the other hand, there are many types of sentences in which a temporal adverbial is not compatible with the lexical aspect of the verb but which have nevertheless a regular interpretation; cf., e.g., Egg (2005). For example, while in (1-a), the verb *cry* denotes an activity and is thus immediately compatible with the *for*-adverbial, the verb *cough* in (1-b) is semelfactive, i.e., denotes a punctual event, and, hence, calls for additional adjustments in order to be compatible with *for*-adverbials.

(1) a. Peter cried for ten minutes.
    b. Peter coughed for ten minutes.

In the present case, the necessary adjustment consists is interpreting (1-b) as describing a sequence or iteration of coughings. In a recent paper, Champollion (2013) proposes an account of *for*-adverbials that explains the coercion phenomena in cases like (1-b), the associated processing costs, and the scopal behavior of the adverbials in the transitive case. The present paper aims at extending and revising this approach towards a model of the syntax-semantics interface which combines a model for syntactic composition (*Lexicalized Lexicalized Tree Adjoining Grammar*) with underspecified descriptions (using *Hybrid Logic*) of *decompositional semantic frames*.

Frames emerged as a representation format of conceptual and lexical knowledge (Fillmore, 1982; Barsalou, 1992; Löbner, 2014). They are commonly presented as semantic graphs with labelled nodes and edges, as in Fig. 1, where nodes correspond to entities (individuals, events, …) and edges to (functional or non-functional) relations between these entities. In Fig. 1 all relations except *part-of* are

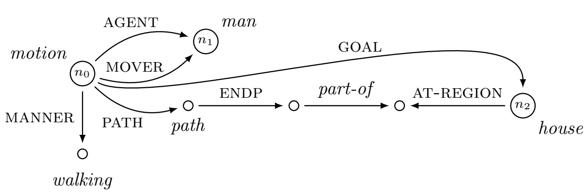

Figure 1: Frame for the meaning of *the man walked to the house* (adapted from Kallmeyer and Osswald (2013))

* This work was supported by the INRIA sabbatical program and by the CRC 991 "The Structure of Representations in Language, Cognition, and Science" funded by the German Research Foundation (DFG).

meant to be functional. This representation offers a fine-grained decomposition of meaning and should not be confused with the FrameNet frames, although the former can help to capture the structural relations of the latter as shown by Osswald and Van Valin Jr. (2014). Frames can be formalized as extended typed feature structures (Petersen, 2007; Kallmeyer and Osswald, 2013) and specified as models of a suitable logical language. Such a language allows for the composition of lexical frames on the sentential level by means of an explicit syntax-semantics interface (Kallmeyer and Osswald, 2013). Yet, this logical framework does not provide means for the lexical items to introduce explicit quantification over entities or events. To this end, we use Hybrid Logic (HL) (Areces and ten Cate, 2007). HL is an extension of modal logic. As such, it is well-suited to the description of graph structures, such as the one of Fig. 1. We use in particular two features of HL. First, we use *nominals*, that allow the logical formulas to refer to specific nodes of the graph. It is then possible, for example, to specify that the AGENT and the MOVER edges from the node n_0 should meet on the n_1 node in Fig. 1. Second, we use *variables* for nodes, and the associated *quantifiers*, that can appear in the logical formulas. Using these language features, we provide lexical items with a logical meaning describing a property that should hold for any node of the frame.

2 The Formal Framework

2.1 Hybrid Logic and Semantic Frames

We slightly adapt the notation of hybrid logical formulas given in Areces and ten Cate (2007) to our purposes. Let $\mathsf{Rel} = \mathsf{Func} \cup \mathsf{PropRel}$ be a set of functional and non-functional relation symbols, Type a set of type symbols, Nom of nominals (node names), and Nvar a set of node variables, with $\mathsf{Node} = \mathsf{Nom} \cup \mathsf{Nvar}$. Formulas are defined as:

(2) $$\mathsf{Forms} ::= \top \mid p \mid n \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \langle R \rangle \phi \mid \mathbf{E}\phi \mid @_n\phi \mid \downarrow x.\phi \mid \exists x.\phi$$

where $p \in \mathsf{Type}$, $n \in \mathsf{Node}$, $x \in \mathsf{Nvar}$, $R \in \mathsf{Rel}$ and $\phi, \phi_1, \phi_2 \in \mathsf{Forms}$. Moreover, $\mathbf{A}\phi \equiv \neg\mathbf{E}\neg\phi$, $\phi \rightarrow \psi \equiv \neg\phi \vee \psi$, etc. The elements of Func will be written in small caps.

Due to lack of space, we keep the definition of satisfaction and truth of a formula in a model under an assignment at an informal level; cf. Areces and ten Cate (2007, pp. 825, 831) for a formal account. A formula $\langle\text{AGENT}\rangle$*man* is true at any node from which there exists an edge AGENT reaching a node of which the type *man* is true. For instance, $\langle\text{AGENT}\rangle$*man* is true at the node named by n_0, and false at the node named by n_1 of Fig. 1. Let ϕ_0 be the formula $\exists x.\langle\text{AGENT}\rangle x \wedge \langle\text{MOVER}\rangle x$. Then ϕ_0 is true at a node v only if there is a node w (to which the variable x is assigned) which is reached from node v by the (functional) edges AGENT and MOVER. So for ϕ_0 to be true at v, the two edges AGENT and MOVER starting at v should reach the same node. Hence ϕ_0 is true at n_0, but nowhere else in Fig. 1. Formulas of HL can also specify at which node a formula should be evaluated: $@_n\phi$ is true at any node if ϕ is true at the node to which n is assigned. More generally, $\mathbf{E}\phi$ is true if there is a node v such that ϕ is true at v. Hence, $\mathbf{E}\phi_0$ is true in Fig. 1. By employing this language, we can characterize the frame of Fig. 1 by the following formula:

(3) $$@_{n_0}\mathit{motion} \wedge \langle\text{AGENT}\rangle(n_1 \wedge \mathit{man}) \wedge \langle\text{MOVER}\rangle n_1 \wedge \langle\text{GOAL}\rangle(n_2 \wedge \mathit{house}) \wedge \langle\text{MANNER}\rangle\mathit{walking} \wedge (\exists x\, y.\langle\text{PATH}\rangle(\mathit{path} \wedge \langle\text{ENDP}\rangle x) \wedge @_{n_2}(\langle\text{AT-REGION}\rangle y) \wedge @_x(\langle \mathit{part\text{-}of} \rangle y))$$

2.2 LTAG and Hybrid Logic

A *Lexicalized Tree Adjoining Grammar* (LTAG; Joshi and Schabes (1997), Abeillé and Rambow (2000)) consists of a finite set of *elementary trees*. Larger trees can be derived via the composition operations *substitution* (replacing a leaf with a tree) and *adjunction* (replacing an internal node with a tree). An adjoining tree has a unique non-terminal leaf that is its *foot node* (marked with an asterisk). When adjoining such a tree to some node n, in the resulting tree, the subtree with root n from the original tree ends up below the foot node.

In order to capture syntactic generalizations, the non-terminal node labels are enriched with feature structures (Vijay-Shanker and Joshi, 1988). Each node has a top and a bottom feature structure (except substitution nodes, which have only a top). Nodes in the same elementary tree can share features. Substitutions and adjunctions trigger unifications: In a substitution step, the top of the root of the new tree unifies with the top of the substitution node. In an adjunction step, the top of the root of the adjoining tree unifies with the top of the adjunction

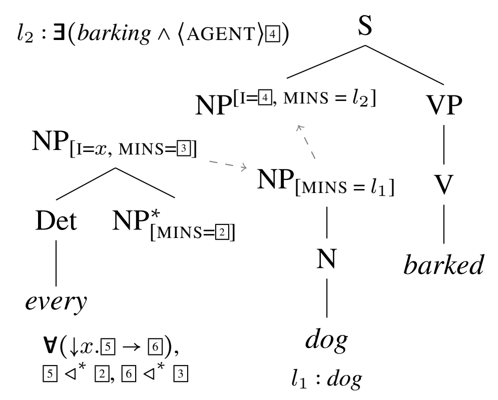

Figure 2: Derivation of *every dog barked*

site and the bottom of the foot of the adjoining tree unifies with the bottom of the adjunction site. Furthermore, in the final derived tree, top and bottom must unify in all nodes.

Our architecture for the interface between TAG syntax and frame semantics builds on previous approaches which pair each elementary tree with a semantic representation that consists of a set of formulas, in this case HL formulas, which can contain holes and which can be labeled. I.e., we apply *hole semantics* (Bos, 1995) to HL and link these underspecified formulas to the elementary trees. Composition is then triggered by the syntactic unifications arising from substitution and adjunction, using interface features on the syntactic trees (Gardent and Kallmeyer, 2003; Kallmeyer and Joshi, 2003; Kallmeyer and Romero, 2008). As a basic example consider the derivation given in Fig. 2. The *every* tree adjoins to the root of the *dog* tree and the derived tree substitutes into the subject slot of the *barked* tree. (The feature MINS provided by the verbal tree determines the minimal scope of any attaching quantifier.) The syntactic unifications lead to $\boxed{4} = x$, $\boxed{2} = l_1$, $\boxed{3} = l_2$. As a result of these equations, we obtain the follwing underspecified representation:

(4) $\forall(\downarrow x.\boxed{5} \rightarrow \boxed{6})$,
$l_1 : dog$, $l_2 : \exists(barking \wedge \langle \text{AGENT} \rangle x)$,
$\boxed{5} \lhd^* l_1, \boxed{6} \lhd^* l_2$

The respresentation in (4) has a unique solution given by the mapping $\boxed{5} \mapsto l_1, \boxed{6} \mapsto l_2$, which leads to (5).

(5) $\forall(\downarrow x. dog \rightarrow \exists(barking \wedge \langle \text{AGENT} \rangle x))$

3 Application to *for*-Adverbials

3.1 *For*-Adverbials and Atelic Events

We start with a basic case of a *for*-adverbial modifying an atelic event description:

(6) Peter swam for one hour

First we have to specify the meaning of *swimming*. Let us simplify here and take it to be represented by a frame described by $swimming \wedge \langle \text{AGENT} \rangle \boxed{2}$. However, we must capture the fact that a swimming event can be conceived as a progression of smaller swimming events, for instance when combining with a *for*-adverbial. In this case, some part of the meaning specification holds also for these subevents while some other aspects are relevant only for the entire event. We want to separate these two parts of the semantics of *swim*. In the lexical entry in Fig. 3, those aspects that also hold for subevents are put into a formula labeled l_3 while the more general formula describing the entire event but not necessarily its subevents is labeled l_2. In our simplified example this is just a hole. In general, we might for instance have a specification of the PATH or TRACE of the entire event here. This more general formula has a hole and there is a constraint ($\boxed{4} \lhd^* l_3$) saying that the specification of the properties that hold for all subevents (if there are any) is contained in this hole. If some modifier details the event structure by inspecting subevents, this modifier is inserted between $\boxed{4}$ and l_3.

Our semantics of *for*-adverbials follows to a large extent Champollion (2013). Champollion's semantics for *for an hour* is given in (7):

(7) $\lambda P \lambda I[AT(P, I) \wedge hours(I) = 1$
$\wedge \forall J[J \in \mathcal{R}_I^{short(I)} \rightarrow AT(P, J)]]$

Here, I is the time interval of the entire event while $\mathcal{R}_I^{short(I)}$ is supposed to be a contextually determined partiton of I into sufficiently short subintervals. The *for*-adverbial applies to some predicate P expressing those properties of the event that also hold for its subevents if conceived as a progression. P then holds for the entire time interval I and also for all subintervals under the given partition.

In our analysis, we do not make the time interval of an event explicit (though, if needed, this can be added). Instead, we talk about a contextually relevant set of subevents that can overlap and that cover the entire event.

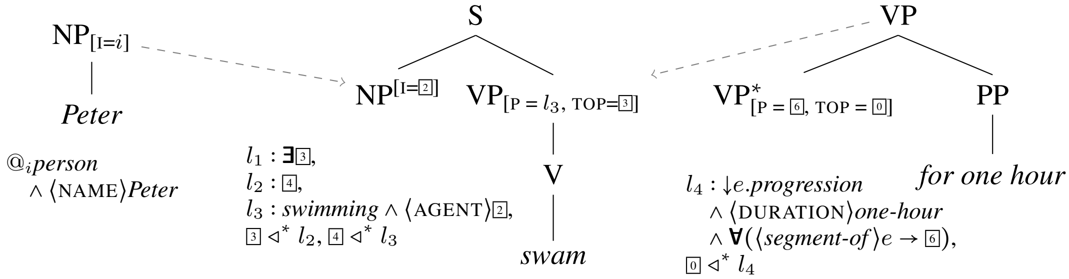

Figure 3: Derivation for (6)

Such a set of subevents is assumed by the *for*-adverbial (see the variable e in Fig. 3). The semantics of *for one hour* states that all segments of e, i.e., all relevant subevents, have to satisfy $\boxed{6}$ where $\boxed{6}$ will be filled by the formulas stating those aspects of the modified event that apply also to its subevents. In our case, we obtain $\boxed{6} = l_3$.

A crucial difference to (7) is that we do not require the entire event to satisfy the formula inserted in $\boxed{6}$, i.e., we do not assume $P(I)$. If e is a progression, we assume that this follows from a general constraint stating that a *progression* is a segment of itself:

(8) $\forall(\downarrow e.\textit{progression} \rightarrow \langle \textit{segment-of} \rangle e)$

We will see in the next section that this is different for iterations.

The substitution and adjunction in Fig. 3 trigger the unifications $\boxed{0} = \boxed{3}, \boxed{2} = i, \boxed{6} = l_3$ on the interface features. As a result, when applying these and collecting the formulas, we obtain the following underspecified semantic formula:

(9)
$$\begin{array}{l} @_i\textit{person} \wedge \langle \text{NAME} \rangle \textit{Peter}, \\ l_1 : \exists\boxed{3}, l_2 : \boxed{4}, \\ l_4 : \downarrow e.\textit{progression} \\ \quad \wedge \langle \text{DURATION} \rangle \textit{one-hour} \\ \quad \wedge \forall(\langle \textit{segment-of} \rangle e \rightarrow l_3), \\ l_3 : \textit{swimming} \wedge \langle \text{AGENT} \rangle i, \\ \boxed{3} \lhd^* l_4, \boxed{3} \lhd^* l_2, \boxed{4} \lhd^* l_3 \end{array}$$

With $\boxed{4} \lhd^* l_3$, we obtain $\boxed{4} \lhd^* l_4$ since the syntactic structures of our formulas have to be tree-shaped. Then the only possible disambiguation mapping is $\boxed{3} \mapsto l_2, \boxed{4} \mapsto l_4$, which yields, with an additional conjunctive interpretation of the set, (10):

(10)
$$\begin{array}{l} @_i\textit{person} \wedge \langle \text{NAME} \rangle \textit{Peter} \\ \wedge \exists{\downarrow}e.(\textit{progression} \wedge \langle \text{DURATION} \rangle \textit{one-hour} \\ \qquad \wedge \forall(\langle \textit{segment-of} \rangle e \rightarrow \\ \qquad\qquad \textit{swimming} \wedge \langle \text{AGENT} \rangle i)) \end{array}$$

Furthermore, with (8), *swimming* $\wedge \langle \text{AGENT} \rangle i$ also holds at e.

3.2 Punctual Events and *for*-Adverbials

Now we consider cases where a *for*-adverbial combines with a punctual event. In this case, the event is reinterpreted as an iteration.

(11) Peter knocked at the door for ten minutes

The meaning of (11) is that we have an iteration of knocking events, each of them involving Peter as an agent and the same door as a patient. This entire iteration takes ten minutes.

(12)
$$\begin{array}{l} \exists(\downarrow e.\textit{iteration} \wedge \langle \text{DURATION} \rangle \textit{ten-minutes} \\ \quad \wedge \forall(\langle \textit{segment-of} \rangle e \rightarrow \\ \qquad \textit{knocking} \wedge \langle \text{AGENT} \rangle i \wedge \langle \text{PATIENT} \rangle j)) \\ \wedge @_i(\textit{person} \wedge \langle \text{NAME} \rangle \textit{Peter}) \wedge @_j\textit{door} \end{array}$$

We propose to extend the semantics of *for*-adverbials given in Fig. 3 by adopting a more general type *nq-event* which is a supertype of *progression* and *iteration* and which is intended to capture *non-quantized* event types in the sense of Krifka (1998). Furthermore, the following constraints make sure that every *nq-event* is either an *iteration* or a *progression* and it cannot be both at the same time.

(13)
$$\begin{array}{l} \forall(\textit{nq-event} \leftrightarrow \textit{iteration} \vee \textit{progression}) \\ \forall(\textit{iteration} \rightarrow \neg\textit{progression}) \end{array}$$

The derivation of (11) (Fig. 4) yields (14):

(14)
$$\begin{array}{l} \exists\boxed{3}, \\ l_2 : \textit{knocking} \wedge \langle \text{AGENT} \rangle i \wedge \langle \text{PATIENT} \rangle j, \\ l_4 : \downarrow e.\textit{nq-event} \\ \quad \wedge \langle \text{DURATION} \rangle \textit{ten-minutes} \\ \quad \wedge \forall(\langle \textit{segment-of} \rangle e \rightarrow l_2), \\ @_i(\textit{person} \wedge \langle \text{NAME} \rangle \textit{Peter}), \\ @_j\textit{door}, \\ \boxed{3} \lhd^* l_2, \boxed{3} \lhd^* l_4 \end{array}$$

The only possible mapping is $\boxed{3} \mapsto l_4$, which leads, with a conjunctive interpretation of the resulting set, to (15).

(15)
$$\begin{array}{l} \exists(\downarrow e.\textit{nq-event} \\ \quad \wedge \langle \text{DURATION} \rangle \textit{ten-minutes} \\ \quad \wedge \forall(\langle \textit{segment-of} \rangle e \rightarrow \textit{knocking} \\ \qquad\quad \wedge \langle \text{AGENT} \rangle i \wedge \langle \text{PATIENT} \rangle j)) \\ \wedge @_i(\textit{person} \wedge \langle \text{NAME} \rangle \textit{Peter}) \wedge @_j\textit{door} \end{array}$$

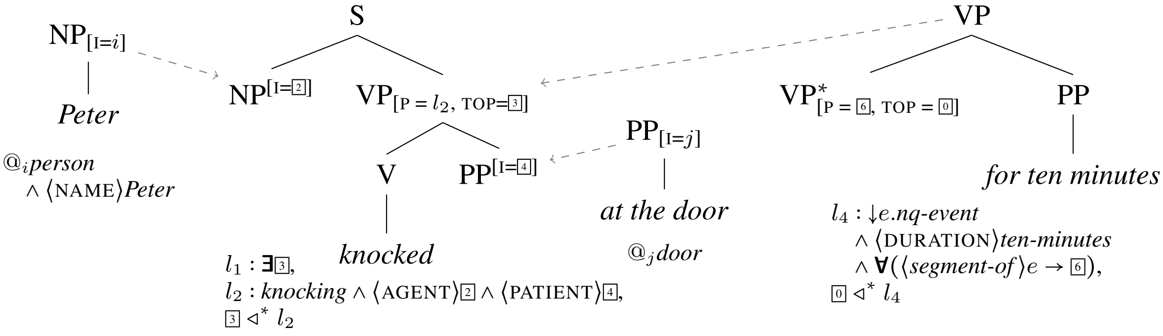

Figure 4: Derivation for (11)

We further adopt additional constraints on iterations and progressions concerning the possible types of their segments:

(16) $\forall(\langle \textit{segment-of}\rangle \textit{iteration} \rightarrow \textit{bounded})$
$\forall(\textit{punctual} \rightarrow \textit{bounded})$
$\forall(\langle \textit{segment-of}\rangle \textit{progression} \rightarrow \neg \textit{bounded})$

With these constraints, e in (15) necessarily has to be of type *iteration* since its segments are knocking events, which are *punctual*. In the case of combining with an atelic event as in (6), we can also use the more general entry for *for*-adverbials. We then obtain that the *nq-event* has to become a *progression* since its segments are not *bounded*.

As already mentioned, this analysis of *for*-adverbials differs from (Champollion, 2013) in that it does not assume that the predicate the adverbial applies to necessarily holds for the entire *iteration/progression* event. In the case of *progression*, this follows from a general constraint on progression. In the case of an *iteration*, the subevents are single bounded events (e.g., *knocking* events) while only the entire event is an iteration of bounded events. In contrast to this, Champollion takes also the subevents to be iterations, i.e., an iteration operator applies first and then the *for*-adverbial is applied to the iteration. According to his analysis, the semantics of (11) is (17) where $^*knock$ is an iteration of knockings:

(17) $$\lambda P \lambda I[\exists e[^*knock(e, Peter, the\ door) \wedge I = \tau(e) \wedge hours(I) = 1 \wedge \forall J[J \in \mathcal{R}_I^{short(I)} \rightarrow \exists e'[^*knock(e', Peter, the\ door) \wedge J = \tau(e')]]]]$$

We do not need this additional iteration operator since our analysis allows to embed the specification of a bounded event under an iteration during composition, for instance with a *for*-adverbial. It seems much more appropriate to conceive an iteration of *knocking* events as a set of single *knocking* events.

4 Conclusion

In this paper, we used HL as a way to include quantification into frame semantics. HL is particularly well-suited for this purpose since it supports the object-centered view that comes with frames and event semantics. We presented a syntax-semantics interface that couples LTAG with HL enriched with holes and labels. This approach allows us to recover analyses of scope and underspecification from previous work. As a test case for the ability to account for interesting cases of event structure, we then considered the combination of telic and atelic events with modifiers that trigger an interpretation as a progression or an iteration, respectively, such as *for*-adverbials. The ingredients of our analysis are that 1) our logic allows quantification over subevents and 2) the types and constraints possible with frames together with underspecification techniques allow to trigger the choice between progression or iteration in a principled way. As a consequence, we have a single entry for *for*-adverbials and we do not require an addition iteration operator. The extended paper will also cover interactions with quantifier scope.

References

Anne Abeillé and Owen Rambow. 2000. Tree Adjoining Grammar: An Overview. In Anne Abeillé and Owen Rambow, editors, *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*, pages 1–68. CSLI.

Carlos Areces and Balder ten Cate. 2007. Hybrid logics. In Blackburn et al. (Blackburn et al., 2007), chapter 14, pages 821–868. DOI: 10.1016/S1570-2464(07)80017-6 .

Lawrence W. Barsalou. 1992. Frames, concepts, and conceptual fields. In Adrienne Lehrer and Eva Feder Kittay, editors, *Frames, Fields, and Contrasts*, New Essays in Semantic and Lexical Organization, chapter 1, pages 21–74. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

Patrick Blackburn, Johan Van Benthem, and Frank Wolter, editors. 2007. *Handbook of Modal Logic*, volume 3 of *Studies in Logic and Practical Reasoning*. Elsevier.

Johan Bos. 1995. Predicate logic unplugged. In Paul Dekker and Martin Stokhof, editors, *Proceedings of the 10th Amsterdam Colloquium*, pages 133–142. URL: http://www.let.rug.nl/bos/pubs/Bos1996AmCo.pdf.

Lucas Champollion. 2013. The scope and processing of *for*-adverbials: A reply to Deo and Piñango. In Todd Snider, editor, *Proceedings of SALT 23*, pages 432–452. URL: http://elanguage.net/journals/salt/article/view/23.432.

Markus Egg. 2005. *Flexible Semantics for Reinterpretation Phenomena*. CSLI Publications, Stanford, CA.

Charles J. Fillmore. 1982. Frame semantics. In The Linguistic Society of Korea, editor, *Linguistics in the Morning Calm*, pages 111–137. Hanshin Publishing Co., Seoul.

Thomas Gamerschlag, Doris Gerland, Rainer Osswald, and Wiebke Petersen, editors. 2014. *Frames and Concept Types*, volume 94 of *Studies in Linguistics and Philosophy*. Springer International Publishing. DOI: 10.1007/978-3-319-01541-5 .

Claire Gardent and Laura Kallmeyer. 2003. Semantic construction in Feature-Based TAG. In *Proceedings of the 10th Meeting of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 123–130. ACL anthology: E03-1030 .

Aravind K. Joshi and Yves Schabes. 1997. Tree-Adjoning Grammars. In Grzegorz Rozenberg and Arto K. Salomaa, editors, *Handbook of Formal Languages*, volume 3, chapter 2, pages 69–123. Springer, Berlin.

Laura Kallmeyer and Aravind K. Joshi. 2003. Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *Research on Language and Computation*, 1(1–2):3–58. DOI: 10.1023/A:1024564228892 .

Laura Kallmeyer and Rainer Osswald. 2013. Syntax-driven semantic frame composition in lexicalized tree adjoining grammars. *Journal of Language Modelling*, 1(2):267–330. DOI: 10.15398/jlm.v1i2.61 .

Laura Kallmeyer and Maribel Romero. 2008. Scope and situation binding in LTAG using semantic unification. *Research on Language and Computation*, 6(1):3–52. DOI: 10.1007/s11168-008-9046-6 .

Manfred Krifka. 1998. The origins of telicity. In Susan Rothstein, editor, *Events and Grammar*, pages 197–235. Kluwer, Dordrecht.

Sebastian Löbner. 2014. Evidence for frames from human language. In Gamerschlag et al. (Gamerschlag et al., 2014), pages 23–67. DOI: 10.1007/978-3-319-01541-5_2 .

Rainer Osswald and Robert D. Van Valin Jr. 2014. Framenet, frame structure, and the syntax-semantics interface. In Gamerschlag et al. (Gamerschlag et al., 2014), chapter 6, pages 125–156. DOI: 10.1007/978-3-319-01541-5_6 .

Wiebke Petersen. 2007. Representation of concepts as frames. In *The Baltic International Yearbook of Cognition, Logic and Communication*, volume 2, pages 151–170.

Zeno Vendler. 1957. Verbs and times. *The Philosophical Review*, 66(2):143–160. DOI: 10.2307/2182371 .

K. Vijay-Shanker and Aravind K. Joshi. 1988. Feature structures based tree adjoining grammar. In *Proceedings of the 12th International Conference on Computational Linguistics, COLING Budapest*, pages 714–719, Budapest. ACL anthology: C88-2147 .