Fachbereich Sprachwissenschaft

Universität Konstanz

Arbeitspapier (to appear)

MARKUS EGG

Reinterpretation from a Synchronic and a Diachronic Point of View

In: Regine Eckardt & Klaus von Heusinger (eds.) Meaning Change – Meaning Variation. Workshop held at Konstanz, Vol. II.

Reinterpretation from a synchronic and a diachronic point of view*

Markus Egg, Universität des Saarlandes
egg@coli.uni-sb.de

Abstract

Frequently an utterance can only be understood if one integrates additional material into its meaning, which mediates between semantically conflicting parts of the utterance. This process is known as *reinterpretation*. From a synchronic viewpoint, it is a 'creative' or 'dynamic' aspect of natural language, to be described and integrated in a formal description of natural language semantics. But reinterpretation phenomena can also be regarded as a gateway for linguistic change, since they may get conventionalized and thus enlarge the domain of compositional semantics. Analyzing reinterpretation will therefore also provide insights into mechanisms of linguistic change.

The proposed account of reinterpretation goes as follows. Semantic construction yields ambiguous structures for reinterpretation cases, which are then monotonically enriched with information from extralinguistic sources (e.g., world knowledge). Semantic ambiguities are described in the framework of underspecification. This account of reinterpretation allows a straightforward modelling of its synchronic and diachronic aspects.

1 Introduction

Reinterpretation arises from a semantic incompatibility between parts of a larger constituent. Cases of aspectual reinterpretation like (1) exemplify this phenomenon: the aspectual properties of the modifier *for ten days* (semantically, the functor) and the modified expression (the sentence radical[1] *Amélie play- the sonata*; semantically, the argument) are at variance: the argument is bounded[2], but the functor needs an unbounded argument.

*The research presented in this paper was supported by the Deutsche Forschungsgemeinschaft in the project CHORUS within the SFB 361 „Ressourcenadaptive kognitive Prozesse". Thanks to my colleagues for comments and discussion, in particular to Alexander Koller and Kristina Strignitz for their feedback on a first version of this paper.

[1]Throughout the paper, issues of sentence mood and tense are neglected. Sentences minus mood and tense (called 'sentence radicals') are written with an uninflected main verb.

[2]Bounded expressions have in their extension no proper parts of elements of their extension.

(1) Amélie played the sonata for ten days

Nevertheless, the sentence is acceptable. This is due to reinterpretation: confronted with (1), hearers reinterpret the modified expression *iteratively* (i.e., Amélie played the sonata repetitively). This makes the modified expression unbounded, which fits the restriction of the modifier.

The topic of this paper is *reinterpretation*, both from a synchronic and a diachronic point of view. From a synchronic point of view, reinterpretation is a feature of natural language that introduces a certain amount of flexibility into generating and understanding complex expressions. It must be integrated into a formal semantic framework (Dölling 1992, Pustejovsky 1995, Moens und Steedman 1988).

But reinterpretation is also relevant from a diachronic perspective on language. In this view, it emerges as a gateway for linguistic change, since reinterpretation phenomena may get conventionalized and thus become part of compositional semantics. Thus, research on reinterpretation will also provide insights into mechanisms of linguistic change.

The present paper proproses an analysis of reinterpretation that allows a straightforward and intuitive modelling of these synchronous and diachronous aspects of reinterpretation. The analysis is based on a view of reinterpretation where the full meaning of a reinterpreted expression is determined by the *interaction of various knowledge sources* (including, but not restricted to, the syntax and semantics of the expression). Since semantic construction cannot fully determine the meaning of a reinterpreted expression, it must yield *ambiguous semantic representations* for these expressions. These ambiguities 'leave room' for other knowledge sources to contribute to the meaning of the reinterpreted expression. E.g., the iterative operator in the interpretation of (1) is provided by world knowledge that the typical duration of an iteration of sonata playings is compatible with a duration of ten days.[3] Integrating the material from the relevant other knowledge sources then disambiguates these representations. Thus, reinterpretation emerges as a *monotonic* process, not as a destructive process of undoing results of semantic construction.[4]

Consider a very simplistic example for the proposed analysis of reinterpretation. (It serves expository purposes only, see section 4.3.) The ambiguous (2a) models the semantics of (1). The dots indicate a position between functor and argument that is not determined by semantic construction and allows integrating information from other sources, e.g., an iterative operator. This results in (2b), which models the meaning of (1) after reinterpretation, viz., the proposition that Amélie played the sonata repeatedly for ten days:

(2) (a) $\mathbf{for\_10d}(\ldots\mathbf{play}(\mathbf{a},\mathbf{the\_s}))$ (b) $\mathbf{for\_10d}(\mathrm{ITER}(\mathbf{play}(\mathbf{a},\mathbf{the\_s})))$

The semantic ambiguities that emerge in the analysis of reinterpretation cases are described non-disjunctively in the framework of *underspecification*: semantic representations do not give the meaning of an expression, but merely *descriptions* of its meaning. Selecting a

[3]The issue of how to store and to retrieve such additional knowledge is not in the focus of this paper, which concentrates on the construction of semantic structures for reinterpreted expressions.

[4]Unfortunately, the very term 'reinterpretation' suggests an explanation of the phenomenon in terms of non-monotonic, destructive processes. But since the expression 'reinterpretation' is so well-established, I will continue to use it, although I do not subscribe to this explanation of the phenomenon.

reading of an reinterpreted expression is then modelled as a monotonic process of adding further descriptions, which suits the view on reinterpretation advocated in this paper.

The paper is structured as follows: After an exposition of patterns of reinterpretation in section 2, section 3.1 introduces the framework of underspecification. Section 3.2 outlines the representation language CLLS, which is used for the analysis of reinterpretation. The synchronous aspects of reinterpretation are covered in section 4. Finally, section 5 discusses how reinterpretation may lead to linguistic change.

2 Reinterpretation phenomena

After the more general introduction of the phenomenon of reinterpretation in the preceding section, this section is devoted to an analysis of different patterns of reinterpretation and of the different knowledge sources that interact in reinterpretation.

2.1 Type coercion vs. landing site coercion

The first pattern of reinterpretation is exemplified in (1) and (3). (1) was expounded in section 1. In (3), argument (the semantics of the verb, a relation between individuals and *masses*) and functor (the semantics of the rest of the sentence, a property of relations between individuals and *things*) do not fit, since the relevant sorts MASS and THING are disjoint. This sortal mismatch is resolved by an operator that maps the argument onto a relation between individuals and things. The preferred operator for (3) introduces the notion of *container*. It is based on world knowledge about the use of glasses as containers for beverages. The resulting reading is that Amélie downed the content of every glass:

(3) Amélie downed every glass

Examples (1) and (3) follow the same reinterpretation pattern: to mediate between conflicting properties of functor and argument that are to be combined, an operator is applied to the argument first. This operator is not part of either functor or argument, hence, additional material – beyond the one provided by semantic construction – must be derived and integrated with the functor-argument pair. If F stands for the functor, A, for the argument, and Op, for the additional material, the general pattern of type coercion is $F(Op(A))$. This pattern is familiar from the literature, it has been dubbed *type coercion* (TC) by Moens und Steedman (1986, 1988). They discuss TC in the aspectual domain, while Dölling (1992, 1995) and Pustejovsky (1995) focus on TC in the nominal domain.

But reinterpretation cases like (4) call for a different explanation:

(4) Amélie left for two hours

(4) exhibits the same aspectual incompatibility as (1). But the preferred way of reconciliating the conflicting properties of functor (the semantics of *for two hours*) and argument (the semantics of *Amélie leave-*) in (4) is different: the functor is applied not to the argument as a whole but to only a part of it. Reinterpretation augments the argument so

that it comprises a part to which the functor may be applied alternatively (an additional 'landing site' for the functor).

For (4), such an additional part of the argument is introduced that refers to an *aftermath* associated with eventualities (this term refers to all kinds of states of affairs) in the extension of the argument. Since the property of being an aftermath of some other eventuality is unbounded, this part of the argument fits the aspectual selection restriction of the adverbial. The resulting interpretation of (4) is that it is not Amélie's leaving but the aftermath of her leaving (her subsequent absence) that lasted for two hours.

This reinterpretation is based on conceptual knowledge. Following Moens and Steedman, it is assumed that conceptually the flow of eventualities is carved in small portions, called 'nuclei'. Nuclei consist of an eventuality together with associated preparatory phase and aftermath. These additional eventualities can be referred to in reinterpretation operators. The abstract pattern of this kind of reinterpretation is A & $a_1(F(a_2))$, where $a_1(a_2)$ is the augmentation of the argument A that introduces the additional landing site a_2 for the functor F ('&' is the generalized conjunction of Rooth und Partee 1982 that conjoins any pair of expressions that have the same type). The pattern is called *landing site coercion.*

Landing site coercion is not restricted to the aspectual domain. It is shown in section 4 that the Pustejovsky example *fast road* also instantiates this phenomenon. Since roads (being immobile things) are themselves not fast, *fast road* must be understood differently, e.g., as 'road along which one can drive fast', where driving along the road is its preferred (but not its only possible) purpose.

In this subsection, two patterns of reinterpretation were distinguished, viz., type coercion, and landing site coercion. An adequate account of these patterns is a first challenge for an analysis of reinterpretation.

2.2 Semantic and non-semantic parts of reinterpretation

The interpretation of reinterpreted expressions involves the integration of additional material. Its origin is neither functor nor argument, hence, syntax and semantics of reinterpretation expressions do not provide it. I.e., syntax and semantics cannot fully determine the meaning of these expressions, which brings forth ambiguity on the semantic level.

But due to the interaction between these knowledge sources, few examples illustrate this point clearly. Language understanding hardly ever proceeds in zero contexts, in particular, one cannot shut off world knowledge, which often immediately resolves ambiguities. E.g., nothing in the semantics of (3) bars its interpretation as 'Amélie downed the molten substance that constituted the glass', but world knowledge prefers the 'container' reading over this reading. Similarly, knowledge about the typical duration of sonatas triggers the choice of the iterative operator in the interpretation of (1).

Examples that are clearly ambiguous are those with more than one equally plausible meaning. For (5), one cannot decide out of context whether the argument is reinterpreted by inserting the progressive or the iterative operator. Both options meet the aspectual requirements of functor and argument. In contrast to (1), the modifier in (5) is too vague to contribute to the disambiguation. Bierwisch's example (6) is analogously ambiguous in that reinterpretation may involve the *writings* or the *thoughts* of Faulkner.[5]

(5) Amélie played the sonata for some time

(6) Faulkner is difficult to understand

These examples show that even if functor and argument in reinterpretation cases are not ambiguous, ambiguity arises as soon as they are combined. Resolving this ambiguity requires extralinguistic knowledge sources. Therefore, the semantic boundary must be delimited carefully in analyses of reinterpretation. The semantic representation of reinterpretation expressions must leave room for the contribution of additional knowledge sources to the determination of their meaning. The method of representing these ambiguities adopted in this paper is *underspecification*, which is the topic of the following subsection.

3 The formal background

In the first part of this section, the framework of underspecification is introduced along with its application to the task of accounting for ambiguity in natural language. The second part presents the chosen representation language CLLS, in particular, the adapted version of CLLS that is used for the account of reinterpretation in this paper.

3.1 The framework of underspecification

The proposed analysis regards reinterpretation cases as ambiguous on the semantic level. These ambiguities are captured in terms of underspecification. Thus, the presented analysis is part of a larger enterprise to describe ambiguity in natural language in terms of underspecification on the basis of suitable representation languages like Quasi-Logical Form (Alshawi 1992), Underspecified DRT (UDRT, Reyle 1993), Underspecified Semantic Description Language (USDL, Pinkal 1996), or Constraint Language for Lambda Structures (CLLS, Egg et al. 1998).

The crucial idea of underspecification is a different use of semantic representations. They *describe* fully specified meanings of linguistic expressions (rather than *denoting* them directly). This makes possible a compact, non-disjunctive account of ambiguity. Descriptions can be too unspecific to single out a single fully specified meaning. In that case, they are compatible with a whole set of such meanings. If this set corresponds to the set of readings of an ambiguous expression, the unspecific description captures the meaning of this expression exhaustively and non-disjunctively. Each meaning that is compatible with the description corresponds to one of the readings of the expression and vice versa.

The well-known ambiguous (7) is an obvious candidate for this kind of analysis. An underspecified account of (7) is a description that is compatible with exactly two meanings. They correspond to the readings of (7) and instantiate one of the two scoping possibilities for the two quantifiers. See (10) below for a formal spellout of this ambiguity.

[5] The reading of *understand* that is at issue here is ‘comprehend’. It requires an object NP whose head refers to abstract information structures. Reinterpretation is called for if the object NP is *Faulkner*.

(7) Every woman loves a man

This approach has highly desirable consequences both from a cognitive and a computational point of view. Consider e.g. the notorious example (8) of Hobbs und Shieber (1987):

(8) Every researcher of a company saw most samples

This example has exactly five readings. Most speakers could never identify them, though they understand the sentence. This suggests that the mental representations of ambiguous sentences abstract away from the *differences* between the readings as spelt out in the fully specified readings but highlight the *common ground* between these readings. Representing ambiguous expressions by underspecification closely models this insight. From a computational point of view, the framework of underspecification has the additional advantage that processing semantic representations of ambiguous expressions (e.g., in Machine Translation) is often considerably easier since disjunctions are avoided.

So far, research in the framework of underspecification focuses on ambiguities arising from quantifier scope and the interaction of ellipsis with phenomena like anaphora and scope. Reinterpretation is another field of ambiguity, hence, underspecified approaches must cover it, too. It is shown that the approach of Egg et al. (1998) fulfills this requirement.

Using the framework of underspecification for an account of reinterpretation is slightly more complex than approaches like Dölling's (1995) or Pulman's (1997). They insert variables between functor-argument pairs during semantic composition, which can be instantiated afterwards by suitable additional (conceptual) knowledge.

The complexity is justified by two observations: First, reinterpretation and phenomena like quantifier scope may cooccur in a linguistic expression. This necessitates an underspecification framework to describe the full range of reinterpretation data. Second, landing site coercion (LSC) calls for an analysis whose level of complexity can no longer be handled appropriately by merely introducing free variables into semantic representations, which will be shown in section 4.3.

The first argument is illustrated by (9), a variant of example (3) where subject and object NPs introduce a quantifier. (9) shows that reinterpretation and phenomena like quantifier scope ambiguity (here, between subject and object NP) may cooccur.

(9) A man downed every glass

The semantic ambiguities that these phenomena introduce are independent of each other. One can reinterpret these representations or determine scope relations between their quantifiers without affecting any ambiguities that arise from the respective other source (see Koller et al. 1999). This calls for an integrated treatment of all kinds of semantic ambiguity, which justifies the complexity of the chosen approach to reinterpretation.

What is more, LSC as presented in section 2.1 requires the very same kind of complexity of the semantic description formalism that is needed for quantifier scope ambiguity: the semantic structure that results from LSC is the same as for these scope ambiguities (see section 4.3). The underspecification framework provides description formalisms of the requested complexity.

After this introduction to the framework of reinterpretation, the next subsection presents the representation language that is used for the account of reinterpretation in this paper.

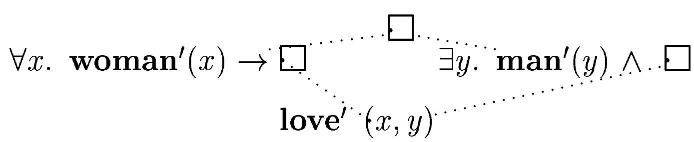
3.2 The representation language CLLS

The language CLLS (Egg et al. 1998) was designed for classic cases of structural ambiguity, especially those that come with scope underspecification and/or ellipsis reconstruction.
In CLLS, semantic representations are *constraints* on meanings. CLLS representations are based on *tree structures* that correspond to λ-terms. This basis makes CLLS a very powerful and flexible formalism. The notation of CLLS constraints is adapted to the needs of this paper. For the full details of CLLS the reader is referred to Egg et al. 1998.
In the adapted notation, CLLS constraints describe λ-terms. They consist of the following:

- fragments of λ-terms ('LT fragments')
- positions within LT fragments where other fragments can be inserted: 'holes' ($\Box$)
- dominance relations that are indicated by dotted lines
- sort information that is given by sorted variables

Dominance relations introduce underspecification. Imagine a constraint where a hole in an LT fragment dominates another LT fragment. This expresses only that the first fragment outscopes the second. Other fragments of the constraint or new material not yet present in the constraint may intervene scopally between the two parts. I.e., the representations are not partial orders of semantic fragments like e.g. in UDRT. This possibility of integrating new material into constraints is crucial for the description of reinterpretation phenomena.
(10), the representation of (7), illustrates this adapted CLLS notation:

(10)



Such structures provide partial descriptions of λ-terms. Holes represent parts of LT fragments that are not yet completely fixed. But whenever a hole dominates an LT fragment, this means that the fragment is an (im-)proper part of the material represented by the hole. E.g., the scope of either quantifier in (10) is described by a hole, since it is not yet fixed. All that one knows right from the start is that the verbal nucleus (the bottom fragment **love′**(x, y)) is part of either scope.[6] Similarly, the hole at the top shows that the semantics of the whole sentence is still open, but either quantifier fragment must be part of it. This 'dominance rhombus' appears frequently in semantic representations for scopally ambiguous expressions.
The semantics of an ambiguous expression is given by a constraint like (10). All λ-terms that are compatible with the constraint stand for one of the readings of the expression. One can derive these λ-terms by filling all holes (this must respect the dominance relations). If a λ-term can be derived in this way without using material not yet present in the constraint, it is a *constructive solution* of the constraint.

[6] CLLS proper does not use variable names to model the binding of variables, in contrast to the adapted notation. Instead, binding is explicated by a function λ. Roughly, this function relates variables to their binders. This point is of no importance here, hence neglected in the adapted notation.

For (10), there are two constructive solutions. Either quantifier fragment can fill the top hole, which forces the respective other quantifier fragment to fill in the scope of the first one, with the verbal nucleus being the scope of the second quantifier. These options give the λ-terms for the two readings of (10):

(11) $\forall x.\mathbf{woman'}(x) \rightarrow \exists y.\mathbf{man'}(y) \wedge \mathbf{love'}(x, y)$

(12) $\exists y.\mathbf{man'}(y) \wedge \forall x.\mathbf{woman'}(x) \rightarrow \mathbf{love'}(x, y)$

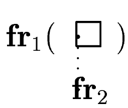
After this account of the framework of underspecification and the representation language CLLS, the next sections analyze synchronic and diachronic aspects of reinterpretation.

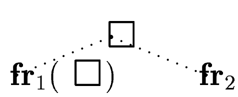
4 Representing reinterpretation

In this section, the synchronic aspects of reinterpretation are analyzed in detail after some preliminaries, first the phenomenon of type coercion, and next, landing site coercion.

4.1 Preliminaries

Reinterpretation is seen as a monotonic process, where the full meaning of a reinterpreted expression is derived by enriching a well-formed but not yet fully determined (hence ambiguous) result of semantic construction with material from extralinguistic sources.
To reach this goal, potential reinterpretation sites ('R-sites') must be distinguished. R-sites involve a functor-argument pair. In the cases discussed in this paper, functor and argument can be identified with the semantic contributions of a modifier/subcategorizer and of its modified/subcategorized expression, respectively. Verbs subcategorizing for NPs are exceptional, here the NP provides the functor, and the verb, the argument.
For functor-argument pairs that might be affected by reinterpretation, semantic construction avoids combining potentially conflicting material of functor and argument in one single LT fragment. The resulting semantic representations are constraints that consist of several fragments. Dominance relations connect these fragments. Schematic examples of such constraints are given in (13), where the $\mathbf{fr}_n$ are abbreviations for LT fragments:[7]

(13) (a) 

(b) 

This means in the first place that any incompatibilities between functor and argument cannot make the result of semantic construction ill-formed, since they are buffered by dominance relations. Semantic construction can proceed without having to bother about potential mismatches between functor and argument.
Second, the dominance relations between the LT fragments determine positions in semantic representations for reinterpreted expressions where additional material can be inserted. I.e.,

[7]These types of constraints, which are characteristic for type coercion and landing site coercion, respectively, will be discussed extensively in sections 4.2 and 4.3.

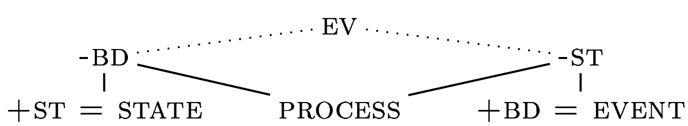
the representations are compatible with more than one λ-term, hence, an underspecified representation of a semantic ambiguity.

In particular, these dominance relations mark positions where the material that is to be contributed by additional, extralinguistic sources must be inserted to fully specify the meaning of the respective reinterpreted expression. Thus, semantic construction strictly restricts itself to the semantic parts of such a meaning, which leaves enough space for extralinguistically derived material. The integration of this additional information is hence monotonic, it need not undo results of semantic construction.

Some comments on the representation of aspect are in order here. Aspect is encoded in sorts of eventualities. Different aspectual classes are modelled as incompatible sorts. In particular, *boundedness* and *unboundedness* are distinguished by the sorts ±BD. This has the consequence that no overlap is assumed between the eventualities in the extension of an unbounded expression and those in the extension of a bounded expression, although eventualities of different aspectual classes may spatiotemporally coincide. For instance, no entity can be in the extension of both *Amélie smile-* and *Amélie smile- for two hours.* This distinction reappears in the nominal domain, where e.g. groups and the pluralities of individuals that constitute them are distinguished in spite of their material identity.

The aspectual properties *stativity/non-stativity* are distinguished by the sorts ±ST. Stativity implies unboundedness. This groups unbounded predicates (the term 'predicate' refers to verbs and their projections [e.g., VPs or sentence radicals]) into (stative) *state predicates* (e.g., *love*, *be in the pub*) and (non-stative) *process predicates* (e.g., *run*, *shake*).[8]

In the sort hierarchy, eventualities (sort EV) are cross-classified (this is indicated by the dotted lines) in unbounded and non-stative eventualities. Process eventualities emerge as the intersection of unbounded and non-stative eventualities:



Finally, the operators that show up in the following are defined, first, the *progressive operator* PROG. PROG(P) holds for e iff there is an eventuality e' for which P holds, and the spatiotemporal extensions of e and e' stand in the part-of relation. In other words, e' is ongoing at e. This is simplified: assuming the existence of such an e' neglects the fact that PROG(P) may hold for an e even if the respective e' is never completed (the 'imperfective paradox'). PROG selects for non-stative properties and returns a stative one.

The *maximality operator* MAX maps properties P onto the bounded property of being a 'maximal phase' of P (Herweg 1991). MAX(P) holds for an eventuality e iff P holds for an e' that spatiotemporally coincides with e, but for no e'' of which e' is a proper part.

The *genericity operator* GEN$[e](R, C)$ is a quantifier over eventualities e, such that R generally implies C (see Krifka et al. 1995 for detailed discussion of GEN).

[8] Aspectual classification is based on so-called *compatibility tests*. Compatibility (without reinterpretation) with durative adverbials distinguishes unbounded from bounded predicates, with time frame adverbials like *in three hours*, the pattern is the reverse. Stative predicates disallow the progressive and modification by adverbials like *quickly*. See Dowty (1979) for an extensive list of such tests.

The semantics of durative adverbials is rendered in terms of the *S-quantity operator* PH. $\mathrm{PH}(\mathbf{Q}, P)$ holds for an eventuality e iff there is an e' in the extension of P that spatiotemporally coincides with e, and a suitable measure function returns $\mathbf{Q}$ as the value of the length of e (see Egg 1995 for further discussion of the S-quantity). The argument of the durative adverbial must be unbounded, the resulting expression is bounded.[9]

4.2 Representing type coercion

R-sites for type coercion are determined in the lexicon. Semantic construction is lexically driven in that modifiers/subcategorizers relate the constraints of functor and of argument. They differ in the way they determine this relation. If they introduce an R-site for TC, the relation is one of *dominance* between a hole in the functor constraint and the argument constraint. If they do not give rise to reinterpretation, the argument constraint is inserted into the functor constraint. The constraint of a modified/subcategorized expression is made available for its modifier/subcategorizer e.g. as the value of some feature CONT|CON in an attribute-value matrix.[10]

E.g., the semantics of the progressive operator, which introduces an R-site for TC, is given in (14a). The predetermined position of the semantics of its argument is indicated by the place-holder '$\underline{\text{arg}}$'. It is separated by a dominance relation from the semantics of the operator. In contrast, in the semantics of *yesterday* (14b), the constraint $\underline{\text{arg}}$ of the modified expression is inserted in the modifier constraint. This indicates that *yesterday* differs from the progressive in that it does not give rise to type coercion:

(14) (a) $\lambda e_{+\mathrm{ST}}.\mathrm{PROG}(\lambda e'_{-\mathrm{ST}}\ \begin{array}{c}\square\\ \vdots\\ \underline{\text{arg}}\end{array}\ (e'))(e)$ (b) $\lambda e.\mathbf{yesterday'}(e) \wedge \underline{\text{arg}}(e)$

The resulting structure of R-sites for TC is depicted in (15a), viz., a functor and an argument that are separated by a dominance relation. This dominance relation determines the position where additional material (in the form of an operator *Op*) must be integrated (15b). This material is inserted into the semantic representation (15c). By closing the gaps in the augmented constraint (15c) one obtains its constructive solution, the λ-term (15d):

(15) (a) $F(\begin{array}{c}\square\\ \vdots\\ A\end{array})$ (b) $F(\begin{array}{c}\square\\ \vdots\\ A\end{array}) \Leftarrow Op(\ \square\ )$ (c) $\begin{array}{c}F(\ \square\ )\\ \vdots\\ Op\ (\begin{array}{c}\square\\ \vdots\\ A\end{array})\end{array}$ (d) $F(Op(A))$

It does not matter whether a given R-site actually gives rise to reinterpretation. If not, the argument constraint can be inserted into the hole of the functor later on.

[9] Note that the complexity of this definition is not a consequence of the chosen approach to reinterpretation, it emerges for *any* account of the DA semantics that introduces some sortal distinction between stative/nonstative (e.g., Dölling 1998) or bounded/unbounded predicates (e.g., Herweg 1991).

[10] Koller et al. (1999) introduce R-sites in the syntax-semantic interface. The lexical introduction as advocated here is more flexible: it can model the fact that R-sites and non-R-sites may have the same constituent structure. This is relevant for landing site coercion, see section 4.3.

Consider e.g. the meaning of (3) (omitting eventuality arguments). Semantic construction yields the constraint on the left hand side of (16). This constraint is an R-site for TC. The scope of the functor (the semantics of the sentence minus the verb) is not yet determined, hence given by a hole. The argument (the verb semantics) is part of the scope:

(16) $\forall x_{\text{THING}}.\mathbf{glass}'(x) \rightarrow \square\ (\mathbf{am\acute{e}lie}', x)$ $\vdots$ $\lambda y_{\text{MASS}}\lambda x.\mathbf{down}'\ (x, y)$ $\Leftarrow$ $\lambda z_{\text{THING}}\lambda x \exists y_{\text{MASS}}.\mathbf{contain}'(z, y) \wedge\ \square\ (x, y)$

Closing the gap would return an ill-formed λ-term, as it would impose conflicting sortal requirements onto the quantified variable. Additional material must intervene, e.g., the right hand side constraint in (16). It introduces an operator that maps a relation P between individuals x and masses y onto a relation between x and the container z of y (z is of sort THING): this new relation holds between x and containers z if P holds between x and y, the content of z. E.g., if P is the relation of downing, the relation holds between Amélie and a glass if Amélie downs the content of the glass.
After the insertion of the constraint, the gaps can be closed. The resulting λ-term is (17). It expresses the desired interpretation of (3) that Amélie downed the content of every glass.

(17) $\forall x_{\text{THING}}.\mathbf{glass}'(x) \rightarrow \exists y_{\text{MASS}}.\mathbf{contain}'(x, y) \wedge \mathbf{down}'(\mathbf{am\acute{e}lie}')(y)$

The predetermination of R-sites prevents overgeneration: resolving the sortal conflict in (3) by deriving an R-site between the semantics of the verb and the one of the rest of the sentence (rather than between object determiner and noun semantics) correctly predicts that the quantification in (3) is on glasses and not on drinks. However, the Nunberg (1993) style example (18) shows that syntax is just one of the determinants of this predetermination. In (18), quantification is on customers, not on sandwiches, which calls for an R-site where the $\bar{\text{N}}$ semantics is the argument in order to make room for an operator from properties of food onto properties of its consumers.

(18) Every ham sandwich complained

(19) $\forall x_{+\text{ANIM}}.\ \square\ (x) \rightarrow \mathbf{complain}'(x)$ $\vdots$ $\lambda y_{\text{-ANIM}}.\ \mathbf{hamsandwich}'(y)$ $\Leftarrow$ $\lambda z_{+\text{ANIM}} \exists y_{\text{-ANIM}}.\mathbf{consume}'(z, y) \wedge\ \square\ (y)$

The analysis of (18) is given in (19). The sorts $\pm$ANIM model the relevant distinction between animate and inanimate individuals. Again, the result of semantic construction is on the left, the extralinguistically derived operator, on the right. The left constraint is an R-site whose argument is the $\bar{\text{N}}$ semantics. Combining both constraints yields the λ-term (20) for the proposition that every consumer of a ham sandwich is angry:

(20) $\forall x_{+\text{ANIM}}.(\exists y_{\text{-ANIM}}.\mathbf{consume}'(x, y) \wedge \mathbf{hamsandwich}'(y)) \rightarrow \mathbf{angry}'(x)$

Separating the verb semantics from the semantics of the rest of the sentence by an R-site like in (16) would lead to a wrong analysis with the quantification on ham sandwiches. There are some speculations about possible causes for these differences in reinterpretation (e.g., Nunberg 1993 assumes that the choice is driven by contextual relevance), but still it is fair to say that further work is necessary to determine the full range of factors that control reinterpretation and to integrate them into a formal account of reinterpretation.

Aspectual TC is illustrated by (21), which features an incompatibility between the progressive, which selects for nonstative predicates, and its stative argument *Amélie be- silly*. The sentence radical of (21) is accounted for analogously to (3). Its semantics is given by the right hand side in (22). The functor (the progressive operator) provides the dominating fragment, the argument (the semantics of *Amélie be- silly*), the dominated one.

(21) Amélie was being silly

(22) $\lambda e_{+\mathrm{ST}}.\mathrm{PROG}(\lambda e'_{\text{-ST}}\ \square\ (e'))(e)$ $\Leftarrow$ $\lambda e''_{+\mathrm{BD}}.\mathrm{MAX}(\ \square\ )(e'')$

$\lambda e_{+\mathrm{ST}}.\mathbf{silly}'\ (\mathbf{amélie}')(e)$

Short of imposing conflicting aspectual requirements on the λ-abstracted variable e', the gap between functor and argument may not simply be closed. The preferred augmentation of this structure is the operator MAX, as depicted on the right hand side of (22). It mediates between the conflicting aspectual properties of functor and argument.

For (22), this models the intuition that Amélie is silly voluntarily: if her silliness is bounded, it is only temporary. Thus, it is none of her traits, hence, something she chooses to adopt. Inserting the constraint and closing the remaining gaps yields the λ-term (23) for the set of eventualities at which maximal phases of Amélie's being silly are ongoing:

(23) $\lambda e_{+\mathrm{ST}}.\mathrm{PROG}(\lambda e'_{+\mathrm{BD}}.\mathrm{MAX}(\lambda e''_{+\mathrm{ST}}.\mathbf{silly}'(\mathbf{amélie}')(e''))(e'))(e)$

To sum up, type coercion is modelled by determining lexically the positions where it is possible and inserting dominance relations at these positions, which allows for a straightforward semantic construction and further determination of instances of type coercion.

4.3 Representing landing site coercion

This subsection is devoted to an account of landing site coercion (LSC). Its goal is to develop an analysis that allows a straightforward semantic composition even for LSC cases. There are three main problems that must be overcome in such an analysis.

The first issue is the question of *compositionality*. The representation of LSC must be capable of inserting a functor within its argument without violating the maxim that the argument is opaque for the process of semantic construction.

Second, reinterpretation should be triggered *in the same fashion* for both type coercion and LSC, since within semantic construction one cannot decide whether a given functor-argument pair is an instance of TC, LSC, or no reinterpretation case at all. Thus, once again it is the task of lexical semantics to determine the R-sites for LSC.

This strategy comes in handy for the issue of *control*, the third problem for this reinterpretation pattern: the observation is that *both* functor and argument must license LSC. This is illustrated by two variants of the LSC example (4). Neither of them can undergo LSC, although each comprises either the modifier or the modified expression of (4):

(24) Amélie left yesterday

(25) Amélie laughed for two hours

This shows that the information of whether an expression may engage in LSC or not must be provided by an appropriate lexical representation of the expression. LSC may happen to a constituent only if none of its immediate subconstituents rules out LSC lexically.
The account of LSC starts with a related phenomenon, the modification of nomina agentis like in *fast eater* ('agentive modification'). This is no instance of LSC (since it belongs wholly to the realm of compositional semantics), but is extremely useful for introducing a pattern of semantic composition which is needed for LSC cases.
When modifying agentive nouns, modifiers like *fast* may pertain to the noun as a whole or only to the verb stem that is part of the noun. The two readings are 'fast person with the habit of eating' and 'person with the habit of eating fast'. If we represent habituality by the genericity operator GEN, the two readings are represented by the λ-terms in (26):[11]

(26) (a) $\lambda x.\mathbf{fast'}(x) \wedge \mathrm{GEN}[e](x\ \mathbf{in}\ e, \mathbf{eat'}(x)(e))$
(b) $\lambda x.\mathrm{GEN}[e](x\ \mathbf{in}\ e, \mathbf{fast'}(e) \wedge \mathbf{eat'}(x)(e))$

To derive these readings, the noun semantics is represented by a twopartite constraint that consists of two fragments, with a dominance relation between them. The embedded fragment is an additional landing site for the semantics of modifiers like *fast*:

(27) $\lambda x.\mathrm{GEN}[e](x\ \mathbf{in}\ e,\ \square\ (e))$
$\lambda e.\mathbf{eat'}\ (x)(e)$

The top fragment is the semantics of the suffix *-er*, the embedded fragment comes from the semantics of the verb stem *eat*. The constraint for the semantics of the whole noun is given as the value of the feature CONT|CON as introduced above. In addition, the embedded fragment is also specified as the value of another feature CONT|EMCON (for 'embedded constraint'). If the constraint for semantics of a lexeme (e.g., of *to walk*) has no such twopartite structure, the value of this feature is not specified.
If the semantics of a modifier (like *fast* or like durative adverbials) may optionally dominate only an embedded fragment in the semantics of the modified expression, this is specified lexically. Such a modifier relates its constraint to the constraint of its modified expression via a dominance relation between a hole in its constraint and the embedded fragment in the constraint of the modified expression, i.e., its CONT|EMCON value. This strategy allows inserting a functor within its argument. It will be reused in the analysis of LSC, which answers the first of the issues for LSC analyses as noted above.
For *fast eater*, the result (28) is a dominance rhombus, since *fast* relates its semantics to the CONT|EMCON value of its argument, and this value is specified in the argument. The hole of the functor dominates the embedded fragment of the argument (the verb stem

[11] The same ambiguity emerges if *fast* is analyzed as non-intersective modifier. 'x **in** e' means that x participates in e.

semantics). The relation between the functor and the part of the argument contributed by the suffix is open. The two constructive solutions of (28) (where all the gaps are closed) are the λ-terms (26a) and (26b) for the two readings of *fast eater*.

(28)
$$\Box$$
$$\lambda y.\ \mathbf{fast'}(y) \wedge \Box\ (y) \qquad \lambda x.\ \text{GEN}[e](x\ \mathbf{in}\ e, \Box\ (e))$$
$$\lambda e.\ \mathbf{eat'}\ (x)(e)$$

Structures like (28) are the goal of LSC. In contrast to agentive modification, these structures are not the result of semantic composition in LSC cases, but the result of augmenting the result of semantic composition.

LSC is similar to agentive modification in that a hole in the functor constraint must dominate the value of CONT|EMCON of the argument. The difference between LSC and nomina agentis cases is that the argument fails to specify the CONT|EMCON value, since its constraint has no embedded fragment that may serve as landing site for the functor. Thus, semantic construction cannot decide the scope of functor and argument, it can only deduce that both are dominated by a common hole. These two features of LSC are introduced in the lexical entries of the constituents that provide functor and argument. This lexical determination of R-sites for LSC accomplishes the second of the issues for LSC analyses.

Consider (29), the semantics of (4). Semantic construction merely states that functor and argument are dominated by a hole at the top of the constraint for the sentence radical. This is due to the fact that durative adverbial lets the hole of its constraint dominate the value of CONT|EMCON of *Amélie leave-*. But this value is not specified in the semantics of the modified expression.

(29)
$$\Box$$
$$\lambda e_{+\text{BD}}.\ \text{PH}(\mathbf{2h}, \lambda e'_{\text{-BD}}.\ \Box(e'))(e) \qquad \lambda e'_{+\text{BD}}.\mathbf{leave'}\ (\mathbf{amélie'})(e')$$

In (29), the gaps cannot simply be closed, since the conflicting aspectual requirements of functor and argument prevent the functor from directly dominating the argument. (29) can be augmented in terms of an LSC that bridges the aspectual conflict.[12] One can insert an operator directly above the argument that introduces an additional landing site for the functor. For (29), the relevant operator is (30):

(30)
$$\lambda e_{+\text{BD}}.\Box(e) \wedge \exists e''.(\ \Box\ )(e'')$$
$$\lambda e''_{+\text{ST}}.\ \text{RES}(e'', e)$$

(30) applies to bounded predicates like *Amélie leave-*. This is expressed in the sort +BD of the λ-abstracted variable e. In an embedded fragment, (30) explicitly introduces the notion of aftermath: RES relates eventualities with their aftermaths. The property of being an aftermath of some eventuality is stative. The embedded fragment fills in the missing value of CONT|EMCON for the predicate to which it applies. (30) maps properties of eventualities

[12] Type coercion might bridge the gap, too, e.g., inserting an iterative operator would yield the reading that Amélie left repeatedly during the two hours.

P onto the set of eventualities e in the extension of P that have an eventuality e' associated with them whose description includes the property of being an aftermath of e.
Inserting (30) directly above the argument in (29) results in another dominance rhombus:

(31)
$$\Box$$
$$\lambda e_{+\text{BD}}.\ \text{PH}(\mathbf{2h}, \lambda e'_{\text{-BD}}.\ \Box(e'))(e) \qquad \lambda e'_{+\text{BD}}.\mathbf{leave'}\ (\mathbf{amélie'})(e') \wedge \exists e''.\ \Box\ (e'')$$
$$\lambda e''_{+\text{ST}}.\ \text{RES}(e'', e')$$

There is only one well-formed solution of (31), viz., the one in which the functor is inserted between the two parts of the argument. The resulting λ-termis (32), which models the desired reading of (4) that the aftermath of Amélie's leaving (her absence), but not the leaving itself lasted two hours.

(32) $\lambda e'_{+\text{BD}}.\mathbf{leave'}(\mathbf{amélie'})(e') \wedge \exists e''_{+\text{BD}}.\text{PH}(\mathbf{2h}, \lambda e_{+\text{ST}}.\text{RES}(e, e'))(e'')$

This representation of LSC highlights the common ground between LSC and agentive modification in that it represents both phenomena in terms of the same kind of semantic structure. What is more, it can be extended to related phenomena like the repetitive-restitutive ambiguity of *again*-sentences (see section 5.2). This generalizability justifies the complexity of the approach, even if competing approaches are simpler. E.g., Dölling (1998) gives an account of (31) and similar LSC cases that is less complex, it uses only underspecified variables in λ-terms. But this account is not generalizable in the same way. The structural similarity between (31) and (10) proves the claim that a comprehensive account of reinterpretation phenomena must use a semantic description formalism that is at least as complex as a formalism that can handle scope underspecification.
This account of LSC provides a handle on the issue of control. Constituents that consist of a modifier/subcategorizer and its modified/subcategorized element are R-sites for LSC only if their semantics (a functor-argument pair) meets two conditions. First, a hole in the functor constraint must dominate an embedded fragment in the argument constraint (the value of CONT|EMCON) but the argument constraint fails to provide such a fragment. Second, the argument must allow augmentation by an operator that provides such a fragment (and thereby determines the value of CONT|EMCON). If one of the two conditions fails, this is already sufficient to block LSC. This is illustrated in (24) and (25).
(24) has no LSC interpretation, since the modifier *yesterday* ignores the the value of CONT|EMCON of its modified expression. It inserts the constraint of its modified expression (the argument) as a *whole* into its constraint (the functor), as depicted in (14b).
In (25), the modified expression blocks LSC although the result of semantic construction for (25) has the same structure as the one of (4). Functor and argument are dominance daughters in this constraint, since again the modified expression fails to specify the value of CONT|EMCON, while the modifier makes reference to this value.

(33)
$$\Box$$
$$\lambda e_{+\text{BD}}.\ \text{PH}(\mathbf{2h}, \lambda e'_{\text{-BD}}.\ \Box(e'))(e) \qquad \lambda e'_{\text{-BD}}.\mathbf{laugh'}\ (\mathbf{amélie'})(e')$$

But the argument in (33) (the semantic contribution of the modified expression) is a constraint for an unbounded predicate (due to the semantics of *laugh*). This unboundedness prevents inserting the operator (30) directly above the argument. Thus, the verb lexically determines that its projections disallow LSC like in (4). This ties in with the intuition that unbounded predicates do not involve the notion of aftermath.[13]

The lexical control of LSC in the proposed approach is more fine-grained than a determination of R-sites for LSC in the syntax-semantics interface. This fine-grainedness is needed for constituents that consist of an adverbial and a (modified) predicate. Some adverbials (e.g., *yesterday*) lexically block LSC for these constituents, others (e.g., durative adverbials) license it, but there is no corresponding syntactic distinction of adverbials. I.e., information beyond the syntactic category is needed to identify R-sites for LSC.

In sum, the lexical introduction and control of LSC in the proposed approach provide an efficient means of avoiding overgeneration for reinterpretation analyses.

LSC is not restricted to aspectual semantics but reappears in the nominal domain, e.g., in *fast road*. Here *fast* (as opposed to e.g. *red*) refers to the value of CONT|EMCON, but the noun does not specify it. The resulting constraint for the semantics of *fast road* is (34):

(34)
$$\square \qquad \lambda x_{\text{-STAT}}.\mathbf{fast}'(x) \wedge \square(x) \qquad \lambda y_{+\text{STAT} \sqcap \text{ARTE}}.\mathbf{road}'(y)$$

The gaps in (34) cannot be closed, since there is once again a sortal conflict between functor (the adjective semantics) and argument (the noun semantics). The functor selects for properties of entities with (im-)proper parts that may undergo some change (typically, a change of location), but the argument is a property of entities that have no such parts. This incompatibility is modelled in terms of the sort ±STAT[ionary]. (For eventualities, the distinction coincides with ±ST, i.e., $+\text{ST} \sqsubseteq +\text{STAT}$.) The sort ARTE stands for 'artefact'.

The constraint (34) can be augmented along the lines expounded above for (4). The augmentation is given in (35) and introduces the notion of purpose. The property of being a purpose for a given entity is not yet restricted with respect to the sorts ±STAT. The operator (35) is restricted to artefacts, as indicated by the sort on the λ-abstracted variable. (35) is provided by world knowledge: artefacts typically are created to serve some purpose.

(35)
$$\lambda x_{\text{ARTE}}.\square(x) \wedge \exists e.\ \square\ (e) \qquad \lambda e.\ \mathbf{purpose}'(e,x)$$

Combining the two constraints yields the constraint (36):

(36)
$$\square \qquad \lambda x_{\text{-STAT}}.\ \mathbf{fast}'(x) \wedge \square(x) \qquad \lambda x_{+\text{STAT} \sqcap \text{ARTE}}.\mathbf{road}'(x) \wedge \exists e.\square\ (e) \qquad \lambda e.\ \mathbf{purpose}'(e,x)$$

[13] Rather than by LSC, (33) is resolved by letting the functor directly dominate the argument. This yields a λ-term for the set of eventualities that last two hours and are spatiotemporally coextensive to an eventuality of Amélie laughing.

Here the structural ambiguity is directly resolved by the sortal information. The functor can only dominate the dominated fragment of the argument. This at the same time rules out stative purposes. The well-formed solution (37) of (36) is the λ-term for the set of roads whose purpose is nonstative and fast:

(37) $\lambda x_{+\text{STAT} \sqcap \text{ ARTE}}.\mathbf{road'}(x) \wedge \exists e_{-\text{ST}}.\mathbf{fast'}(e) \wedge \mathbf{purpose'}(e, x)$

This analysis deliberately leaves open the exact nature of the purpose of the road (in contrast to Pustejovsky, who assumes that this purpose is invariably driving on the road). It takes into account that roads may serve a different purpose in different contexts.

The issue of lexical control is treated like for the aspectual LSC cases. Nominal modifiers specify their liability to engage in LSC by relating their semantics to the semantics of their arguments via the value of CONT|EMCON (and not the CONT|EMCON value). E.g., *red* belongs to the former group of modifiers, which blocks LSC for expressions like *red car*.

The semantics of nouns indicates whether the purpose LSC (35) is possible or not: this kind of LSC presupposes that the main λ-abstracted argument of the noun semantics has a sort that is compatible with ARTE. Hence there is no such reinterpretation for *fast comet* and the like, as desired.

This concludes the presentation of the synchronic aspects of reinterpretation. This section has shown how reinterpretation patterns can be accounted for in a flexible semantic analysis that is designed to interact with additional knowledge sources in the full determination of the meaning of a reinterpretation expression.

5 Reinterpretation and language change

This section is devoted to the diachronic aspects of reinterpretation. It sketches a theory of the way in which reinterpretation may cause linguistic change. The proposed approach allows an exact and formal spellout of this theory.

Two stages are distinguished for linguistic changes that are triggered by reinterpretation: first, *reanalysis*, which integrates reinterpretation operators into lexical semantics, and, second, *restructuring*, which rearranges the result of reanalysis.

5.1 Reanalysis

The first stage of linguistic change caused by reinterpretation consists of a *reanalysis* of reinterpretation expressions. That is, frequent use of a reinterpretation pattern for a certain expression may become conventionalized in that the boundary between the semantic and the non-semantic contributions to the meaning of the expression is being shifted: parts of the non-semantic contributions are no longer felt to stem from sources outside semantic construction, but are reclassified as a part of semantic construction. Eventually, they are attributed to one (or several) of the subexpressions that make up the reinterpretation expression, which changes the meaning of these subexpressions.

The German verb *schmelzen* 'melt' illustrates this process. At a first glance, it is not completely clear whether it denotes a property of masses only, or a property of masses or things. This has far-reaching consequences for the interpretation of examples like (38):

(38) Ein Würfel schmilzt 'A cube is melting'

If one follows Dölling in claiming that *schmelzen* is a property of masses, (38) is an instance of type coercion. The semantic analysis of (38) would have to be (39), which follows the pattern of (3). The left hand side of (39) would be the result of semantic construction, with the right hand side being the extrasemantically derived operator that mediates between the conflicting sortal requirements of the subject NP semantics and the VP semantics:

(39) $$\exists x_{\mathrm{THING}}.\mathbf{cube'}(x) \wedge \Box\,(x) \qquad \lambda x_{\mathrm{MASS}}.\,\mathbf{melt'}(x) \quad \Leftarrow \quad \lambda z_{\mathrm{CONC}} \exists y_{\mathrm{MASS}}.\mathbf{const'}(y,z) \wedge \Box\,(y)$$

The core of the reinterpretation operator in (39) is the function **const′**. This function is Link's (1983) function h that maps things onto masses and is the identity mapping for masses. The sortal specification CONC[rete] for the first argument of **const′** is to indicate that this argument is restricted to concrete entities of all sorts. Integrating both sides in (39) would yield the λ-term (40), which models the interpretation of (38) 'the substance of the cube is melting':

(40) $$\exists x_{\mathrm{THING}}.\mathbf{cube'}(x) \wedge \exists y_{\mathrm{MASS}}.\mathbf{const'}(y,x) \wedge \mathbf{melt'}(y)$$

However, this analysis does not mesh with intuitions about (38). Intuitively, *schmelzen* can be a property of either masses or things, hence, (38) involves no reinterpretation. One can unify both intuitions in terms of an ongoing linguistic change that moves *schmelzen* from a predicate for masses only to a predicate for masses or things. This change is a case of reanalysis: the original derivation of the meaning of (38) is (39), but reanalysis attributes the extralinguistic operator to the semantics of (38), in particular, to the verb meaning. The result of this change is a new semantics for (38), viz., (41). This constraint comprises an embedded fragment for the property of concrete entities that the mass that constitutes or equals them is melting. (41) has as its constructive solution the λ-term (40):

(41) $$\exists x_{\mathrm{THING}}.\mathbf{cube'}(x) \wedge \Box\,(x) \qquad \lambda z_{\mathrm{CONC}} \exists y_{\mathrm{MASS}}.\,\mathbf{const'}(y,z) \wedge \mathbf{melt'}(y)$$

This new semantic representation is a constraint that contains no parts that must be further determined with respect to the context. Therefore, reinterpretation-triggered linguistic change is restricted to reanalysis for cases like *schmelzen*. But if reanalysis returns a new semantic representation with parts that need further determination, there is a second feasible step of reinterpretation-triggered linguistic change, which is called *restructuring*. This process is the topic of the following subsection.

5.2 Restructuring

Restructuring is triggered by specific semantic representations that are the output of reanalysis. The first characteristic of these representations is that reanalysis has introduced into them LT fragments that need further determination. Examples are fragments like $\lambda e.\text{RES}(e, e')$, the property of being an aftermath of an eventuality e', or $\lambda e.\mathbf{purpose}'(e, x)$, the property of being a purpose of an artefact x. The second characteristic is that this further determination is already provided by other fragments of the constraint, which results in a rearranging of the material of the constraint.

As an example, consider the case of (4). The conventionalization of such examples suggests that reanalysis is taking place, which interprets the operator (30) as part of the semantic structure of the bounded predicate *leave*[14]. The semantics of *leave* after reanalysis is (42):

(42)
$$\begin{array}{l} \lambda x \lambda e_{+\text{BD}}.\mathbf{leave}'(x)(e) \wedge \exists e''.\ \square\ (e'') \\ \qquad\qquad\qquad \lambda e''_{\text{-BD}}.\ \vdots\text{RES}(e'', e) \end{array}$$

In prose: (42) is a constraint for λ-terms that introduce pairs of individuals x and eventualities e such that x is leaving at e and there is a second eventuality e'', which is characterized by a property that is not yet fully determined, all one knows is that the property of being an aftermath of e is part of this property.

This property of being an aftermath of e calls for further determination. But this determination is already provided within the constraint: the verb *leave* itself describes the aftermath of eventualities in its extension lexically, roughly, as the property of being away. In a decomposition analysis, this property would be featured in a semantic representation as the argument of the operator BECOME. '$\lambda x \lambda e.\mathbf{leave}'(x)(e)$' is spelt out as '$\lambda x \lambda e.\text{BECOME}(\lambda e'.\mathbf{be\text{-}away}'(x)(e'))(e)$'. That is, (42) is an abbreviation of (43):[15]

(43)
$$\begin{array}{l} \lambda x \lambda e_{+\text{BD}}\text{BECOME}(\lambda e'_{\text{-BD}}.\mathbf{be\text{-}away}'(x)(e'))(e) \wedge \exists e''.\ \square\ (e'') \\ \qquad\qquad\qquad\qquad\qquad\qquad \lambda e''_{\text{-BD}}.\ \vdots\text{RES}(e'', e) \end{array}$$

Hence, restructuring can move this information over to the dominated fragment, which returns the new, simpler semantic representation (44) for *leave*. The existential quantification can also be omitted, as it follows from the semantics of BECOME:

(44)
$$\begin{array}{l} \lambda x \lambda e_{+\text{BD}}.\text{BECOME}(\ \square\ )(e) \\ \qquad\qquad \lambda e'_{\text{-BD}}.\ \vdots\mathbf{be-away}'(x)(e') \end{array}$$

[14]Note that some bounded predicates (e.g., *cough* or *play the Moonshine Sonata*) disallow this aftermath reinterpretation, since they do not introduce a change of state. This fact, which is neglected in the present paper, must be modelled in terms of a subclassification of the group of bounded predicates along the lines proposed in Egg (1995), where this group of predicates is called *intergressive predicates*.

[15]Analyzing the semantics of *leave* as a change of state from being here to not being here is simplified: it omits the question of whether it must introduce the notion of *movement*. BECOME(φ) is defined as the property of being the smallest eventuality in between two eventualities for which $\neg\varphi$ and φ hold.

In sum, the hypothesis is that this final step of reinterpretation-driven linguistic change is at present turning the semantics of *leave* (and of other similar predicates) into an expression that allows the compositional derivation of the meaning of (4) along the lines expounded for the modification of agentive nouns. This closely models our intuition that the reinterpretation in cases like (4) is becoming more and more conventionalized.

The restructuring analysis (44) is also very useful for the ambiguity of *again*-sentences. *Again* presupposes the existence of (preceding) additional eventualities, which are either characterized by the semantics of the modified expression as a whole (the 'repetitive' reading) or, if this expression lexically specifies an aftermath, by the predicate for this aftermath (the 'restitutive' reading). E.g., in (45) the repetitive reading of (45) presupposes another leaving of Amélie, the restitutive one, only a preceding absence of Amélie:

(45) Amélie left again

Assuming that the presuppositions are determined compositionally by different semantic scoping possibilities for *again*, this ambiguity can be modelled straightforwardly with the help of a semantic representation of *leave* like (44), which allows the semantics of *again* to take scope over the predicate for the aftermath only.

This section advocated a theory on reinterpretation-triggered linguistic change and provided an exact spellout of its assumptions. This change may consist of one step (*reanalysis*) only, in which reinterpretation operators become part of the compositional semantics, or involve the second step of *restructuring*. Reanalysis may introduce into the semantic representations of lexemes new elements that need further determination. If other parts of the same representation provide this determination, this causes restructuring of the material of the semantic representation.

6 Conclusion and further work

This paper presented an analysis of reinterpretation for the representation of its synchronic and diachronic aspects. The representation is couched in the framework of underspecification, which makes possible a tractable implementation of the analysis. This is being pursued in the project CHORUS in the SFB ('Special Research Division') 378 in Saarbrücken.

The presented work is part of a larger current of research on reinterpretation. It focuses on those parts of the semantics of reinterpretation expressions that are fixed by semantic construction. Hence, it must be complemented by an account of how the other, non-semantic parts are to be described and integrated with the contribution of semantic composition in the determination of a fully specified meaning of a reinterpretation expression.

It is an advantage of the proposed analysis that it is flexible enough to be integrated into a variety of analyses for these other, non-semantic parts. It might serve as the non-defeasible part of a default approach to reinterpretation phenomena, or provide one of the knowledge sources that are adduced by an abductive system along the lines of Hobbs et al. (1993) in order to determine the meaning of reinterpretation expressions.

But the presented approach also raises questions on the limits of the proposed strategy. Modelling reinterpretation by deliberately relaxing semantic constraints is just one phenomenon that can be modelled by relaxation. Many more ambiguities (e.g., those that arise from ellipsis reconstruction, see Egg et al. 1998) are modelled in the very same fashion. The combination of these relaxations might lead to unwanted interactions, in particular, it might give rise to unintended readings. Koller et al. (1999) discuss this problem. They identify positions in constraint structures where further relaxation is possible without running the risk of unwanted ambiguities in terms of a general *safety criterion*. They show that the kind of semantic representations for type coercion expressions that was presented in this paper does not introduce such unwanted ambiguities.

References

Alshawi, H. (ed.) (1992). *The Core Language Engine*. Cambridge: MIT Press.

Dölling, J. (1992). Flexible Interpretation durch Sortenverschiebung. In I. Zimmermann und A. Strigin (eds), *Fügungspotenzen*, 23–62. Berlin: Akademie-Verlag.

Dölling, J. (1995). Ontological domains, semantic sorts and systematic ambiguity. *International Journal of Human-Computer Studies 43*, 785–807.

Dölling, J. (1998). Modifikation von Resultatszuständen und lexikalisch-semantischen Repräsentationen. In P. Ludewig und B. Geurts (eds), *Lexikalische Semantik aus kognitiver Sicht*, 173–206. Narr.

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

Egg, M. (1995). The intergressive as a new category of verbal aktionsart. *Journal of Semantics 12*, 311–356.

Egg, M., J. Niehren, P. Ruhrberg, und F. Xu (1998). Constraints over lambda-structures in semantic underspecification. In *Proceedings of ACL/COLING '98*, Montreal, Canada.

Herweg, M. (1991). Temporale Konjunktionen und Aspekt. *Kognitionswissenschaft 2*, 51–90.

Hobbs, J. und S. Shieber (1987). An algorithm for generating quantifier scoping. *Computational Linguistics 13*, 47–63.

Hobbs, J., M. Stickel, D. Appelt, und P. Martin (1993). Interpretation as abduction. *Artificial Intelligence 63*, 69–142.

Koller, A., J. Niehren, und K. Striegnitz (1999). Relaxing underspecified semantic representations for reinterpretation. In *Proceedings of MOL 6*, Orlando.

Krifka, M., F. Pelletier, G. Carlson, A. ter Meulen, G. Chierchia, und G. Link (1995). Genericity: an introduction. In G. Carlson und F. Pelletier (eds), *The generic book*, 1–124. University of Chicago Press.

Link, G. (1983). The logical analysis of plurals and mass terms: a lattice-theoretic approach. In R. Bäuerle, C. Schwarze, und v. Stechow. A. (eds), *Meaning, use, and the interpretation of language*, 303–323. Berlin: de Gruyter.

Moens, M. und M. Steedman (1986). Temporal information and natural language processing. Technical Report Research Paper RP-2, CCS, University of Edinburgh.

Moens, M. und M. Steedman (1988). Temporal ontology and temporal reference. *Computational Linguistics 14*, 3–14.

Nunberg, G. (1993). Indexicality and deixis. *Linguistics & Philosophy 16*, 1–43.

Pinkal, M. (1996). Radical underspecification. In P. Dekker und M. Stokhof (eds), *Proceedings of the 10th Amsterdam Colloquium*, Amsterdam, 587–606. ILLC.

Pulman, S. (1997). Aspectual shift as type coercion. *Transactions of the Philological Society 95*.

Pustejovsky, J. (1995). *The generative lexicon*. Cambridge: MIT Press.

Reyle, U. (1993). Dealing with ambiguities by underspecification: construction, representation, and deduction. *Journal of Semantics 10*, 123–179.

Rooth, M. und B. Partee (1982). Conjunction type ambiguity and wide scope "or". In *Proceedings of WCCFL 1*, 353–362. Stanford University.