Res Lang Comput (2007) 5:435–455
DOI 10.1007/s11168-007-9039-x

Against Opacity

Markus Egg

Published online: 12 October 2007

Abstract Compositionality often presupposes a notion of *opacity* in that the combination of the meanings M and N of the subconstituents of a constituent C into the meaning of C must be blind to the inner structure of M and N. However, for cases where there is no direct 1-1 mapping between (surface) syntactic and semantic structure, opacity seems to be too strong a restriction on semantic construction. One could resolve this dilemma by basing semantic construction on another, not directly visible level of syntactic structure. But this strategy is not adopted by advocates of a surface-oriented syntactic structure such as the one in HPSG (Pollard and Sag, 1994, *Head-driven Phrase Structure Grammar*. CSLI and University of Chicago Press). I will argue that the more relaxed notion of compositionality advocated in Egg (2004, Mismatches at the Syntax-semantics interface. In S. Müller (Ed.), *Proceedings of the 11th International Conference on Head-Driven Phrase Structure Grammar* (pp. 119–139). Stanford: CSLI Publications), which relinquishes opacity, allows semantic construction from the syntactic surface even if there is no direct 1-1 mapping between the two. As it allows a constituent to refer to only a *part* of its syntactic sister constituent, it extends straightforwardly to cases like *John's former car* (Larson and Cho, Nat Lang Semant 11:217–247, 2003), where the semantic contributions of its syntactic subconstituents are *intertwined* in the meaning of the whole expression in that this meaning looks schematically like $m_1(n_1(m_2(n_2)))$, where M breaks down into m_1 and m_2, and N into n_1 and n_2, respectively.

Keywords Syntax-semantics interface · Opacity · Underspecification formalisms · Constraints · Syntax-semantics mismatches · Structural ambiguity · Scope

M. Egg (✉)
Center for Language and Cognition Groningen, Rijksuniversiteit Groningen, Groningen, The Netherlands
e-mail: egg@let.rug.nl

1 Introduction

Although the so-called Frege principle of compositionality merely states that the meaning of a whole is determined exclusively by the meanings of its parts plus the way in which these are put together, the principle is very often implemented in a much more restricted way: The semantic contribution of a constituent is the result of functional application of the meaning of one subconstituent (the *functor*) to the meaning of the other one (the *argument*). The inner structure of the meaning of either subconstituent is *opaque* to this application. Constituents contributing functors ('semantic heads') are distinguished syntactically as syntactic heads, except in adjunction structures, here the adjunct provides the functor.

For many expressions, however, this strict 1-1 relation between syntax and semantics cannot be postulated on the basis of their visible syntactic structure. I.e., there are mismatches between syntax and semantics; examples for such mismatches include a multitude of cases in which a constituent C_1 may pertain semantically to only *part* of its syntactic sister C_2.

E.g., in the preferred interpretation of (1) as 'person who usually dances beautifully', the adjective pertains to the verb stem only.

(1) beautiful dancer

To see this, consider first (2), which breaks down the semantics of *dancer* into the stem and the affix meaning. In such agentive nouns, the stem semantics is the argument of the affix semantics:

(2) $\underbrace{\text{'person who usually'}}_{\text{affix meaning}} \quad \ldots \quad \underbrace{\text{'dances'}}_{\text{stem meaning}}$

Second, assume that the adjective meaning pertains immediately to the *verb stem* meaning only, which returns the property of dancing beautifully. This entails that the adjective ends up in the scope of the affix *-er*. The preferred interpretation of (1) follows then directly from applying the affix meaning 'person who usually *X*-es' (where *X* is the meaning of the scope domain of the affix) to the meaning of the modified stem. The structure of the resulting semantic representation is depicted schematically by (3), where the 'arg_n' are argument parts:

(3) $\text{arg}_1(\text{functor}(\text{arg}_2))$

In addition, (1) has a reading 'beautiful person who usually dances'. Semantic construction for this reading is trivial, because the adjective pertains semantically to the modified noun as a whole in this reading.

In a similar fashion, the adverbial in (4) pertains only to that part of the verb semantics that specifies the *aftermath* of the change of state described by the verb. The resulting reading is 'go away and be absent for two hours'.

(4) leave for two hours

Formally, the aftermath-specifying part of the verb semantics can be formalised as the argument of a change-of-state operator like BECOME in a decomposition analysis of the verb semantics (Dowty 1979).

To bridge the gap between the (visible) syntactic structure and semantics for examples such as (1) and (4), generative approaches as Larson (1998) and von Stechow (1996) assign them a not directly visible but semantically relevant syntactic layer. The 1-1 relation is then postulated between this second syntactic layer and semantics. E.g., for von Stechow (1996), the relevant layer is the one of *Logical Form* (May 1985).

However, this strategy is not adopted by advocates of more surface-oriented syntactic analyses such as Head-Driven Phrase Structure Grammar (HPSG; Pollard and Sag 1994). To prove the generality and viability of such analyses it is therefore crucial to be able to account for these challenging examples within such a syntactic analysis.

Egg (2004, 2006) suggests interfaces for cases as (1) and (4) that allow semantic reference of a semantic head to only part of its syntactic sister constituent. I.e., semantic compositionality is relaxed in that the argument is no longer semantically opaque in every case.

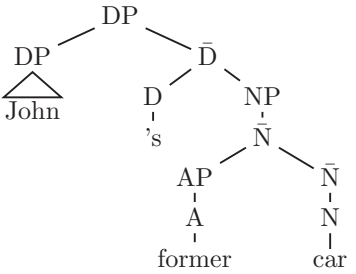
However, there are even more complex data, where opacity must be given up for *both* functor and argument, in particular, expressions like (5):

(5) John's former car

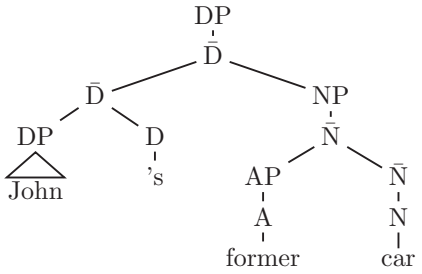
Larson and Cho (2003) point out that the possessive relation involved in the genitive may be in the scope of the scopal adjective *former* or not, which gives rise to ambiguity. The two resulting readings can be glossed as 'set of properties of the only *x* that used to be in the possession of John and a car' and 'set of properties of the only *x* in the possession of John that used to be a car'. In the following, we will focus on the first reading of (5).

In a surface-oriented syntactic analysis, (5) is assigned a structure like the ones in (6). Here and in the following, bar nodes in nonbranching tree parts have been omitted. The *'s* element of the so-called Anglo-Saxon genitive is analysed not as a nominal affix but as an (enclitic) constituent that attaches to whole DPs. These DPs (as *John* in *John's*) can either be regarded as specifiers (6a) or as complements of the *'s* element (6b):

(6a)



(6b)



But, regardless of which alternative in (6) is chosen, the syntax-semantics interface faces a severe problem in the construction of the semantic representation for the first reading of (5) from the chosen underlying syntactic structure: The semantic contribution of the *'s* element and its NP complement are *interleaved*, because this element contributes definiteness and the possessive relation to the meaning of the resulting $\bar{D}$ constituent. This constellation is depicted in schematically in (7a). Here 'Det_1' is the part of the determiner semantics that introduces the definiteness, 'Det_2', the one that introduces the possessive relation. Similarly, 'NP_1' refers to the part of the NP semantics that is contributed by the adjective *former*, 'NP_2', to the rest of the NP semantics (here, the meaning of *car*). (7b) tries to relate this schema to the gloss of the first reading of (5) given above:

(7) (a) $Det_1(NP_1(Det_2(NP_2)))$
(b) set of properties of the only x … that used to be … in the possession of (John) … and a car

I.e., neither the semantic head nor its sister constituent are semantically opaque anymore. In this paper, I will show that the approach to compositionality adopted in Egg (2004, 2006) for structures such as (3) can be extended to handle the more involved structure (7). This approach relinquishes opacity and thereby gains considerable expressivity for the formulation of syntax-semantics interfaces.

The paper is structured as follows. Section 2 discusses interface strategies to cope with problematic cases like (1) and (5), which includes a review of previous treatments of these examples in Larson and Cho (2003) and Egg (2004, 2006). Then I will apply a revised version of the latter approach to examples like (5) in Sects. 3–5 and show how *semantic underspecification* can be used for a suitable interface that allows semantic construction for these examples on the basis of a surface-oriented syntactic analysis. The paper closes with an outlook on a further extension of the analysis.

2 Interface Strategies

In this section, I will first review the interface strategies of Larson and Cho (2003) and Egg (2004, 2006) and their application to (5). I will show that this example is analysed as a case of *structural ambiguity* in either strategy, but that these ambiguities are handled in a different fashion. These differences follow from a different notion of compositionality, which will be discussed in a subsequent subsection.

Before the strategies are discussed in detail, consider the formalisation of the intuitions on the semantics of (5) in (8)[1]:

(8) (a) $\lambda P \exists!x.[\textbf{former}'(^{\wedge}(\textbf{car}'(x) \wedge \text{POSS}(\textbf{john}', x)))] \wedge P(x)$
(b) $\lambda P \exists!x.[\textbf{former}'(^{\wedge}(\textbf{car}'(x))) \wedge \text{POSS}(\textbf{john}', x)] \wedge P(x)$

(8a) represents the meaning of the interesting reading of (5), viz., the set of properties of the unique entity with the following property: It used to be a car in the possession of John. The gloss for (8b) is analogously 'set of properties of the unique entity with the following property: It used to be a car, and is in the possession of John'.

Three comments are in order on these representations. First, the *'s* genitive introduces definiteness just like a definite determiner, thus, its semantic contribution includes a (generalised) quantifier whose restriction is fulfilled by one entity only (Barwise and Cooper 1981). I will model this quantifier by the Russellian (9a) (ignoring the subsequent debate about definiteness), which will be abbreviated by (9b) to keep the representations readable:

(9) (a) $\lambda Q \lambda P \exists x.(\forall y.Q(y) \rightarrow y = x) \wedge P(x)$
(b) $\lambda Q \lambda P \exists!x.[Q(x)] \wedge P(x)$

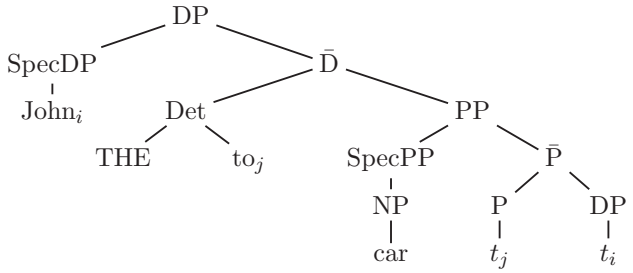
In addition, 'POSS' glosses the possession relation, and **former**′ is a relation between the intension of a predicate and an individual x that holds at a given world-time pair $\langle w, t \rangle$ iff the extension of the predicate at $\langle w, t \rangle$ does not hold for x but its extension at $\langle w, t' \rangle$ (for a t' that precedes t) holds for x.

2.1 Larson and Cho's Strategy

Larson and Cho (2003) explain the ambiguity of (5) by postulating an underlying syntactic structure of its own for every reading. These underlying structures correspond to the same visible syntactic surface structure, whence the ambiguity.

They assume the following structure for *John's car*:

(10)



The syntactic structure of the genitive is described as a PP ('*car to John*') whose head is the possession-indicating *to*. This PP is the complement of an abstract definite determiner THE. The complement of the preposition moves to SpecDP and the

[1] The restriction of the quantifier is indicated by brackets.

preposition incorporates into the determiner. The resulting Det element spells out as '*s*' and cliticizes to the DP *John*.

The ambiguity in (5) can then directly be put down to different possibilities of adjoining the adjective (phrase): It can be adjoined to *car* only (which functions as specifier of the PP) or to the whole PP that comprises the possessive relation. Since the adjunction site determines the scope of the modifier, the scope of *former* may or may not extend over the possessive relation. I.e., the structural ambiguity is the direct consequence of different syntactic structures. Since either structure corresponds to the same surface structure (the AP precedes its adjunction site, which happens to be between *to* and *car* in either case), this surface structure cannot distinguish between the underlying structures.

There is, however, a problem for this analysis of examples like (5): It determines the scope between the *'s* element and its specifier, which plays no role for (5) itself but emerges as a problem as soon as we replace the specifier *John* by a clearly scope-bearing DP, e.g., *every man*.

(11) every man's car was red

This sentence has two readings; the preferred one makes a statement about every man (his car was red), the other one, a statement about a specific car (the one belonging to every man), viz., that it was red. Thus, the DP argument of the preposition in a structure like (10) must also be able to move to another position *inside* the $\bar{\text{D}}$ constituent in (10) to receive scope below Det. It is not immediately obvious what this position could be, but, what is more, such an analysis would no longer be compatible with an analysis of the Anglo-Saxon genitive in terms of cliticisation, because the DP *John* would have to appear to the right of the *'s* element.

Taking into account scope ambiguity and in particular this second reading of (11) adds to the challenge posed by expressions like (5): If we combine these two sources of ambiguity in expressions such as (12), readings multiply:

(12) every man's former car is red

The two ambiguities are independent of each other; there is scope interaction between *former* and the possessive relation on the one hand and the *'s* element and its specifier on the other hand, but these ambiguities do not interact. The readings of (12) can be rendered by (13):

(13) (a) $\exists!x.[\forall z.\mathbf{man}'(z) \rightarrow \mathbf{former}'(\hat{}(\mathbf{car}'(x))) \wedge \text{POSS}(z,x)] \wedge \mathbf{red}'(x)$
(b) $\exists!x.[\forall z.\mathbf{man}'(z) \rightarrow \mathbf{former}'(\hat{}(\mathbf{car}'(x) \wedge \text{POSS}(z,x)))] \wedge \mathbf{red}'(x)$
(c) $\forall z.\mathbf{man}'(z) \rightarrow \exists!x.[\mathbf{former}'(\hat{}(\mathbf{car}'(x) \wedge \text{POSS}(z,x)))] \wedge \mathbf{red}'(x)$
(d) $\forall z.\mathbf{man}'(z) \rightarrow \exists!x.[\mathbf{former}'(\hat{}(\mathbf{car}'(x))) \wedge \text{POSS}(z,x)] \wedge \mathbf{red}'(x)$

The first two express that a specific entity is red, viz., the unique entity in the possession of every man that used to be a car (13a), and the unique car that used to be in the possession of every man (13b), respectively. The latter two express statements about every man: The unique entity that used to be a car in his possession is red (13c), and the unique entity in his possession that used to be a car is red (13d), respectively. Any approach to semantic construction will have to be able to model the fact that (12) has exactly these four readings.

2.2 Egg's Strategy

In contrast to the strategy of Larson and Cho (2003), the strategy of Egg (2004, 2006) is to derive one single semantic representation from one single (surface-oriented) syntactic structure. This type of approach was originally developed for quantifier scope ambiguities like the one in (14) (see among others Schubert and Pelletier 1982; Hobbs and Shieber 1987; Reyle 1993; Richter and Sailer 2004; Copestake et al. 2005):

(14) Every man loves a woman

To capture cases like (1) and (5)–and (12)—in such an approach, reference to *parts* of syntactic constituents is needed during semantic construction, which relinquishes opacity, but allows very powerful and flexible rules of semantic construction. Note, however, that no complexity is added, since such a strategy is needed anyway to handle scope ambiguity in an underspecified fashion.

For example (1), this means that the modifier may pertain semantically only to a part of its syntactic sister *dancer*, viz., the verb stem *dance*. In this case, the modifier has immediate scope over the verb stem only.

Let me make these ideas a bit more precise. If we neglect issues of argument binding for the purposes of this paper, the semantics of the agentive affix *-er* can (roughly) be defined as in (15a) as a function from the verb semantics P to the set of individuals that are identical to an individual x such that when x participates in an eventuality e (this is expressed by the relation **in**), then e is usually a P-eventuality where x is the agent.[2] Here '$\vec{y}$' is shorthand for a sequence of zero or more individual arguments of the verb.

The definition (15b) of the generic quantifier GEN is (one version of) the quantifier as discussed in Krifka et al. (1995)[3]:

(15) (a) $\lambda P \lambda z.\mathrm{GEN}[e, x](x \ \mathbf{in}\ e \wedge z = x, \exists \vec{y}.P(x, \vec{y})(e))$
(b) $\mathrm{GEN}[e, x](R(x)(e), C(x)(e))$ iff $R(x)(e)$ usually entails $C(x)(e)$

The meaning of *dancer* is then (16a), the set of people such that when they are participating in a (contextually relevant) eventuality, it is usually an eventuality of them dancing. Here the semantic contribution of the verb stem is underlined. If we now pertain the semantics of the adjective to only this underlined part, we obtain the representation (16b) for the preferred reading of (1). Here the adjective semantics is in the scope of GEN, thus, the expression refers to people who are usually dancing beautifully. Its other reading is represented by (16c), which refers to beautiful people who are usually dancing. Here the modifier pertains to *dancer* as a whole:

(16) (a) $\lambda y.\mathrm{GEN}[e, x](x \ \mathbf{in}\ e \wedge y = x, \underline{\mathbf{dance}'(x)(e)})$
(b) $\lambda y.\mathrm{GEN}[e, x](x \ \mathbf{in}\ e \wedge y = x, \underline{\mathbf{dance}'(x)(e)} \wedge \mathbf{beautiful}'(e))$
(c) $\lambda y.\mathrm{GEN}[e, x](x \ \mathbf{in}\ e \wedge y = x, \mathbf{dance}'(x)(e)) \wedge \mathbf{beautiful}'(y)$

[2] The term 'eventuality' refers to states of affairs of all kinds; following Davidson (1967), verbs and their projections have an additional eventuality argument in their semantics.

[3] For the argumentation in this paper, nothing hinges on the formalisation of the affix semantics as long as it contains a scope-bearing operator that extends over the modifier semantics in the non-trivial (but not the trivial) reading of examples such as (1).

However, this structural ambiguity is not based on syntax: Only one single syntactic structure is assumed, which is mapped onto an *underspecified* semantic structure that describes a whole *set* of semantic representations. Structural ambiguity arises whenever this set comprises several elements, e.g., the underspecified semantic structure for (1) describes (16b) and (16c). See Sect. 3 for details.

For (5), this means that semantic construction allows both the *'s* element and its NP complement to pertain to only a part of their syntactic sister. As a consequence, there is a unique result of semantic construction, but one which leaves open whether *former* has scope over the possessive relation or not. This deliberate underspecification models the structural ambiguity which is the reason for the fact that there are two readings for (5). See Sect. 5 for a step-by-step account of the semantic construction of (12), which extends (5) with the additional problem of the scope of the specifying DP.

2.3 Compositionality for Cases of Structural Ambiguity

The proposed approach to structural ambiguity raises the question of what its underlying notion of compositionality is. The basic problem is that there is no 1-1 relation between syntactic structures and semantic interpretations for cases of structural ambiguities like (5); its extension (12) also (re-)introduces the well-known phenomenon that the scope of quantifying expressions can be ambiguous. In all these cases, one syntactic structure seems to correspond to several semantic structures.

But this one-many correspondence would be in conflict with the *functional* nature of semantic interpretation, which associates one specific syntactic structure with only one single semantic structure (see Westerståhl 1998; Hodges 2001).

The problem has been tackled from several angles. To preserve a 1-1 relation between syntactic and semantic structure, approaches within Generative Syntax assume that each reading corresponds to a unique syntactic structure on a not directly visible but semantically relevant syntactic level. At this level, structural ambiguities are resolved. This approach underlies Larson and Cho's (2003) analysis of example (5) as sketched in Sect. 2.1.

Another way of multiplying syntactic entities in order to associate each reading with exactly one syntactic entity stems from Montague's (1974) account of quantifier scope ambiguity and is adopted in approaches like Hoeksema (1985): The 1-1 relation is postulated between syntactic *derivation trees* (a syntactic structure and its derivation history) and semantic structures rather than between syntactic and semantic structures. This approach is motivated by the definition of semantic interpretation as a homomorphism from the syntactic to the semantic algebra (every syntactic operation is translated into a semantic one).

In this approach, one single syntactic structure might be associated with several different interpretations as soon as a string of words can be assigned the same syntactic structure in different ways, since each syntactic operation is interpreted by a semantic composition rule of its own. For (3), e.g., this effect could be achieved by syntactic

operations that go beyond mere concatenation, e.g., Hoeksema's (1985) rule of *head adjunction*.[4]

The second approach to structural ambiguity is to introduce nondeterminism in the mapping from syntax to semantics explicitly (Cooper 1983): Roughly, quantifying DPs have—apart from their standard interpretation—another, structured interpretation, which sets aside their semantic contribution in a special storage mechanism (known as 'Cooper storage'). The semantic representation for any constituent C is thus structured into a tuple. The one element of the tuple is a (possibly empty) storage that lists the semantic contribution of quantifying NPs within C, its other element is the rest of the semantics of C. The storage list is then inherited up the syntactic tree; its elements can be combined with the main semantics of any constituent the NP is part of as long as this main semantics is of the type of a proposition.

The underspecification approach pursued in this paper contrasts with these two approaches in that it sticks to the 1-1 relation between syntactic and semantic structure without multiplying syntactic entities. It bridges the gap between syntax and semantics in the case of structural ambiguity by relating one syntactic structure to one *underspecified* semantic structure. This underspecified structure defines the set of readings R of the syntactic structure in that it denotes a set of (object-level) semantic representations (here, λ-terms), one for each element of R.

The contrast to approaches like Hoeksema (1985) lies in the assumption that each syntactic structure has a *unique* derivation history (ignoring trivial variation such as the question of the order in which sister constituents are derived). Thus, for each syntactic structure the semantic rules that are used for its interpretation and the order in which they contribute to this interpretation are fixed due to the homomorphism from the syntactic to the semantic algebra. In this way, a unique semantic representation is determined for the syntactic structure, which ensures a 1-1 relation between syntactic and semantic structure.

I will show that the underspecification approach builds the semantic analyses on top of comparatively surface-oriented syntactic structures, because one can define very flexible syntax–semantics interfaces with the help of underspecification formalisms; this will be expounded in the next sections.

3 The Semantic Representation Formalism

This section sketches the underlying semantic representation formalism *Constraint Language for Lambda Structures* (CLLS; Egg et al. 2001), on which the analyses in

[4] In one version of this rule, the expressions [abc] and x form together the expression [$abxc$], where b is the head of [abc]. The interpretation of [$abxc$] is functional application of the semantics of x to the semantics of [abc]. With such a rule, one could derive the non-trivial interpretation of (3) easily: One would first combine *beautiful* and *dance*, then the affix would be integrated by head adjunction. The resulting interpretation would give the affix wide scope over both adjective and noun. The trivial reading would be derived by combining verb and affix first.
This derivation, however, could not explain why the modifier *beautiful* emerges syntactically as a preceding adjective; a verbal constituent *beautiful dance* is strange because of its word order and the modification of a verb by an adjective. This seems to suggest an underlying structure in which the adjective modifies the noun as a whole in the syntax.

this paper are based. CLLS is an *underspecification formalism*, i.e., it captures structural ambiguities in *descriptions* or *constraints* that describe whole sets of semantic representations, one for each reading of a structurally ambiguous expression.

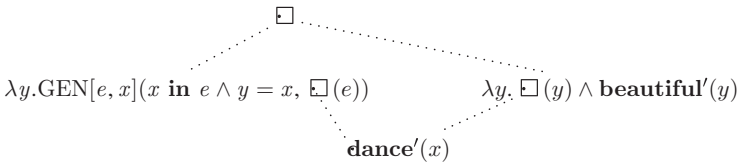
There are many other underspecification formalisms, e.g., Underspecified DRT (UDRT; Reyle 1993), and Minimal Recursion Semantics (MRS; Copestake et al. 2005). These formalisms permit an adequate representation of structural ambiguity and, what is more, they allow for definitions of flexible syntax-semantics (and morphology-semantics) interfaces.

In this paper, I will use an abbreviated form of CLLS, which facilitates reading considerably. In a second subsection I will sketch the stand of CLLS (and other underspecification formalisms) on the question of how to define compositionality taking into account the fact that the 1-1 relation between syntactic structures and meanings cannot be upheld for cases of structural ambiguities like (5).

CLLS expressions describe fully specified semantic representations, here, λ-terms. Those λ-terms that are described by or compatible with a constraint are called its *solutions*. In this paper, we are only concerned with solutions that comprise only material explicitly mentioned in the constraint. Then constraints emerge as a *partial order* on sets of fragments of semantic representations.

Consider for example the CLLS constraint (17) for the semantic representation of (1).

(17)



(17) illustrates the ingredients of simplified CLLS expressions:

– *fragments* of λ-terms
– not yet known parts of these fragments, indicated by '*holes*' (□)
– *dominance relations* (depicted by dotted lines) that relate fragments to holes

Dominance relations between a fragment and a hole express that the fragment is an (im-)proper part of what the hole stands for. These dominance relations model scope, and are therefore also used to model quantifier scope ambiguities. Structures like (17) are called *dominance diamonds*.[5]

The fact that the top fragment in (17) consists of a hole only indicates that the λ-term described by the constraint cannot yet be specified at all. However, the dominance relations between this hole and the fragments on the right (the meaning of the modifier) and the left (the semantic contribution of the affix *-er*) indicate that these fragments are the immediate parts of this λ-term. The verb stem semantics in the bot-

[5] I will sometimes abbreviate 'hole h in fragment f_1 dominates fragment f_2' by 'fragment f_1 dominates fragment f_2' when h can be identified from the context.

tom fragment is dominated by the right and left fragments, thus, ends up in the scope of both the adjective and the affix.[6]

To resolve the ambiguity in such constraints, we can add information monotonically, in particular, by strengthening dominance relations between holes and fragments to *identity*. In the case of (17), we have two choices: We can identify the affix fragment with the top hole, the modifier fragment with the hole in the affix fragment, and the verb stem fragment with the hole in the modifier fragment. This returns (16b). Alternatively, we can start this procedure with the modifier fragment, which eventually returns the λ-term (16c). There are no other solutions, thus, (17) is an adequate representation of the semantics of (1).

4 The Interface Rules

This section describes the syntax-semantics interface that serves for the derivation of constraints like (17) from a surface-oriented syntactic analysis.

This interface presumes that the constraint for the semantic contribution of every syntactic constituent distinguishes a *main* and an *embedded* fragment. Interface rules can then address both kinds of fragments, which makes possible a very fine-tuned semantic construction.

Lexemes may already distinguish two such fragments, see e.g. the lexical entry for the semantics of the agentive noun *dancer*:

(18)
$$\begin{array}{l} [\![\mathtt{N}]\!] : \lambda x.\mathrm{GEN}[e,y](y \ \mathbf{in} \ e \wedge y = x, \boxdot(e)) \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \vdots \\ [\![\mathtt{N_S}]\!] : \ \diamond\mathbf{dance}'(y) \end{array}$$

CLLS constraints like (18) indicate the main fragment of a constituent C by '$[\![\mathtt{C}]\!]$' and its secondary fragment, by '$[\![\mathtt{C_S}]\!]$'. '$[\![\mathtt{C}]\!]:\mathtt{F}$' means that the main fragment of C is defined as fragment F.

(18) indicates the basic interface strategy: If it is possible for a constituent C_1 to pertain semantically only to a part of the semantic contribution of a sister constituent C_2, then this part is already demarcated in the lexical entry of C_2 as a fragment of its own.

Lexical entries may also introduce dominance relations that relate the fragments. Some of these relations can be put down to variable binding: The fragment that comprises the binder must always dominate the one with the corresponding bound variable, e.g., in (18). They are nevertheless included in the constraints to facilitate reading.

Interface rules specify for each non-lexical constituent C how the constraints Con_1 and Con_2 of its immediate constituents, which are inherited by C, are combined into a new constraint for C. The rules combine Con_1 and Con_2 by addressing their main and secondary fragments and determine these features for C. For instance, the simple rule

[6] One aspect of the CLLS formalism cannot be fully captured in the abbreviated version presented here: Variable binding such as the binding of x in the bottom fragment of (17) by the generic quantifier in the left hand fragment cannot be broken by other binders in intervening fragments. See Egg et al. (2001) for this crucial aspect of CLLS.

that non-branching $\bar{\mathtt{X}}$ constituents inherit their fragments from their heads is written as (19):

(19) $[_{\bar{\mathtt{X}}}\ \mathtt{X}] \overset{\text{(SSI)}}{\Rightarrow} [\![\bar{\mathtt{X}}]\!] : [\![\mathtt{X}]\!];\quad [\![\bar{\mathtt{X}}_{\mathtt{S}}]\!] : [\![\mathtt{X}_{\mathtt{S}}]\!]$

The semantic representation of modification (adjunction) structures[7] is determined by the interface rule (20). The main fragment of the whole constituent ($[\![\bar{\mathtt{X}}_1]\!]$) is equal to $[\![\bar{\mathtt{X}}_2]\!]$, the one of the modified expression. But its secondary fragment $[\![\bar{\mathtt{X}}_{1\mathtt{S}}]\!]$ is not inherited from this expression, it consists of an application of the modifier fragment $[\![\mathtt{Mod}]\!]$ to a hole that dominates the secondary fragment $[\![\bar{\mathtt{X}}_{2\mathtt{S}}]\!]$ of the modified expression. The modifier fragments are equated ($[\![\mathtt{Mod}]\!]$: $[\![\mathtt{Mod}_{\mathtt{S}}]\!]$) to facilitate reading.

(20)

$$[_{\bar{\mathtt{X}}_1}\mathtt{Mod}\ \bar{\mathtt{X}}_2] \overset{\text{(SSI)}}{\Rightarrow} [\![\bar{\mathtt{X}}_1]\!]:[\![\bar{\mathtt{X}}_2]\!];\ [\![\bar{\mathtt{X}}_{1\mathtt{S}}]\!] : [\![\mathtt{Mod}]\!]\Big(\ \boxdot\ \Big);\ [\![\mathtt{Mod}]\!]: [\![\mathtt{Mod}_{\mathtt{S}}]\!]$$
$$\vdots$$
$$[\![\bar{\mathtt{X}}_{2\mathtt{S}}]\!]$$

For the semantic construction of $\bar{\mathtt{D}}$ constituents from the semantic contributions of determiner and NP complement, we must now distinguish two cases, viz., $\bar{\mathtt{D}}$ constituents that directly project to XP (without a specifier) and those that become the syntactic sister of an appropriate specifier. Distinguishing these two cases already on the $\bar{\mathtt{D}}$ level will avoid overgeneration. To this end, we exploit the fact that the former $\bar{\mathtt{D}}$ constituents have a determiner head whose semantic contribution is simple in that main and secondary fragment coincide. Only in the latter case do we find determiners with different main and secondary fragments in their semantic contribution.

For (eventually) specifierless $\bar{\mathtt{D}}$ constituents, the interface rule is (21), which is a reformulation of the rule introduced in Egg (2006) for the semantic construction for Turkish DPs. This rule covers the case where main and secondary fragment in the determiner semantics coincide. This condition is tested in the rule (by defining $[\![\mathtt{D}]\!]$ as $[\![\mathtt{D}_{\mathtt{S}}]\!]$).

(21)

$$[_{\bar{\mathtt{D}}}\ \mathtt{D}\ \mathtt{NP}] \overset{\text{(SSI)}}{\Rightarrow} [\![\mathtt{D}]\!]\ (\lambda x.\ \boxdot\ )\ \ (\lambda x.\ \boxdot);\qquad [\![\mathtt{D}]\!] : [\![\mathtt{D}_{\mathtt{S}}]\!];$$
$$\vdots \qquad\qquad \vdots$$
$$[\![\mathtt{NP}]\!]\ (x)\quad [\![\bar{\mathtt{D}}]\!] : x \qquad [\![\bar{\mathtt{D}}_{\mathtt{S}}]\!] : [\![\mathtt{NP}_{\mathtt{S}}]\!]$$

In prose: Apply the determiner semantics to two λ-abstracted holes. The first hole dominates the main NP fragment applied to the respective abstracted variable. This dominance relation reappears (22), where it is relevant for nested quantifiers as in (12), where in one reading the nested universal quantifier (from the specifying DP) may end up in the restriction of the quantifier introduced by the *'s* expression. The secondary fragment of the emerging $\bar{\mathtt{D}}$ constituent is inherited from the NP.

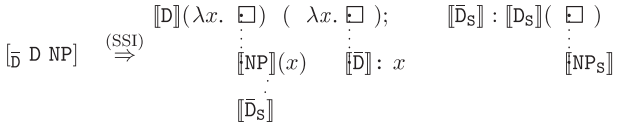
The second hole dominates only the bound variable. This variable constitutes the main fragment of the new $\bar{\mathtt{D}}$ constituent. This is the position where the semantics of the subcategorising element (e.g., a verb) will be integrated later on. The formal

[7] Syntactically, adjunction of XP to a constituent *C* means that XP and *C* are the daughters of a node in the syntax tree that is of the same category as *C*.

details are explicated in rule (25) below.[8] The secondary NP fragment constitutes the secondary fragment of the resulting $\bar{\text{D}}$ constituent.

The rule for determiners whose $\bar{\text{D}}$ projection will become the syntactic sister of a specifier is a bit more elaborate because it must also address the secondary fragment of the determiner. This fragment gets scope below the ⟦NP⟧ but above the $⟦\text{NP}_\text{S}⟧$ fragment.

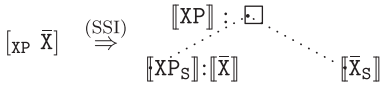
(22)



Rule (22) applies only to determiners whose semantics distinguishes main and secondary fragments. This follows from the fact that these fragments end up in different larger fragments, one of which dominates the other. This is compatible with determiners contributing different main and a secondary fragments, but would be incompatible with the identity of main and secondary fragment, which characterises determiners whose $\bar{\text{D}}$ projection projects directly to DP.

Finally, there are the rules for the semantic construction on the phrasal level. First, the interface links the syntax rule $\text{XP} \rightarrow \bar{\text{X}}$ to a semantic rule that introduces a hole as the main fragment of the XP. Since this hole dominates both fragments of the $\bar{\text{X}}$ constituent, the effect of this rule is often the construction of the upper half of a dominance diamond:

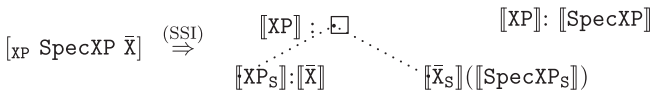
(23)



For example, the semantic representation of (1) in (17) can be constructed with the rules (19), (20), and (23) from the semantic contribution of *dancer* (18) and the standard semantic representation of *beautiful* ($\lambda P \lambda x . P(x) \wedge \textbf{beautiful}'(x)$; here the main and the secondary fragment are identical). See Egg (2004, 2006) for details.

The second rule for semantic construction at the XP level is more involved than (23) in that the semantics of the specifier must also be integrated into the resulting constraint. Since the main fragment of a DP indicates its scope, we can indicate the scope underspecification between the specifier and the specified $\bar{\text{D}}$ constituent by equating the main fragments of specifier and XP:

(24)



[8] Differences between this rule and the one presented in Egg (2006) are due to the fact that Egg (2006) does not go beyond the semantics of a simple DP, whereas the present paper extends this approach to issues of quantifier scope. This extension is needed to handle DPs comprising other DPs as specifiers. The extended version of the approach can also handle the crucial Turkish data discussed in Egg (2006).

Finally, rule (25) combines the semantics of a $\bar{\mathrm{V}}_2$ constituent with the one of a subcategorised DP in that the main fragments (for the scope domain of the DP) are equated. The secondary fragment of the emerging $\bar{\mathrm{V}}_1$ constituent is formed by applying the secondary fragment of $\bar{\mathrm{V}}_2$ to the one of the DP:

(25) $[_{\bar{\mathrm{V}}_1}\ \bar{\mathrm{V}}_2\ \mathrm{DP}] \overset{\text{(SSI)}}{\Rightarrow} [\![\bar{\mathrm{V}}_{1\mathrm{S}}]\!]: [\![\bar{\mathrm{V}}_{2\mathrm{S}}]\!]([\![\mathrm{DP}_{\mathrm{S}}]\!]); [\![\bar{\mathrm{V}}_1]\!]: [\![\mathrm{DP}]\!]; [\![\bar{\mathrm{V}}_1]\!]: [\![\bar{\mathrm{V}}_2]\!]$

If we let this rule abstract away from the ordering of $\bar{\mathrm{V}}_2$ and DP, it is applicable to subject DPs, too. In this case, $\bar{\mathrm{V}}_1 = \mathrm{S}$.

5 Semantic Construction of the Critical Example

This section describes the semantic construction of (12) on the basis of the syntactic analysis of the DP as shown in (6a). This means that the surface string *man's* breaks down into an enclitic determiner *'s* and the noun *man*.

The semantic representations for the constituents of (12) are (26a–f) for the adjectives, the nouns, and the two quantifiers. (26a–e) are simple in that main and secondary fragment coincide.

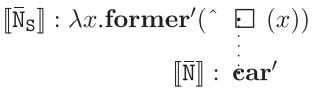
(26) (a) $[\![\mathrm{A}]\!], [\![\mathrm{A}_{\mathrm{S}}]\!]$: $\lambda P \lambda x.\mathbf{former}'(\hat{}P(x))$
(b) $[\![\mathrm{A}]\!], [\![\mathrm{A}_{\mathrm{S}}]\!]$: $\mathbf{red}'$
(c) $[\![\mathrm{N}]\!], [\![\mathrm{N}_{\mathrm{S}}]\!]$: $\mathbf{car}'$
(d) $[\![\mathrm{N}]\!], [\![\mathrm{N}_{\mathrm{S}}]\!]$: $\mathbf{man}'$
(e) $[\![\mathrm{D}]\!], [\![\mathrm{D}_{\mathrm{S}}]\!]$: $\lambda Q \lambda P \forall x.Q(x) \rightarrow P(x)$
(f) $[\![\mathrm{D}]\!]$: $\lambda Q \lambda P \exists!x.[Q(x)] \wedge P(x)$
$[\![\mathrm{D}_{\mathrm{S}}]\!]$: $\lambda P \lambda z \lambda y.P(y) \wedge \mathrm{POSS}(z, y)$

The semantic representation for the *'s* constituent in (26f) comprises two fragments, which are not yet related by a dominance relation. At the core of the secondary fragment is the possessive relation POSS; the possessor is not yet determined and therefore expressed in terms of a λ-abstracted variable, which will eventually be bound to the bound variable of the specifier semantics.

First, the bar projections of A and N inherit their semantic representations from their sole, lexical daughter node by (19). With rule (23), the semantic representation of the AP *former* emerges as (27). Then rule (20) combines (27) with the semantics of *car* to derive (28), the semantics of *former car* as $\bar{\mathrm{N}}$. Here the secondary fragment dominates the main one, which is due to the identity of main and secondary fragment for the noun *car*.

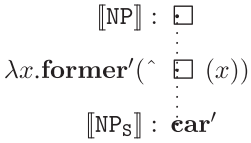
(27)
$[\![\mathrm{AP}]\!]$: ⊏□
⋮
$[\![\mathrm{AP}_{\mathrm{S}}]\!]$: $\lambda P \lambda x.\mathbf{former}'(\hat{}P(x))$

(28)



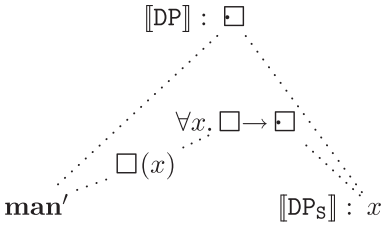
According to rule (23), *former car* as NP is then assigned the semantic constraint in (29):

(29)



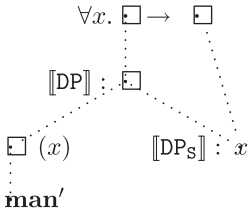
Semantic construction for the specifier proceeds in an analogous fashion. Using the rules (19), (23), (21) and (23)—in this order—yields the constraint (32) from the lexical entries (26d, e):

(30)



This constraint implies already that the universal fragment cannot outscope the main DP fragment. This is due to the fact that the main fragment dominates the two bottom fragments, and that these fragments are dominated by different holes in the ∀-fragment.[9] Any attempt to let the ∀-fragment dominate the main DP fragment by one hole would thus result in a structure that is incompatible with a tree structure. Consider e.g. an attempt to let the left hole of the universal fragment dominate the main DP fragment. Here the secondary DP fragment would be dominated by both holes of the universal fragment:

(31)

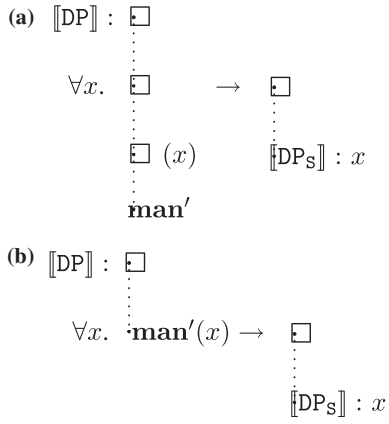


[9] Recall that the existence of two holes in one fragment means that there is a *branching* in the underlying CLLS constraint, and that material located on one branch of the constraint cannot dominate material on the other branch.

Letting the right hole of the universal fragment dominate the main fragment (the second attempt at giving the universal fragment scope over the main DP fragment) would result in a mirror-inverted, likewise tree-structure-incompatible structure.

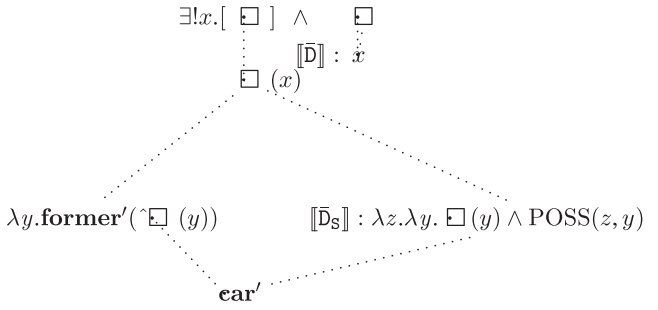
Thus, the main DP fragment has scope over the universal fragment, which is reflected in the constraint (32a). This is the eventual result of semantic construction for the DP; for the sake of readability, I will simplify this constraint by strengthening the dominance relations in the restriction of the universal fragment to identity.[10] This yields the eventual constraint (32b) for the DP *the man*:

(32)



The result of combining (26f) and (29) into the semantics of the $\bar{\text{D}}$ constituent of *every man's former car* by rule (22) is then the dominance diamond (33):

(33)

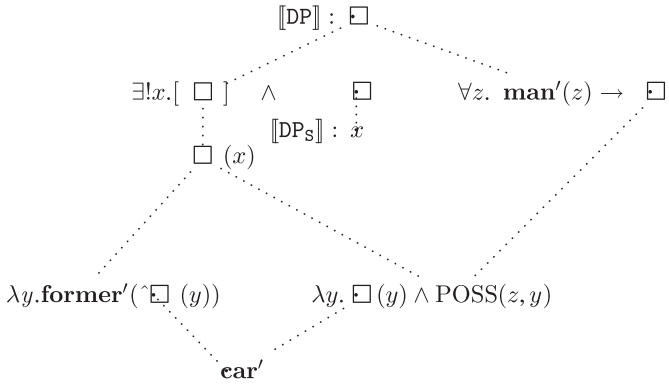


The next step is then the integration of the specifier semantics (32) by rule (24). According to this rule, the hole that constitutes the main DP fragment dominates material that is dominated by the first hole of the existential fragment (the secondary fragment of the $\bar{\text{D}}$ constituent) as well as material dominated by the second hole in the existential fragment (the main $\bar{\text{D}}$ fragment). But this means that the main DP fragment

[10] These holes are irrelevant for the simple DP *the man*; they come in because semantic construction for DPs generalises to the worst case. Thus, they will play a crucial role in the construction of structure (35) for the semantics of (12) as a whole.

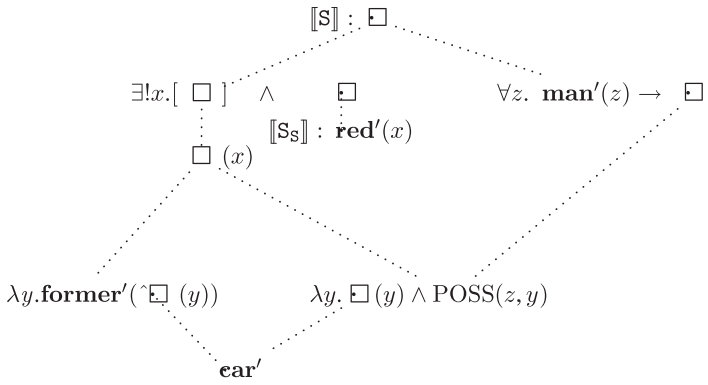
dominates the existential fragment (the argumentation was expounded for the similar case (30) above).

(34)



Adding the semantics of the VP *is red* according to rule (25) yields the constraint (35) for the semantics of (12):

(35)



This constraint combines two dominance structures, with the fragment '$\square(x)$' and the possessive fragment being part of both. I will now show that this constraint has as solutions only the four λ-terms as described in Sect. 2. There is scope ambiguity between the two DPs, and between *former* and the possessive relation; since these ambiguities are independent, there are four possible solutions. This means that (35) does not overgenerate, even though a casual glance at the structure might suggest that *former* might get wide scope over the specifying DP.

To see this, one has to take into account that the involved fragments are of a specific type and that only fragments of a specific type can fill a specific hole.

In the upper right dominance structure, all fragments are of type t (staying extensional for simplicity's sake) except the possessive fragment at the bottom. The holes in this structure must all be filled by a fragment of type t, except the ones in the fragment '$\square(x)$' and the possessive fragment, whose holes are of type $\langle e, t\rangle$.

The fragments in the lower left dominance structure are of type $\langle e, t\rangle$, except '$\square(x)$', which is of type t. The holes in this structure must all be filled by a fragment of type $\langle e, t\rangle$.

The result of this is that the two dominance structures cannot interfere with each other. In particular, immediate scope of the *former*-fragment over the universal fragment is barred because the hole of the former is of type $\langle e, t\rangle$, while the latter is of type t. I.e., the only solutions of (35) are (13a–d), repeated here as (36a–d):

(36) (a) $\exists!x.[\forall z.\mathbf{man}'(z) \rightarrow \mathbf{former}'(\hat{}(\mathbf{car}'(x))) \wedge \mathrm{POSS}(z, x)] \wedge \mathbf{red}'(x)$
(b) $\exists!x.[\forall z.\mathbf{man}'(z) \rightarrow \mathbf{former}'(\hat{}(\mathbf{car}'(x) \wedge \mathrm{POSS}(z, x)))] \wedge \mathbf{red}'(x)$
(c) $\forall z.\mathbf{man}'(z) \rightarrow \exists!x.[\mathbf{former}'(\hat{}(\mathbf{car}'(x) \wedge \mathrm{POSS}(z, x)))] \wedge \mathbf{red}'(x)$
(d) $\forall z.\mathbf{man}'(z) \rightarrow \exists!x.[\mathbf{former}'(\hat{}(\mathbf{car}'(x))) \wedge \mathrm{POSS}(z, x)] \wedge \mathbf{red}'(x)$

6 Conclusion and Outlook

In this paper, I sketched an approach to the semantic construction of a challenging case that combines two phenomena, viz., scope ambiguity and a complicated semantic interaction of syntactic sister constituents. The proposed approach used the flexibility inherent in semantic underspecification formalisms to formulate interface rules that derive underspecified semantic representations on the basis of a surface-oriented syntactic analysis. This strategy made possible an integrated approach to the two phenomena.

I will conclude this paper by pointing out at least one straightforward extension of the approach to an even more complicated example, as pointed out by Larson and Cho (2003):

(37) John's old new car

There are three scope bearing items in the restriction of the quantifier introduced by the *'s* element in (37), viz., the adjectives and the possessive relation. If one assumes that the word order of the adjectives determines their scope, there are still three theoretical scoping possibilities: The possessive relation might have scope above, between, or below the adjectives. The corresponding readings are[11]:

(38) (a) 'the car that belongs to John and is quite old for a new car' (POSS $<$ *old* $<$ *new*)
(b) 'the car that is quite old for a new car belonging to John' (*old* $<$ POSS $<$ *new*)
(c) 'the car that is quite old for a recently acquired car of John' (*old* $<$ *new* $<$ POSS)

Larson and Cho (2003) accept only the second of these readings for (37). But, considering the fact that the simpler (39) is acceptable (in the interpretation 'the car which is comparatively old for a new car'), I believe that the first reading is possible, too.

[11] '$<$' is an abbreviation for the relation 'has scope over'.

(39) the old new car

This reading can be based on appropriate semantic analyses of *old* and *new*, e.g., (40):

(40) **old′/new′** holds between properties (sets of entities) P and individuals x iff $P(x)$ and the lifespan of x extends further/less further back than the average lifespan of the elements of P

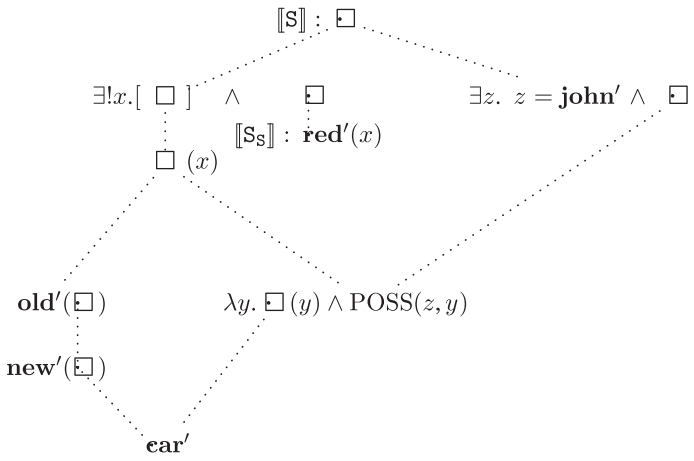
Accepting the third reading (38c) too, is in line with Larson and Cho's (2003) analysis as sketched in (10): The surface order of the adjectives is compatible with both of them adjoining to the PP or to the noun, or with the first one adjoining to the PP, and the second, to the noun. This follows directly from their analysis: if the first adjective adjoins to the NP *car* it is impossible for the second one to adjoin to a higher node, in particular, the PP node. In contrast, their analysis could not block a reading in which either adjective adjoins to the PP.

These semantic intuitions can also be modelled in the proposed syntax-semantics interface. I will discuss the semantic representation of the following example:

(41) John's old new car was red

The semantic representation of (41) is (42). The crucial difference to (35) is the twofold application of rule (20)[12]:

(42)



Since the proper name has no real scope, which can be modelled by simplifying $\exists z.z = \mathbf{john}' \wedge P(z)$ to $P(\mathbf{john}')$, for any P, there are exactly three solutions for this constraint. To keep these solutions as simple as possible, I use η-equality (equivalence of $\lambda x.P(x)$ and P, if x does not occur free in the function P):

[12] To maximise the common ground between (35) and (42), I use a semantic representation for *John* that involves an existential quantifier. The meaning of *John* is represented as the set of properties such that there is an entity equal to John that has them:

(43) (a) $\exists!x.[\lambda y.\mathbf{old}'(\mathbf{new}'(\mathbf{car}'))(y) \wedge \mathrm{POSS}(\mathbf{john}', y)](x) \wedge \mathbf{red}'(x)$
(b) $\exists!x.[\mathbf{old}'(\lambda y.\mathbf{new}'(\mathbf{car}')(y) \wedge \mathrm{POSS}(\mathbf{john}', y))](x) \wedge \mathbf{red}'(x)$
(c) $\exists!x.[\mathbf{old}'(\mathbf{new}'(\lambda y.\mathbf{car}'(y) \wedge \mathrm{POSS}(\mathbf{john}', y)))](x) \wedge \mathbf{red}'(x)$

These three readings correspond exactly to the three interpretations of (37) as listed in (38). Each of them states that a specific entity is red, in (43a), the entity described in (38a), and so on. This result is achieved because the scope of the adjectives is fixed in (42), but the scope of the possessive relation is not. Consequently, it can take scope above, between, or below the adjectives.

In sum, the proposed approach to compositionality can be adapted even to extremely complicated challenges to compositionality. The price that one has to pay (relinquishing opacity) is more than compensated by its flexibility and its compatibility with surface-oriented syntactic analyses.

(i) $[\![\mathtt{DP}]\!] : \boxed{\cdot}$

$\exists x.\ x = \mathbf{john}' \wedge \boxed{\cdot}$

$[\![\mathtt{DP_S}]\!] : x$

Acknowledgements Thanks to the audience at the workshop "Empirical challenges and analytical alternatives to strict compositionality" at ESSLLI '05 in Edinburgh, the guest editors, and three anonymous reviewers for valuable comments and discussion.

References

Barwise, J., & Cooper, R. (1981). Generalized quantifiers and natural language. *Linguistics & Philosophy, 4*, 159–219.

Cooper, R. (1983). *Quantification and syntactic theory*. Dordrecht: Reidel.

Copestake, A., Flickinger, D., Pollard, C., & Sag, I. (2005). Minimal Recursion Semantics. An introduction. *Research on Language and Computation, 3*, 281–332.

Davidson, D. (1967). The logical form of action sentences. In N. Rescher (Ed.), *The logic of decision and action* (pp. 81–95). Pittsburgh: Pittsburgh University Press.

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

Egg, M. (2004). Mismatches at the syntax-semantics interface. In S. Müller (Ed.), *Proceedings of the 11th International Conference on Head-Driven Phrase Structure Grammar* (pp. 119–139). Stanford: CSLI Publications.

Egg, M. (2006). Anti-Ikonizität an der Syntax-Semantik-Schnittstelle. *Zeitschrift für Sprachwissenschaft, 25*, 1–38.

Egg, M., Koller, A., & Niehren, J. (2001). The constraint language for lambda-structures. *Journal of Logic, Language, and Information, 10*, 457–485.

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

Hodges, W. (2001). Formal features of compositionality. *Journal of Logic, Language, and Information, 10*, 7–28.

Hoeksema, J. (1985). *Categorial morphology*. New York: Garland Publishing.

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

Larson, R. (1998). Events and modification in nominals. In D. Strolovitch & A. Lawson (Eds.), *Proceedings from SALT VIII* (pp. 145–168). Ithaca.

Larson, R., Cho, S. (2003). Temporal adjectives and the structure of possessive DPs. *Natural Language Semantics, 11*, 217–247.

May, R. (1985). *Logical Form. Its structure and derivation*. Cambridge: MIT Press.
Montague, R. (1974). In R. Thomason (Ed.), *Formal philosophy: Selected papers of Richard Montague*. New Haven: Yale University Press.
Pollard, C., & Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. CSLI and University of Chicago Press.
Reyle, U. (1993). Dealing with ambiguities by underspecification: Construction, representation, and deduction. *Journal of Semantics, 10*, 123–179.
Richter, F., & Sailer, M. (2004). Basic concepts of Lexical Resource Semantics. In A. Beckmann & N. Preining (Eds.), *European Summer School in Logic, Language and Information 2003. Course Material I*, Vol. 5 of *Collegium Logicum*. Wien: Publication Series of the Kurt Gödel Society, pp. 87–143.
Schubert, L., & Pelletier, F. (1982). From English to logic: Context-free computation of 'conventional' logical translation. *American Journal of Computational Linguistics, 8*, 26–44.
von Stechow, A. (1996). The different readings of 'Wieder': A structural account. *Journal of Semantics, 13*, 87–138.
Westerståhl, D. (1998). On mathematical proofs of the vacuity of compositionality. *Linguistics & Philosophy, 21*, 635–643.