

Ellipsis Resolution with Underspecified Scope

Michael Schiehlen*

Institute of Natural Language Processing
Azenbergstr. 12
70174 Stuttgart
Fed. Rep. of Germany
mike@ims.uni-stuttgart.de

Abstract

The paper presents an approach to ellipsis resolution in a framework of scope underspecification (Underspecified Discourse Representation Theory). It is argued that the approach improves on previous proposals to integrate ellipsis resolution and scope underspecification (Crouch, 1995; Egg et al., 2001) in that application processes like anaphora resolution do not require full disambiguation but can work directly on the underspecified representation. Furthermore it is shown that the approach presented can cope with the examples discussed by Dalrymple et al. (1991) as well as a problem noted recently by Erk and Koller (2001).

1 Introduction

Explicit computation of all scope configurations is apt to slow down an NLP system considerably. Therefore, underspecification of scope ambiguities is an important prerequisite for efficient processing. Many tasks, like ellipsis resolution or anaphora resolution, are arguably best performed on a representation with fixed scope order. An underspecification formalism should support execution of these tasks. This paper aims to upgrade an existing underspecification formalism for scope ambiguities, Underspecified Discourse Representation Theory (UDRT) (Reyle, 1993), so that both ellipsis and anaphora resolution can work on the underspecified structures.

*Many thanks for discussion and motivation are due to the colleagues in Saarbrücken.

Several proposals have been made in the literature on how to integrate scope underspecification and ellipsis resolution in a single formalism, e.g. Quasi-Logical Forms (QLF) (Crouch, 1995) and the Constraint Language for Lambda Structures (CLLS) (Egg et al., 2001). That work has primarily aimed at devising methods to untangle quantifier scoping and ellipsis resolution which often interact closely (see Section 6). To this end, description languages have been modelled in which the disambiguation steps of a derivation need not be executed but rather can be explicitly recorded as constraints on the final structure. Constraints are only evaluated when the underspecified representation is finally interpreted. In contrast, UDRT aims at providing a representation formalism that supports interpretation processes such as theorem proving and anaphora resolution. Understood in this sense, underspecification often obviates the need for complete disambiguation. Another consequence is, however, that the strategy of postponing disambiguation steps is in some cases insufficient. A case in point is the phenomenon dubbed Missing Antecedents by Grinder and Postal (1971), illustrated in sentence (1): One of the pronoun's antecedents is overt, the other is supplied by ellipsis resolution.

- (1) Harry sank a destroyer_{*i*} and so did Bill and they_{*i,j*} both went down with all hands. (Grinder and Postal, 1971, 279)

Most approaches to ellipsis and anaphora resolution, e.g. (Asher, 1993; Crouch, 1995; Egg et al., 2001), can readily derive the reading. But consider:

- (2) Harry sometimes reads a book about a sea-battle and so does Bill. They borrow those books from the library.

Example (2) still retains five readings (Are there two or even more books? are there one, two, or more than two sea-battles?). An underspecified representation should not be committed to any of these readings, but it should specify that “a book” has narrow scope with respect to the conjunction. Furthermore, an approach to underspecification and ellipsis resolution should make clear why this representation is to be constructed for the discourse (2). While QLF fails the first requirement (a single representation), CLLS fails the second (triggers for construction).

- (3) * A destroyer_i went down in some battle and a cruiser did too. Harry sank both destroyers_{i,j}.

The discourse in (3) is not well-formed. But none of the approaches mentioned can ascertain this fact without complete scope resolution (or ad-hoc restrictions).

The paper is organized as follows. Section 2 gives a short introduction to UDRT. Section 3 formulates the general setup of ellipsis resolution assumed in the rest of the paper. Section 4 presents a proposal to deal with scope parallelism in an underspecified representation. Section 5 shows how ellipsis can be treated if it is contained in its antecedent. Section 6 describes a way to model the interaction of ellipsis resolution and scope resolution in an underspecified structure. In section 7 strict and sloppy identity is discussed. Section 8 concludes.

2 Underspecified Discourse Representation Structures

Reyle (1993) proposes a formalism for underspecification of scope ambiguity. The underspecified representations are called Underspecified Discourse Representation Structures (UDRSs). Completely specified UDRSs correspond to the Discourse Representation Structures (DRSs) of Kamp and Reyle (1993). A UDRS is a triple consisting of a **top label**, a set of **labelled conditions** or discourse referents, and a set of **subordination constraints**. A UDRS is (partially) disambiguated by adding subordination constraints. A UDRS must, however, always comply with the following well-formedness conditions: (1) It does not contain cycles (subordination is a partial order). (2) No label is subordinated to two labels which are siblings, i.e.

part of the same complex condition (subordination is a tree order).

Figure 1 shows the UDRS for sentence 4 in formal and graph representation.

- (4) Every professor found most solutions.

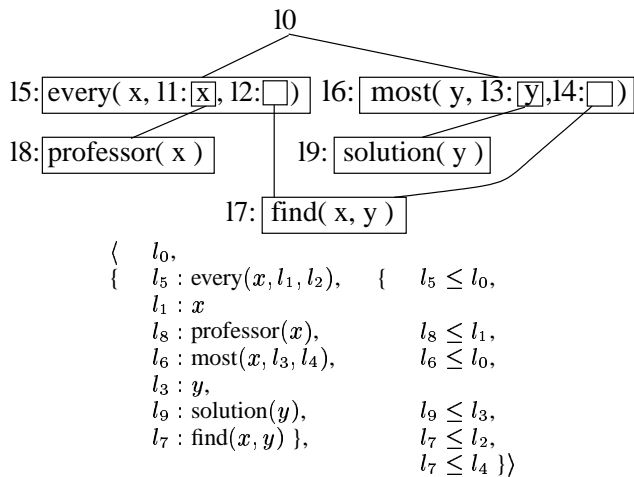


Figure 1: UDRS for sentence (4)

For pronouns and definite descriptions another type of constraint is introduced, **accessibility constraints**. l_2 is accessible from l_1 ($l_1 \text{ acc } l_2$) iff $l_1 \leq l_2$ or $\exists l_3 : l_1 \leq l_3$ and l_3 is a right sibling of l_2 in a condition expressing material implication or a generalized quantifier (Kamp and Reyle, 1993). An accessibility constraint $l_1 \text{ acc } l_2$ indicates that l_2 is an anaphoric element or a presupposition; it thus can be used as a trigger for anaphora resolution and presupposition binding (van der Sandt, 1992). To bind an anaphor l_2 to some antecedent expression l_3 , a **binding constraint** ($l_2 \leftrightarrow l_3$) and an equality constraint between two discourse referents are introduced. Binding constraints are interpreted as equality in the subordination order. Any unbound presuppositions remaining after anaphora resolution (corresponding to accessibility constraints without binding constraints) are accommodated, i.e. they end up in an accessible scope position which is as near to the top as possible. Figure 2 shows the UDRS for sentence (5). Accessibility constraints are marked by broken lines, binding constraints are shown as squiggles.

- (5) John revised his paper.

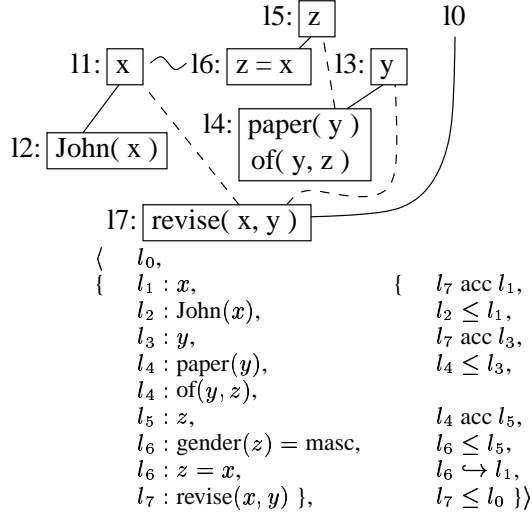


Figure 2: UDRS for sentence (5)

3 Ellipsis Resolution

Sag (1976) and Williams (1977) have argued convincingly that VP ellipsis should be resolved on a level where scope is fixed. Dalrymple et al. (1991) distinguish two tasks in ellipsis resolution:

1. determining parallelism, i.e. identifying the source clause c^s (the antecedent of the ellipsis), the parallel elements in the source clause p_1^s, \dots, p_n^s , the parallel elements in the target (i.e. elliptical) clause p_1^t, \dots, p_n^t , and the non-parallel elements in the target n_1^t, \dots, n_m^t ,
2. interpreting the elliptical (target) clause c^t , given the interpretation of $c^s, p_1^s, \dots, p_n^s, p_1^t, \dots, p_n^t$.

The paper does not have much to say about task 1. Rather, some “parallelism” module is assumed to take care of task 1. This module determines the UDRS representations of the source clause and of the source and target parallel elements. It also provides a bijective function π associating the parallel labels and discourse referents in source and target.

For task 2 we adopt the substitutional approach advocated by Crouch (1995): The semantic representation of the target c^t is a copy of the source c^s where target parallel elements have been substituted for source parallel elements ($c^t = c^s[p_1^s/p_1^t, \dots, p_n^s/p_n^t]$). In contrast to Higher-Order

Unification (HOU) (Dalrymple et al., 1991) substitution is deterministic: Ambiguities somehow cropping up in the interpretation process (i.e. the strict/sloppy distinction) require a separate explanation.

4 Scope Parallelism

It has frequently been observed that structural ambiguity does not multiply in contexts involving ellipsis: A scope ambiguity associated with the source must be resolved in the same way in source and target. Sentence (6) e.g. has no reading where all professors found the same solution but the students who found a solution each found a different one.

- (6) Every professor found a solution, and most students did, too.

Scope parallelism seems to be somewhat at odds with the idea of resolving ellipses on scopally underspecified representations. If the decisions on scope order have not yet been taken, how can they be guaranteed to be the same in source and target? The QLF approach (Crouch, 1995) gives an interesting answer to this question: It uses re-entrancy to propagate scope decisions among parallel structures.

In sentence (6), we see that a scope decision can resolve more than one ambiguity. In UDRT, scope decisions are modelled as subordination constraints. Consequently, sentence (6) shows that subordination constraints may affect more than one pair of labels. Remember that in each process of ellipsis resolution i the parallelism module returns a bijective function π_i which expresses the parallelism between labels and discourse referents in source and target. As sentence (6) shows, a subordination constraint that links two source labels l_1 and l_2 also links the labels corresponding to l_1 and l_2 in a parallel structure i , i.e. $\pi_i(l_1)$ and $\pi_i(l_2)$ for all i . Thus the subordination constraint does not distinguish between source label and parallel labels. Formally, we define two labels l_1 and l_2 to be equivalent ($l_1 \approx l_2$) iff $l_1 = l_2 \vee \exists i : (l_1 = \pi_i(l_2) \vee l_2 = \pi_i(l_1)) \vee \exists l_3 : (l_1 \approx l_3 \wedge l_3 \approx l_2)$. Now we can model the parallelism effects by stipulating that a subordination constraint connects two *equivalence classes* $[l_1]_{\approx}$ and $[l_2]_{\approx}$ rather than two individual labels l_1 and l_2 . But every label in one class should not be linked

to every label in the other class. If l_1 and l_2 are the source labels, it does not make sense, and actually will often lead to a structure violating the well-formedness conditions, to connect e.g. the source label l_1 with some target label $\pi_i(l_2)$. Thus we still need a proviso that only such labels can be linked that were determined to be parallel to the source label in the *same sequence* of ellipsis resolutions. We talk about a sequence here, because, as sentence (7) shows, ellipses may be nested.

- (7) John arrived before the teacher did (*1 arrive*), and Bill did too (*2 arrive before the teacher did (1 arrive)*).

For the implementation of classes, we take our cues from Prolog (Erbach, 1995; Mellish, 1988). In Prolog, class membership is most efficiently tested via unification. For unification to work, the class members must be represented as instances of the representation of the class. If class members are mutually exclusive, their representations must have different constants at some argument position. In this vein, we can think of a label as a Prolog term whose functor denotes the equivalence class and whose argument describes the sequence of ellipsis resolutions that generated the label. Such a sequence can be modelled as a list of numbers which denote resolutions of particular ellipses. An empty list indicates that the label was generated directly by semantic construction. We will call the list of resolution numbers associated with a label the label’s **context**. For reasons that will become clear only in section 7 discourse referents also have contexts. Although subordination constraints connect classes of labels, they are always evaluated in a particular context. Thus $l_1 \leq l_9$ (or, more explicitly, $\forall K : l_1[K] \leq l_9[K]$) can be spelled out as $l_1[] \leq l_9[]$ or $l_1[1] \leq l_9[1]$, but never $l_1[] \leq l_9[1]$ because in this case context changes.

While scope resolution is subject to parallelism and binding is too (see Section 7), examples like (9) suggest that accommodation sites need not be parallel¹. (“*The jeweler*” is accommodated with wide

¹Asher et al. (2001) use parallelism between subordination and accommodation to explain the “wide-scope puzzle” observed by Sag (1976). Sentence (8) has only one reading: A specific nurse saw all patients.

- (8) A nurse saw every patient. Dr. Smith did too.

scope, but “*his wife*” is not.)

- (9) If Peter is married, his wife is lucky and the jeweler is too.

Ellipsis resolution works as follows. In semantic construction, all occurrences of labels and discourse referents (except those in subordination constraints) are assigned the empty context ($[]$). Whenever an occurrence of ellipsis is recognized, a counter is incremented. Let i be the counter’s new value. All parallel labels l and discourse referents x in the target are replaced by their counterparts in the source ($\pi_i(l)$ and $\pi_i(x)$). After substitution proper ($c^t = c^s[p_1^s/p_1^t, \dots, p_n^s/p_n^t]$), the new resolution number i is added to the context of every label and discourse referent in c^t . Finally, the non-parallel target elements (n_1^t, \dots, n_m^t), if any, are added to the semantic representation of the target. Figure 3 shows the UDRS for sentence (6) after ellipsis resolution.

$$\left\{ \begin{array}{l} l_0[], \\ l_1[] : \text{every}(x[], l_2[], l_3[]), \\ l_2[] : x[], \\ l_4[] : \text{professor}(x[]), \\ l_5[] : y[], \\ l_6[] : \text{solution}(y[]), \\ l_7[] : \text{find}(x[], y[]), \\ \\ l_8[] : \text{and}(l_9[], l_9[1]), \\ l_1[1] : \text{most}(x[1], l_2[1], l_3[1]), \\ l_2[1] : x[1], \\ l_4[1] : \text{student}(x[1]), \\ l_5[1] : y[1], \\ l_6[1] : \text{solution}(y[1]), \\ l_7[1] : \text{find}(x[1], y[1]) \end{array} \right\} \quad \left\{ \begin{array}{l} l_1 \leq l_9, \\ \\ l_4 \leq l_2, \\ l_5 \leq l_9, \\ l_6 \leq l_5, \\ l_7 \leq l_3, \\ l_7 \leq l_5 \\ l_8 \leq l_0 \end{array} \right\}$$

Figure 3: UDRS for sentence (6)

Erk and Koller (2001) discuss sentence (10) which has a reading in which each student went to the station on a different bike. The example is problematic for all approaches which assume source and target scope order to be identical (HOU, QLF, CLLS).

- (10) John went to the station, and every student did too, on a bike.

Erk and Koller (2001) go on to propose an extension of CLLS that permits the reading. In the approach proposed here no special adjustments are needed: The indefinite NP is designated by labels that do not have counterparts in the source. The subordination order is still the same in source and target.

5 Antecedent-Contained Ellipsis

The elliptical clause can also be contained in the source, cf. example (11).

(11) John greeted every person that Bill did.

In this case the quantifier embedding the elliptical clause necessarily takes scope over the source. The treatment of this phenomenon in QLF and CLLS, which consists in checking for cyclic formulae after scope resolution, cannot be transferred to UDRT, since it presupposes resolved scope. Rather we make a distinction between *proposed* source and *actual* source. If the target is not contained in the (proposed) source, the actual source is the proposed source. Otherwise, the actual source is defined to be that part of the proposed source which is potentially subordinated² by the nuclear scope of the quantifier whose restriction contains the target.

6 Interaction of Ellipsis Resolution and Quantifier Scoping

Sentence (6) has a third reading in which the indefinite NP “*a solution*” is raised out of the source clause and gets wide scope over the conjunction. In this case, the quantifier itself is not copied, only the bound variables which remain in the source. Generally, a quantifier that may or may not be raised out of the source is only copied if it gets scope inside the source. Thus the exact determination of the semantic material to be copied (i.e. of the source) is dependent on scope decisions. Consequently, in an approach working on fully specified representations (Dalrymple et al., 1991) scope resolution cannot simply precede ellipsis resolution but rather is interleaved with it. Crouch (1995) considers order-sensitivity of interpretation a serious drawback. In his approach, underspecified formulae are copied in ellipsis resolution. In such formulae, quantifiers are not expressed directly but rather stored in “q-terms”. Q-terms are interpreted as bound variables. Quantifiers are introduced into interpreted structure only when their scope is resolved. Since scope resolution is seen as constraining the structure rather than as an operation of its own, the QLF approach manages to

² l_1 is potentially subordinated to l_2 in a UDRS iff the subordination constraint $l_1 \leq l_2$ could be added to the UDRS without violating well-formedness conditions.

untangle scope resolution and ellipsis resolution. In CLLS (Egg et al., 2001) no copy is made in the underspecified representation. In both approaches, the quantifier is not copied until scope resolution.

But the Missing Antecedents phenomenon (1) shows that a copy of the quantifier must be available even before scope resolution so that it can serve as antecedent. But this copy may evaporate later on when more is known about the scope configuration. We will call conditions that possibly evaporate *phantom conditions*. For their implementation we make use of the fact that a UDRS collects labelled conditions and subordination constraints in *sets*. In sets, identical elements collapse. Thus, a condition that is completely identical to another condition will vanish in a UDRS. Phantom conditions only arise by parallelism; hence they are identical to their originals but for the context of their labels and discourse referents. To capture the effect of possible evaporation, it suffices to make the update of context in a phantom condition dependent on the relevant scope decision. To implement phantom conditions in a Prolog-style environment, we insert a meta-variable in place of the context and control its instantiation by a special constraint expressing the dependence on the pertinent subordination constraint (a **conditional constraint**). Conditional constraints have the form $X = (l_1 \leq l_2? [i|K] : K)$ where X is the context variable, i is a resolution number, and K is some context.

$$\left\langle \begin{array}{l} l_0[], \\ l_1[] : \text{every}(x[], l_2[], l_3[]), \\ l_2[] : x[], \\ l_4[] : \text{professor}(x[]), \\ l_5[] : y[], \\ l_6[] : \text{solution}(y[]), \\ l_7[] : \text{find}(x[], y[]), \\ \\ l_8[] : \text{and}(l_9[], l_9[1]), \\ l_1[1] : \text{most}(x[1], l_2[1], l_3[1]), \\ l_2[1] : x[1], \\ l_4[1] : \text{student}(x[1]), \\ l_5(X) : y(X), \\ l_6(X) : \text{solution}(y(X)), \\ l_7[1] : \text{find}(x[1], y(X)) \end{array} \right\}, \quad \left\{ \begin{array}{l} l_1 \leq l_9, \\ \\ l_4 \leq l_2, \\ l_5 \leq l_0, \\ l_6 \leq l_5, \\ l_7 \leq l_3, \\ l_7 \leq l_5, \\ l_8 \leq l_0, \\ \\ X = (l_5 \leq l_9? [1] : []) \end{array} \right\}$$

Figure 4: UDRS for sentence (6)

Figure 4 illustrates a UDRS with a phantom condition (again representing sentence (6)). A graphical

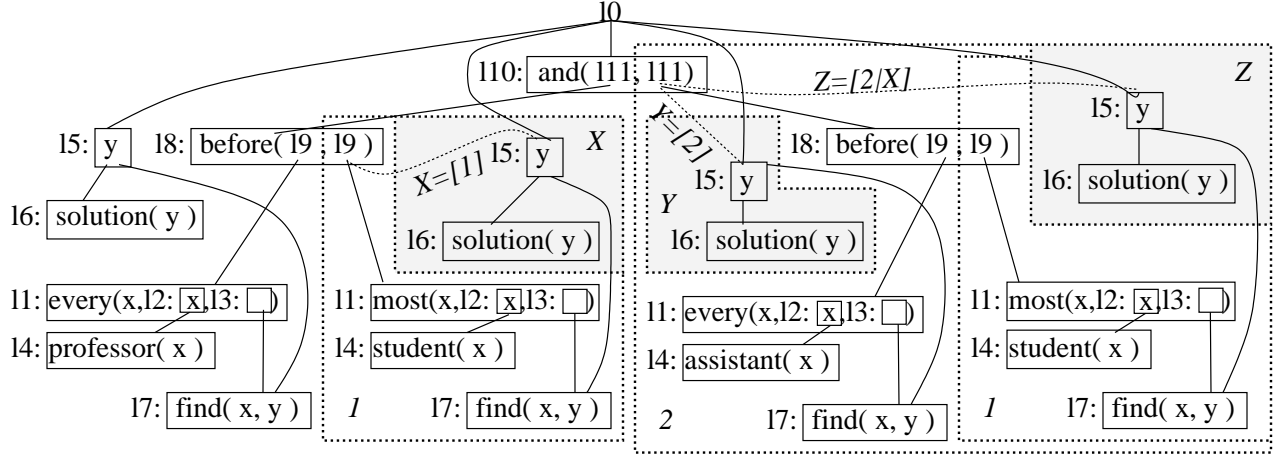


Figure 5: UDRS for sentence (12)

representation of this UDRS can be seen in the first conjunct of Figure 5. Contexts are marked by dotted boxes, conditional constraints by a dotted subordination link with an equation.

If the subsequent discourse contains a plural anaphoric NP such as “*both solutions*”, two or more discourse referents designating solutions are looked for. Two such discourse referents are found ($y[]$ and $y(X)$), but they will collapse unless X is set to $[1]$. After consultation of the conditional constraint, the subordination constraint $l_5 \leq l_9$ is added. If the subsequent discourse contains a singular anaphoric NP “*the solution*”, anaphora resolution introduces the converse subordination constraint $l_8 \leq l_5$.

Examples involving nested ellipsis (cf. sentence (12)) require copying of context variables and conditional constraints.

(12) Every professor found a solution before most students did, and every assistant did too.

To copy a context variable X , it is replaced by a new variable Z . The conditional constraint evaluating X ($X = (l_5 \leq l_9? [1] : [])$) is copied to a conditional constraint evaluating Z . In this constraint Z is conditionally bound back to X : $Z = (l_5 \leq l_{11}? [i|X] : X)$, where i is the new resolution number and l_{11} is the top label of the source. Consider the UDRS for sentence (12) in Figure 5 with three conditional constraints: $X = (l_5 \leq l_9? [1] : [])$, $Y = (l_5 \leq l_{11}? [2] : [])$, and $Z = (l_5 \leq l_{11}? [2|X] : X)$. The ex-

istential NP “*a solution*” is copied three times (if $l_5 \leq l_9$), once (if $l_5 > l_9$ and $l_5 \leq l_{11}$), or not at all (if $l_5 > l_{11}$).

7 Strict and Sloppy Identity

In the treatment of strict/sloppy ambiguity, we follow the approach of Kehler (1995) which predicts five readings for the notorious example (13) from Gawron and Peters (1990).

(13) John revised his paper before the teacher did, and Bill did too.

In Kehler’s (1995) approach, strict/sloppy ambiguity results from a bifurcation in the process of ellipsis resolution: There are two ways to copy the binding constraint linking an anaphor with its antecedent if the antecedent is in the source³. Let $l_1(\mathbf{K}) : x(\mathbf{K}) = y(\mathbf{J})$, $l_1(\mathbf{K}) \hookrightarrow l_2(\mathbf{J})$ be a binding constraint as introduced by anaphora resolution. The sloppy way to copy the constraint is the usual one, i.e. updating the contexts with the new resolution number.

³If the antecedent of a pronoun is outside the source, the copied pronoun is bound to the source pronoun (strict interpretation), not directly to the antecedent, cf. the reading missing in sentence (14) in which Bill will say that Mary helped Bill before Susan helped John.

(14) John will testify that Mary helped him before Susan did, and so will Bill.

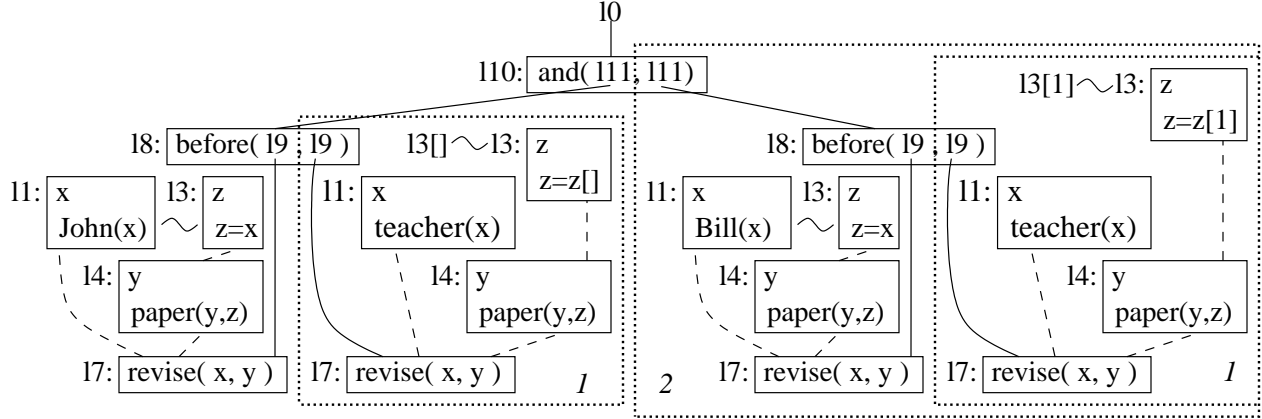


Figure 6: UDRS for a reading of sentence (13)

sloppy $l_1[i|K] : x[i|K] = y[i|J], l_1[i|K] \leftrightarrow l_2[i|J]$

The strict way is to bind the variable of the copied pronoun to the variable of the source pronoun.

strict $l_1[i|K] : x[i|K] = x(K), l_1[i|K] \leftrightarrow l_1(K)$

Figure 6 shows the UDRS for a particular reading of sentence (13): John and Bill revised their own papers before the teacher revised John’s paper. The pronoun is first copied strict ($z[1] = z[] (= x[])$), then sloppy ($z[2] = x[2]$), and finally strict again ($z[2, 1] = z[1] (= x[])$).

We have tacitly assumed that source pronouns are resolved before ellipsis resolution. No mechanism has been provided to propagate binding constraints in parallel structures. But note that the order of operations in anaphora resolution is also constrained by structure: Anaphors embedded in other anaphors need to be resolved first (van der Sandt, 1992). Ellipsis resolution may be considered on a par with anaphora resolution in this respect.

Anaphors can occur in phantom conditions as well (cf. sentence (15)).

(15) John revised a paper of his before the teacher did, and Bill did too.

An extension of the copy rules for binding constraints along the lines of Section 6 is straightforward (see below). If the embedding quantifier gets

wide scope ($l_i > l_j$), source and target constraints collapse (sloppy), or the target constraint asserts self-binding (strict).

sloppy $l_1(X) : x(X) = y(Y), l_1(X) \leftrightarrow l_2(Y),$

strict $l_1(X) : x(X) = x(K), l_1(X) \leftrightarrow l_1(K)$

$$X = (l_i \leq l_j ? [i|K] : K),$$

$$Y = (l_i \leq l_j ? [i|J] : J)$$

There are, however, some problems with this extension. See Figure 7 for the strict-sloppy-strict reading of sentence (15). If the indefinite NP gets intermediate scope between “before” and “and”, the context variable X will be set to [], Y and Z to [2]. A clash follows, since $l_3[2]$ is bound both to $l_1[2]$ and $l_3[]$. To remedy this defect, we stipulate that resolving the strict/sloppy ambiguity may partially disambiguate the scope structure: If in the course of resolving a particular ellipsis several anaphors are copied with different choices in the strict/sloppy bifurcation, the conditional constraints are evaluated so that the anaphors cannot turn out to be the same. This condition ensures that in the strict-sloppy-strict reading illustrated in Figure 7 the indefinite NP gets narrow scope under “before”.

8 Conclusion

The paper has presented a new approach to integrate ellipsis resolution with scope underspecification. In contrast to previous work (Crouch, 1995)

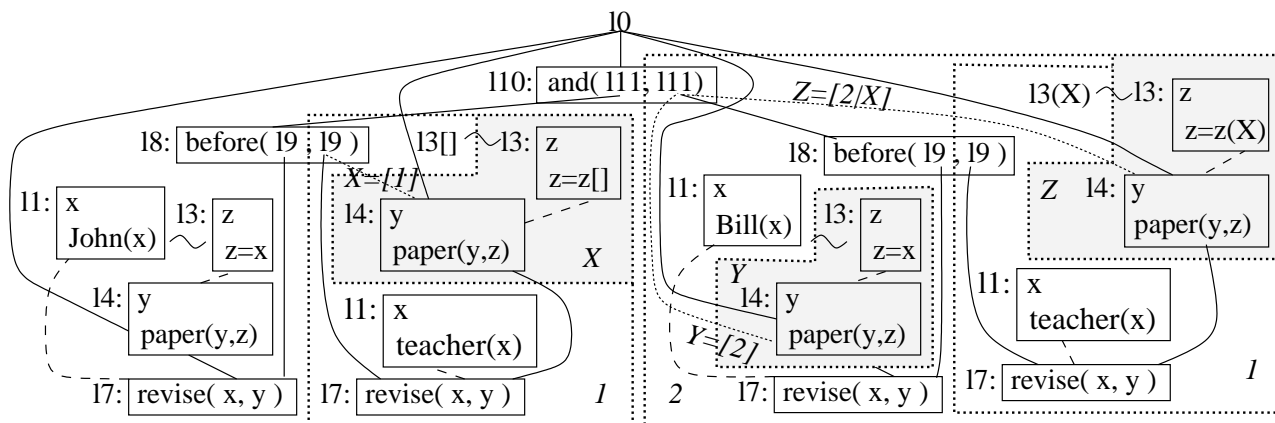


Figure 7: UDRS for sentence (15)

(Egg et al., 2001) the proposed underspecified representation facilitates the resolution of anaphora by providing explicit representations of potential antecedents. To this end, a method to encode “phantom conditions” has been presented, i.e. subformulae whose presence depends on the scope configuration. Furthermore, a method to deal with scope parallelism in scopally underspecified structures has been proposed. The proposed method has no trouble accounting for cases where the scope order in antecedent clause and elliptical clause is not entirely identical (Erk and Koller, 2001). Finally, it has been shown that the approach can cope with a wide variety of test examples discussed in the literature.

References

- Nicholas Asher, Daniel Hardt, and Joan Busquets. 2001. Discourse Parallelism, Ellipsis, and Ambiguity. *Journal of Semantics*, 18(1).
- Nicholas Asher. 1993. *Reference to Abstract Objects in Discourse*. Kluwer.
- Richard Crouch. 1995. Ellipsis and Quantification: A Substitutional Approach. In *Proceedings of EACL’95*, pages 229–236, Dublin, Ireland.
- Mary Dalrymple, Stuart M. Shieber, and Fernando C.N. Pereira. 1991. Ellipsis and Higher-Order Unification. *Linguistics and Philosophy*, 14:399–452.
- Markus Egg, Alexander Koller, and Joachim Niehren. 2001. The Constraint Language for Lambda Structures. *Journal of Logic, Language and Information*, 10.
- Gregor Erbach. 1995. ProFIT: Prolog with Features, Inheritance and Templates. In *Proceedings of EACL’95*, Dublin, Ireland.
- Katrin Erk and Alexander Koller. 2001. VP Ellipsis by Tree Surgery. In *Proceedings of the 13th Amsterdam Colloquium*.
- Jean Mark Gawron and Stanley Peters. 1990. *Anaphora and Quantification in Situation Semantics*. Number 19 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, CA.
- John Grinder and Paul M. Postal. 1971. Missing Antecedents. *Linguistic Inquiry*, 2:269–312.
- Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic: An Introduction to Modeltheoretic Semantics of Natural Language*. Kluwer.
- Andrew Kehler. 1995. *Interpreting Cohesive Forms in the Context of Discourse Inference*. Ph.D. thesis, Harvard University.
- Chris Mellish. 1988. Implementing Systemic Classification by Unification. *Computational Linguistics*, 14:40–51.
- Uwe Reyle. 1993. Dealing with Ambiguities by Underspecification: Construction, Representation and Deduction. *Journal of Semantics*, 10(2):123–179.
- Ivan Sag. 1976. *Deletion and Logical Form*. Ph.D. thesis, MIT.
- Rob A. van der Sandt. 1992. Presupposition Projection as Anaphora Resolution. *Journal of Semantics*, 9(4):333–377.
- Edwin Williams. 1977. Discourse and Logical Form. *Linguistic Inquiry*, 8(1):101–139.