The Availability of Partial Scopings in an Underspecified Semantic Representation

Alistair Willis Suresh Manandhar

Department of Computer Science
University of York
York, YO10 5DD, UK.
{agw,suresh}@cs.york.ac.uk

Abstract

Much of the recent work in computational semantics has addressed the problem of analysing ambiguous sentences; that is, sentences which have more than one possible meaning. A favoured solution has been to represent ambiguous sentences with *underspecified* meaning languages which are not committed to any one of the possible meanings, but which allow additional information to be used to determine a correct meaning as it becomes available.

A problem facing many underspecified meaning representations is that partial scopes may be expressed which do not correspond to well formed final meaning representations. This paper presents work in progress on providing a solution to this; a structure is proposed which is independent of the grammatical and semantic theories used, but which allows only those partial scopings which are actually available to a speaker. An efficient deductive framework for determining whether a partial scoping is available is also provided.

Keywords: Quantifier scope ambiguity, underspecification

1 Introduction

A long standing problem within computational semantics is the correct analysis of ambiguous sentences; that is, sentences which may have more than one meaning. Given a sentence displaying quantifier scope ambiguity, such as *Every man loves a woman*, part of the problem of representing the sentence's meaning is to distinguish between the two possible meanings:

$$\forall x(rep(x) \rightarrow \exists y(woman(y) \wedge love(x, y)))$$

where every man loves a (possibly) different woman, or

$$\exists y(woman(y) \wedge \forall x(man(x) \rightarrow love(x, y)))$$

where a single woman is loved by every man.

Various methods exist to obtain all the available readings of such sentences. Cooper [Coo83] described a system of "storing" the quantifiers as λ-expressions during the parsing process and retrieving them at the sentence level; different orders of quantifier retrieval generate different readings of the sentence. More recently, ambiguity has been captured by exploiting non-determinism in deductive semantic construction systems, exemplified by the constraint-based semantics project [DLPS96] and Carpenter's categorial logic [Car94].

However, systems are required which, as well as capturing ambiguity, should *underspecify* meaning. We characterise underspecification as a pair of requirements:

1. the meaning of a sentence should be represented in a way that is not committed to any one of the possible (intended) meanings of the sentence, and

2. it should be possible to incrementally introduce *partial* information about the meaning, if such information is available, and without the need to undo work that has already been done (monotonicity).

This paper discusses some shortcomings with quantifier scope representation in some contemporary underspecified systems (eg. Alshawi and Crouch's Quasi-Logical Form (QLF) [AC92] and Reyle's Underspecified Discourse Representation Theory (UDRT) [Rey95]), and outlines a system that is currently being developed to address them.

2 Requirements for Underspecified Scope Representation

A principal aim of systems providing an underspecified representation of quantifier scope ambiguity is to be able to represent *partial* scopings. That is, it should be possible to state that some of the quantifiers have some scope relative to each other, while remaining uncommitted to the relative scope of the remaining quantifiers. However, representations which simply allow partial scopes to be stated without further analysis do not adequately capture the behaviour of quantifiers in a sentence.

A semantic representation that is similar to a (very naïve form of) QLF is used to illustrate how simply *representing* partial scope orders without further analysis does not clearly provide all the information that is available from a partial scoping. Quantified terms in this representation are labelled with an index and a partial scoping is represented by a list of indices at a predicate, so that the order of the indices represents the relative scope of quantifiers in the final logical form. For example, for the sentence *Every man loves a woman*, the two fully disambiguated forms are:

```
[+i,+j]:love(<+i every x man(x)>, <+j exists y woman(y)>)
```

$$\forall x(man(x) \rightarrow \exists y(woman(y) \wedge love(x, y)))$$

and

```
[+j,+i]:love(<+i every x man(x)>, <+j some y woman(y)>)
```

$$\exists y(woman(y) \land \forall x(man(x) \rightarrow love(x, y)))$$

(shown with their equivalent logical forms).

This representation of the relative scoping is adequate for a sentence containing only two quantifiers, but difficulties emerge when the sentences become more complicated. Consider the sentence *Every representative of a company saw most samples.* The fully unresolved underspecified representation would be something like[1]:

```
_:see(<+i every x rep.of(x, <+j exists y co(y)>)>,
                                        <+k most z sample(z)>)
```

In principle, a scope ordering can be expressed again as the ordered list, so the fully resolved representation:

```
[+j,+i,+k]:see(<+i every x rep.of(x, <+j exists y co(y)>)>,
                                        <+k some z sample(z)>)
```

could be intended to represent the logical form:

```
exists(y, co(y), every(x, rep.of(x, y),
                      most(z, sample(z), see(x, z)))
```

This approach turns out to be inadequate because there is no link between the well formedness of the underspecified structure and the logical form it is intended to represent. So for example, consider:

```
[+i,+k,+j]:see(<+i every x rep.of(x, <+j exists y co(y)>)>,
                                        <+k most z sample(z)>)
```

Although this is a well formed QLF, there is no well formed sentence of logic to which it corresponds. One possible logical form that this might be intended to represent is:

```
every(x, rep.of(x, y), most(z, sample(z),
                    exists(y, co(y), see(x, z))))
```

but this is not well formed as the variable `y` in `rep.of` appears outside its quantifier. In fact, there are no well formed sentences which reflect the desired scoping. A condition on well formedness could be that it should be possible to generate at least one well formed semantic form which respects the partial scoping given; there are many approaches to ambiguity which avoid this problem of overgeneration, such as nested Cooper Storage [Kel86]. Our aim, however, is to present a system in which it is immediately clear whether or not a *partial* scoping leads to a well-formed sentence, without requiring recourse to the final logical form.

While this representation is too permissive in expressing semantic forms, it is also insufficiently informative about the interaction between the quantifiers when partial scopes are considered. Consider the previous example again, with the partial scoping stipulating that *a* should outscope *most*:

[1] In fact, QLF would generally have two lists of quantifiers; one to scope to `love`, as shown, and another to scope to `rep.of`.

```
[+j,+k]:see(<+i every x rep.of(x, <+j exists y co(y)>)>,
                                         <+k most z sample(z)>)
```

There are two well formed logical forms in which *a* outscopes *most*, those being:

```
exists(y, co(y), every(x, rep.of(x, y)),
                   most(z, sample(z), see(x, z)))
```

and

```
exists(y, co(y), most(z, sample(z),
                 every(x, rep.of(x, y), see(x, z))))
```

In all readings where *a* outscopes *most*, *a* must also outscope *every*; this should be readily apparent from a fully understood theory of quantifier scopes without the need to generate all the possible readings. But further, given the theory of scope availability that we are using (see section 3), the partial scoping only licenses the first of the above two readings. Again, this should be immediately apparent from the formal representation.

So a solution to the problem of representing partial scopes of a sentence displaying quantifier scope ambiguity should have the following properties:

- It should be provable whether a partial scoping represents an available reading of a sentence, without reference to the final logical form. It would be intuitively better if the representation language expressed *only* those scopings which are actually available to the speaker.

- The representation should be "maximally informative"; that is, if a partial scoping imposes constraints upon the other quantifiers in a sentence, this should be provable, again without recourse to the final logical form. For example, if there is only one reading of the sentence which respects the partial scoping, then this should be immediately apparent.

- This information should be available entirely from the representation of the sentence's scope information. It should not be necessary to refer to the sentence's grammatical form or semantic form.

And a further desirable property of the system would be that:

- It should be possible to adapt the system to different semantic representations without their requiring significant modifications.

It should be stressed here that, while QLF has been used as the running example, the points in this section apply equally to any system that represents partial relative scopes of quantifiers either as a list (as done by QLF and in HPSG [PS94]) or as a set of pairs of quantifiers (as in UDRT).

In this paper, we propose a method of representing quantifier scope in an ambiguous sentence that fulfils the above criteria. The structures described are intended to be independent of whatever underspecified representation is used to represent the sentence's meaning, but to allow as much information as possible about the availability of partial scopes to be deduced. QLF, UDRT or the f-structures of LFG [KB82] are all appropriate underspecified representations; the close correlation between these three structures is discussed by van Genabith and Crouch [vGC96], although f-structures are intended to be syntactic rather than semantic representations. Suppose F is an underspecified representation of the grammatical relations in a sentence. An associated scope structure, F_ψ is defined so that terms representing quantifiers in F map onto elements in F_ψ. A relation $\triangleright$ is defined over the terms in F_ψ, such that a quantifier in F outscopes some other quantifier in F only if the relation $\triangleright$ holds between the corresponding terms in F_ψ. A constraint language for representing scope structures is then defined in which it is provable whether a partial scoping between elements in F_ψ is available. It is also possible to deduce additional partial scope information about F_ψ within the constraint language without needing to refer back either to F, or to the original NL sentence.

The structures are not intended to be semantic representations as such, rather, they should be considered to provide *some* of the information that is necessary for semantic composition (specifically, that which refers to relative scopes of quantifiers). In that sense, the structure can be thought of as analogous to the scope order prefixes which are present in a QLF, although it is intended that the structure be sufficiently rich to express the required scope availability information in its own right.

Nor do we discuss the actual mechanisms by which disambiguation is to take place (for example, partial scopes due to pronoun resolution). Our aim is to show how any arbitrary sets of partial scopes may be correctly represented, rather than to provide an analysis of how such partial scopes might be found.

3 A Structure for Representing Quantifier Scopes

Our approach is based on the theory of scope availability that claims that for any n-ary relation in a sentence, there are $n!$ possible orderings of quantifiers around that relation (this has recently been expounded by Park [Par95]). Other quantifiers in the sentence should not "intercalate" between those which are single arguments to a relation. So for example, in the sentence *Every representative of a company saw most samples*, there are four possible scopes, because there are two relations; *saw* and *of*. Around *saw*, *every* can outscope *most*, or vice versa, and around *of*, *every* can outscope *a*, or vice versa. What is not possible is a reading where *a* outscopes *most* which outscopes *every*; although this can be represented by a well formed sentence of logic (with no unbound variables), it is not available to a speaker of English.

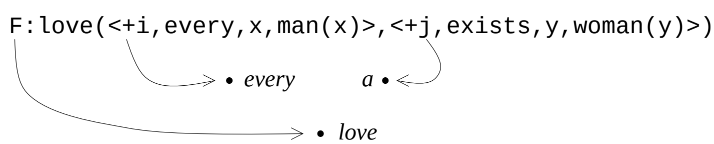
By using this theory as the basis of underspecification, we can say:

- underspecification is to be captured by allowing different possible relative scope assignments around the predicates, but
- although the predicates determine the availability of scopes, relative scope must still be expressed as a relation between the quantifiers.

Because our concern is with both the relations in a sentence and the quantifiers, the main grammatical objects under consideration are *quantifiers* and *predicates* (for example, verbs and prepositions). Quantifiers and the relations which determine their relative scope can be represented by an (irreflexively) ordered set, where the ordering gives the relative scopes. However, because the interaction between the predicates in the sentence has implications for possible scopings, it is also necessary to consider the relationship between the ordered sets. For the remainder of this section, the structures being proposed are discussed using examples and a consideration of the main linguistic phenomena to be covered. A more formal system is presented in section 5.

First consider the main grammatical objects. Here, it is assumed that the relevant grammatical terms are (only) the main relations and the quantifiers, although it is not necessary that these be the terms used. For sentences involving only extensional verbs, it would be equally possible to consider only quantifiers as the main relations (as in the example of section 5). Here we are using the terms which most clearly allow the main ideas to be demonstrated.

A set of basic elements, $\mathcal{O}$ and a mapping function ψ are required, such that ψ maps each grammatical object in the representation onto a member of $\mathcal{O}$. Figure 1 illustrates the case where a QLF-like structure is being used as the representation of the sentence *Every man loves a woman*; ψ defines a mapping of the main relation of the sentence (*love*, represented by `F` in the QLF) and the two quantifiers (represented by `+i` and `+j` respectively in the QLF) onto elements (here, *love*, *every* and *a*) in $\mathcal{O}$.



$\psi(\texttt{F}) \mapsto love$
$\psi(\texttt{+i}) \mapsto every$
$\psi(\texttt{+j}) \mapsto a$

Figure 1: Mapping QLF terms onto scope elements

We can now define a relation to represent the relative scope of the grammatical objects. First consider sentences consisting of quantifiers scoped around an extensional verb. The possible scopes can be represented as a set with an irreflexive partial ordering, $\rhd$, over its members. For the sentence *Every man loves a woman*, the possible orderings of the quantifiers *every* and *a* around the verb *love* are represented by the set and ordering:

$$\{every, a, love\} \quad every \rhd love, a \rhd love$$

The two ordering relations state that both quantifiers outscope the verb, but say nothing about their scopes relative to each other. This represents a completely underspecified meaning. An unambiguous reading of the sentence is represented when $\rhd$ defines a complete ordering of the set. So if the relation $every \rhd a$ were added, the reading:

$$\forall x.man(x) \to \exists y.woman(y) \wedge love(x, y)$$
$$every \rhd a \rhd love$$

would be represented. Alternatively, adding $a \rhd every$ to the underspecified form would represent the reading:

$$\exists y.woman(y) \wedge \forall x.man(x) \rightarrow love(x,y)$$
$$a \rhd every \rhd love$$

The introduction of a further relation which does not lead to a well formed sentence (such as *love* $\rhd$ *every*) is shown by the antisymmetry of $\rhd$ being violated.

This system also allows the interaction of quantifiers with negation and intensional verbs to be represented, by using an additional element to represent the negation or the intensional structure. *Everyone didn't leave* should be represented as:

$$\{every, neg, leave\} \quad every \rhd leave, neg \rhd leave$$

Control structures can be introduced in a similar fashion, where the behaviour of the quantifiers and the control verb can be captured by an appropriate relation $\rhd$. For example, consider *Every teacher persuaded a pupil to listen.* The possible readings are of the form:

$$\forall x.teacher(x) \rightarrow \exists y.pupil(y) \wedge persuade(x, y, listen(y))$$

and

$$\exists y.pupil(y) \wedge \forall x.teacher(x) \rightarrow persuade(x, y, listen(y))$$

The possible scopes can be captured by including a further element for *persuade.* This gives the possible scopes as:

$$\{every, a, persuade, listen\} \quad every \rhd persuade, a \rhd persuade$$
$$persuade \rhd listen$$

which only allows for the required two possible readings (*every* outscopes *a* or vice versa). However, for object raising verbs, a *de dicto* reading is required:

$$\forall x.teacher(x) \rightarrow want(x, \exists y.pupil(y) \wedge listen(y))$$

as well as the two possible *de re* readings which are available (as with *persuade*):

$$\forall x.teacher(x) \rightarrow \exists y.pupil(y) \wedge want(x, listen(y))$$

$$\exists y.pupil(y) \wedge \forall x.teacher(x) \rightarrow want(x, listen(y)).$$

The underspecified versions of *Every teacher wanted a pupil to listen* is:

$$\{every, a, want, listen\} \quad every \rhd want, a \rhd listen$$
$$want \rhd listen$$

This relation $\rhd$ allows any of the three possible readings; because *a* may appear as an argument to *want* the *de dicto* reading is accounted for.

While using a single set of elements correctly accounts for the possible scopes of quantifiers in the restricted set of sentences discussed so far, the introduction of prepositional attachment to NPs and relative clauses is more complex. Consider the sentence *Every representative of a company saw most samples.* The presence of two binary relations *of* and *saw* implies that there should be $2!.2! = 4$ readings. Continuing with the system developed so far, these possibilities could be represented by a pair of sets:

$$\{\textit{every, most, see}\} \quad \textit{every} \rhd \textit{see},\ \textit{most} \rhd \textit{see}$$

$$\{\textit{every, a, rep.of}\} \quad \textit{every} \rhd \textit{rep.of},\ \textit{a} \rhd \textit{rep.of}$$

where completing the scope orderings generates the four available readings. However, this does not capture all the available information on partial scopings. Consider the two readings where *most* outscopes *every* (in a format more similar to generalised quantifiers):

$$most(z, sample(z), every(x, exists(y, co(y), rep.of(x, y)), see(x, z)))$$

and

$$most(z, sample(z), exists(y, co(y), every(x, rep.of(x, y), see(x, z)))).$$

In both cases, as well as outscoping *every*, *most* always outscopes *a*, but this is not apparent from the representation as is. This can be remedied by defining a *dominance* relation. In the current case, say that *every* dominates *a*. Then anything that outscopes *every* also outscopes anything that *every* dominates (here, *a*). So if *most* outscopes *every*, then it also outscopes *a* because *every* dominates *a*. This is a product of Park's remark about intercalating quantifiers; if *most* were to outscope *every* but not *a*, then *most* would be intercalating between the other two quantifiers.

This behaviour can be captured by using a tree structure, where each of the nodes is one of the ordered sets representing the scopes around a relation. At any node, N, each of the daughter nodes has (exactly) one element in common with N, otherwise, any element appears only once in the structure. So, considering again the sentence *Every representative of a company saw most samples*, the scope information is represented as the tree:

$$\begin{array}{ll} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{see},\ \textit{most} \rhd \textit{see} \\ \quad\quad | & \\ \{\textit{every, a, rep.of}\} & \textit{every} \rhd \textit{rep.of},\ \textit{a} \rhd \textit{rep.of} \end{array}$$

where $\rhd$ defines relations between the elements both at the parent node and the daughter node. Now, say that an element X dominates another element Y (denoted as $X \hookrightarrow Y$) if X and Y are (distinct) elements in a set at some node, and X is also in the parent node. Also, $\hookrightarrow$ is transitive and irreflexive. So in the example given:

$$\textit{every} \hookrightarrow a \text{ and } \textit{every} \hookrightarrow \textit{of},$$

but *every* $\not\hookrightarrow$ *every*. Now we can say that X outscopes Y either if $X \rhd Y$, or if $X \rhd Z$ and $Z \hookrightarrow Y$. So if *most* outscopes *every*, it also outscopes a as *every* $\hookrightarrow a$.

4 Representing Partial Scopes

It was stated in section 3 that scope availability is accounted for by the relative scope of quantifiers around their predicates, and that this would form the basis of our theory of underspecification. In this section, we discuss how arbitrary partial scopings can be rewritten as scopings around predicates, even if there is no predicate to which the quantified terms are both arguments. Doing so yields a clear account of scope interaction and a check for scope availability; the details are given in section 5, but an intuitive overview is given here.

Consider again the sentence *Every representative of a company saw most samples*, but with the additional requirement that *most* should outscope *a*. The representation of the quantifiers and relations in this sentence is:

$$\begin{array}{cl} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{see},\ \textit{most} \rhd \textit{see} \\ | & \textit{most} \rhd a \\ \{\textit{every, a, rep.of}\} & \textit{every} \rhd \textit{of},\ a \rhd \textit{rep.of} \end{array}$$

While this represents the partial scope required, it is not obvious that *most* must also outscope *every*. What we are looking for is the scopes of quantifiers around their relations that give the required partial scoping. If the scope *every* $\rhd$ *most* were specified at the topmost node:

$$\begin{array}{cl} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{most},\ \textit{every} \rhd \textit{see} \\ | & \textit{most} \rhd \textit{see} \\ \{\textit{every, a, rep.of}\} & \textit{every} \rhd \textit{of},\ a \rhd \textit{of} \end{array}$$

then *most* $\rhd$ *a* follows from *most* $\rhd$ *every* and *every* $\hookrightarrow$ *a*. But it is also explicit that *most* outscopes *every*; the additional information required. The aim of the rewriting is to find the minimal ordering at the nodes that gives the required partial scope in all available readings of the sentence.

For a second example, note that there is only one possible reading of the sentence which respects the scoping *a* $\rhd$ *most*, that being:

$$exists(y, co(y), every(x, rep.of(x, y), most(z, sample(z), see(x, z))))$$

This should be reflected by the existence of only a single ordering of the elements within their sets whereby this scoping is represented. This is indeed the case, the only possible such ordering being:

$$\begin{array}{cl} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{most} \rhd \textit{see} \\ | & \\ \{\textit{every, a, of}\} & a \rhd \textit{every} \rhd \textit{of} \end{array}$$

The scoping *a* $\rhd$ *most* follows from this structure from *a* $\rhd$ *every*, *every* $\rhd$ *most* and the transitivity of $\rhd$.

Finally, there are cases where quantifiers do not stand in any relation to each other. Consider the reading of the sentence:

$$every(x, exists(y, co(y), rep.of(x, y)), most(z, sample(z), see(x, z)))$$

represented by the orderings:

$$\begin{array}{cl} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{most} \rhd \textit{see} \\ | & \\ \{\textit{every, a, of}\} & \textit{every} \rhd a \rhd \textit{of} \end{array}$$

It is not possible to infer either *most* $\rhd$ *a* or *a* $\rhd$ *most* from the information available. This is correct, and arises because the quantifier *a* occurs within *every*'s restriction, and *most* occurs within *every*'s scope.

5 Constraints for scope representation

The previous sections provided an intuitive overview of the structures to be used for representing partial scope availability, and showed how different linguistic phenomena were to be accounted for; this section provides a more formal discussion of the structures and their interpretation. The main requirement, discussed in section 2, is that all the information about which scopes are available must be obtained directly from these structures, rather than needing recourse to the grammatical structure or semantic representation.

As described in section 3, the available scopings are represented by a tree structure, where each node of the tree is a set of elements, P, taken from a set $\mathcal{O} = \{\alpha, \beta, \gamma, \ldots\}$. An irreflexive, antisymmetric, transitive relation $\rhd$ is defined over pairs of elements which appear in the tree. For any node, each daughter node is also a set, such that each daughter set d_i has exactly one element in common with P, a different element for each of the d_i. An element can only appear once in the tree, unless it appears in a mother/daughter node pair. So the tree:

$$\begin{array}{c} \{\alpha,\beta,\gamma\} \\ | \quad\quad | \\ \{\alpha,\delta,\xi\} \quad \{\beta,\zeta,\eta\} \end{array}$$

is a correct scope representation, but the trees

$$\begin{array}{c} \{\alpha,\beta,\gamma\} \\ | \\ \{\delta,\xi,\zeta\} \end{array} , \quad \begin{array}{c} \{\alpha,\beta,\gamma\} \\ | \\ \{\alpha,\beta,\delta\} \end{array} \quad \text{and} \quad \begin{array}{c} \{\alpha,\beta,\gamma\} \\ | \quad\quad | \\ \{\alpha,\delta,\xi\} \quad \{\alpha,\zeta,\eta\} \end{array}$$

are not; in the first, there is no common node between the parent and daughter, in the second there are two common nodes, and in the third, the element α appears in two daughter nodes.

This behaviour can be captured by defining a scope representation as a pair $\langle P, \mathcal{D} \rangle$, where P is the set of elements at that node and $\mathcal{D}$ is the set of daughters. Then a scope representation can be recursively defined as:

- There is a (countably infinite) set of elements, $\mathcal{O} = \{\alpha, \beta, \gamma, \ldots\}$
- If $P \subset \mathcal{O}$, then $\langle P, \{\} \rangle$ is a scope representation.
- If S and S' are scope representations, such that $S = \langle P, \mathcal{D} \rangle$ and $S' = \langle P', \mathcal{D}' \rangle$ where no element occurs in both S and S' and there is some element, σ, such that $\sigma \in P$, then $\langle \{\sigma\} \cup P', \{S\} \cup \mathcal{D}' \rangle$ is a scope representation.
- An element σ *occurs in* a scope representation $\langle P, \mathcal{D} \rangle$ iff either:
  - $\sigma \in P$, or
  - there is some scope representation d_i such that $d_i \in \mathcal{D}$ and σ occurs in S'.
- If S is a scope representation such that $S = \langle P, \mathcal{D} \rangle$, then:

– S is a substructure of S, and

– If $d_i \in \mathcal{D}$ then any substructure of d_i is a substructure of S.

We now consider how a scope representation can be represented in a constraint language. A scope representation can be effectively defined in terms of dominance between elements and common set membership at a node. The constraint language is:

$$\begin{array}{lcll} \psi, \phi & \longrightarrow & \sigma \circ \rho & \text{Common set membership} \\ & & \sigma \hookrightarrow \rho & \text{Dominance} \\ & & \sigma \rhd \rho & \text{Outscoping} \\ & & \psi \wedge \phi & \text{Conjunction} \end{array}$$

where $\sigma, \rho \in \mathcal{O}$.

S is a well formed scope representation. F_S is the conjunction of constraints of C, where C is the minimal constraint set satisfying the following:

1. If $\langle P, \mathcal{D} \rangle$ is a substructure of S, where $\sigma, \rho \in P$ and $\sigma \neq \rho$, then C contains the constraint $\sigma \circ \rho$

2. If $\langle P, \mathcal{D} \rangle$ is a substructure of S, such that $d_i \in \mathcal{D}$ and $d_i = \langle P', \mathcal{D}' \rangle$, then if $\sigma \in P, \sigma \in P', \rho \in P'$ and $\sigma \neq \rho$, then C contains the constraint $\sigma \hookrightarrow \rho$

3. If C contains the constraints $X \hookrightarrow Y$ and $Y \hookrightarrow Z$, then C contains the constraint $X \hookrightarrow Z$.

The clause F_S provides the structural information for S. So for example, the scope representation:

$$\begin{array}{c} \{\alpha, \beta, \gamma\} \\ \quad | \qquad | \\ \{\alpha, \delta\} \quad \{\beta, \xi\} \end{array}$$

is represented by the clause:

$$\begin{array}{ll} & \alpha \circ \beta \wedge \alpha \circ \gamma \wedge \beta \circ \gamma \wedge \alpha \circ \delta \wedge \beta \circ \xi \\ \wedge & \\ & \beta \circ \alpha \wedge \gamma \circ \alpha \wedge \gamma \circ \beta \wedge \delta \circ \alpha \wedge \xi \circ \beta \\ \wedge & \\ & \alpha \hookrightarrow \delta \wedge \beta \hookrightarrow \xi \end{array}$$

Note that the symmetry of $\circ$ is stated explicitly in the structural constraint. For clarity, future examples state only one of each pair of constraints $X \circ Y$ and $Y \circ X$.

The structural constraint may be conjoined with any number of scope constraints of the form $X \rhd Y$, where X and Y appear in S. So the constraint representation of the structure:

$$\begin{array}{ll} \{\textit{every, most, see}\} & \textit{every} \rhd \textit{see, most} \rhd \textit{see} \\ | & \\ \{\textit{every, a, rep.of}\} & \textit{every} \rhd \textit{rep.of, a} \rhd \textit{rep.of} \end{array}$$

is (leaving the symmetry of $\circ$ implicit):

$$every \circ most \wedge every \circ see \wedge most \circ see \wedge every \circ a \wedge every \circ rep.of \wedge a \circ rep.of \wedge$$
$$every \hookrightarrow a \wedge every \hookrightarrow rep.of \wedge every \rhd see \wedge every \rhd rep.of \wedge most \rhd see \wedge a \rhd rep.of$$

A set of rules are defined on the constraints, so that additional scope information may be deduced; this process does not affect scope information already present (monotonicity). The deduction process is driven by determining what scoping of quantifiers around their relations would give the required partial scoping (as discussed in section 4). It turns out that for a partial scoping $X \rhd Y$, what is required is *the node in which different elements dominate X and Y*. To see this, consider the scope representation for the sentence *Every representative of a company saw most samples with some defects*:

$$\begin{array}{ccc} & \{every, most\} & \\ \diagup & & \diagdown \\ \{every, a\} & & \{most, some\} \end{array}$$

where for clarity, the nodes representing the relations *saw, of* and *with* have been omitted. If the partial scoping $a \rhd some$ were required, then the only node where different elements dominate a and *some* is the topmost node; $every \hookrightarrow a$ and $most \hookrightarrow some$. Because *every* and *most* appear at a common node, add the constraint $every \rhd most$. This reduces the problem to solving $a \rhd every$; because a and *every* appear in a common node, no further information can be obtained by this method.

A set of rules to capture this behaviour is:

$$S1: \quad \frac{\Gamma \wedge X \circ Y \wedge X \hookrightarrow X' \wedge X' \rhd Y}{\Gamma \wedge X \circ Y \wedge X \hookrightarrow X' \wedge X' \rhd Y \wedge X \rhd Y \wedge X' \rhd X}$$

$$S2: \quad \frac{\Gamma \wedge X \circ Y \wedge Y \hookrightarrow Y' \wedge X \rhd Y'}{\Gamma \wedge X \circ Y \wedge Y \hookrightarrow Y' \wedge X \rhd Y' \wedge X \rhd Y}$$

$$S3: \quad \frac{\Gamma \wedge X \circ Y \wedge X \hookrightarrow X' \wedge Y \hookrightarrow Y' \wedge X' \rhd Y'}{\Gamma \wedge X \circ Y \wedge X \hookrightarrow X' \wedge Y \hookrightarrow Y' \wedge X' \rhd Y' \wedge X' \rhd X \wedge X \rhd Y}$$

$$Trans: \quad \frac{\Gamma \wedge X \rhd Y \wedge Y \rhd Z}{\Gamma \wedge X \rhd Y \wedge Y \rhd Z \wedge X \rhd Z}$$

$$Dom: \quad \frac{\Gamma \wedge X \rhd Y \wedge Y \hookrightarrow Z}{\Gamma \wedge X \rhd Y \wedge Y \hookrightarrow Z \wedge X \rhd Z}$$

where Γ is any conjunction of literals and the associativity and commutativity of $\wedge$ are assumed. For every constraint of the form $X \rhd Y$, where X and Y do not appear at a common node, at most one of the rules $S1, S2$ and $S3$ applies. We then say that:

- A constraint is in *normal form* iff applying the rules $S1, S2, S3$, *Trans* and *Dom* does not yield any new constraints.

Also note that if F is a constraint set such that for every constraint of the form $X \rhd Y$ there is a constraint $X \circ Y$, then the normal form of F can be obtained by application of only *Trans* and *Dom*.

If Γ is a clause in normal form then:

- Γ represents an *available scoping* iff it does not contain a constraint of the form $X \rhd X$.

- Γ represents a *complete scoping* iff it represents an available scoping, and for every constraint of the form $X \circ Y$ there is either a constraint $X \rhd Y$ or a constraint $Y \rhd X$.

The unavailability of a scoping follows from the irreflexivity of $\rhd$.

To illustrate the constraint solving, consider again the sentence *Every representative of a company saw most samples*, where the fully underspecified representation of the possible scopes is:

$$\begin{array}{c} \{every,\ most\} \\ | \\ \{every,\ a\} \end{array}$$

and the relations *see* and *of* have been omitted for clarity. In the fully unscoped representation, the relation $\rhd$ does not hold between any nodes. This structure is represented by the conjunction:

$$every \circ most \wedge every \circ a \wedge every \hookrightarrow a$$

If the partial scoping $most \rhd a$ were required, the simplification would be:

$$\begin{array}{ll} & every \circ most \wedge every \circ a \wedge every \hookrightarrow a \wedge most \rhd a \\ \rightarrow_{S2} & every \circ most \wedge every \circ a \wedge every \hookrightarrow a \wedge most \rhd a \wedge most \rhd every \end{array}$$

where no further rules apply so the clause is in normal form. The clause now contains the additional constraint that followed from the original scoping.

Now, suppose the partial scopings $most \rhd a$ and $every \rhd most$ were both required (to attempt to give the final scoping $every \rhd most \rhd a$, which is not an available reading). Adding these constraints gives:

$$\begin{array}{ll} & every \circ most \wedge every \circ a \wedge every \hookrightarrow a \wedge most \rhd a \wedge every \rhd most \\ \rightarrow_{S2} & every \circ most \wedge every \circ a \wedge every \hookrightarrow a \wedge most \rhd a \wedge every \rhd most \wedge \\ & \qquad most \rhd every \\ \rightarrow_{Trans} & every \circ most \wedge every \circ a \wedge every \hookrightarrow a \wedge most \rhd a \wedge every \rhd most \wedge \\ & \qquad most \rhd every \wedge most \rhd most \end{array}$$

The presence of the constraint $most \rhd most$ now indicates that the partial scopes do not represent an available reading of the sentence.

6 Conclusion and Comments

We have presented a structure which correctly represents the interaction of quantifiers in a sentence displaying quantifier scope ambiguity. The partial scopes which are expressible by the representation are exactly those which are available according to the linguistic theory of quantifiers being used. This contrasts with most underspecified representations, in which partial scopes may be expressed which may not correspond to any final well formed sentence, or which may represent readings not available to a speaker. We have also indicated how the structures might be used to represent the scopes of an underspecified system such as QLF. We have not discussed in this paper the place of these structures in a full theory of meaning; the theory currently is concerned with questions of scope availability rather than its place in semantic interpretation.

Acknowledgements The authors would like to thank Alan Frisch, Richard Crouch and Mark Steedman for helpful discussions on this work, and three anonymous reviewers for comments on this paper. The first author is funded by an EPSRC grant.

References

[AC92] H. Alshawi and R. Crouch. Monotonic Semantic Interpretation. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 32–39, Newark, Delaware, 1992.

[Car94] B. Carpenter. Quantification and scoping: A deductive account. In *Proceedings of the 13th West Coast Conference on Formal Linguistics.* CSLI Press., 1994.

[Coo83] R. Cooper. *Quantification and Syntactic Theory.* Reidel, 1983.

[DLPS96] M. Dalrymple, J. Lamping, F. Pereira, and V. Saraswat. Quantifiers, anaphora, and intensionality. cmp-lg/9504029, 1996.

[KB82] R. Kaplan and J. Bresnan. Lexical functional grammar: A formal system for grammatical representation. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–281. MIT Press, 1982.

[Kel86] W. Keller. Nested cooper storage: The proper treatment of quantification in ordinary noun phrases. In U. Reyle and C. Rohrer, editors, *Natural Language Parsing and Linguistic Theory*, Studies in Linguistics and Philosophy, pages 432–437. Reidel, 1986.

[Par95] J.C. Park. Quantifier scope and constituency. In *Proceedings of the Annual Meeting for the Association for Computational Linguistics (ACL)*, pages 205–212. Cambridge, MA, 1995.

[PS94] C. Pollard and I. A. Sag. *Head-driven Phrase Structure Grammar.* Chicago: University of Chicago Press and Stanford: CSLI Publications, 1994.

[Rey95] U. Reyle. On reasoning with ambiguities. In *Proceedings of the EACL, Dublin*, 1995.

[vGC96] J. van Genabith and R. Crouch. F-Structures, QLFs and UDRSs. In *Proceedings of the first LFG Conference*, 1996. Also available online from: `http://www-csli.stanford.edu/publications/LFG/lfg1.html`.