

Disambiguating quantifier scope in DTS

Livio Robaldo & Jurij Di Carlo
Dipartimento di Informatica, Università di Torino
robaldo@di.unito.it, dicarlo.jurij@educ.di.unito.it

Abstract

This paper proposes an extension of Dependency Tree Semantics (DTS), an underspecified logic originally proposed in [20], that uniformly implements constraints on Nested Quantification, Island Constraints and logical Redundancy. Unfortunately, this extension makes the complexity exponential in the number of NPs, in the worst cases. Nevertheless, we conducted an experiment on the Turin University Treebank [6], a Treebank of Italian sentences annotated in a syntactic dependency format, whose results seem to indicate that these cases are very rare in real sentences.

1 Introduction

Quantifier scope ambiguities may engender several interpretations of an NL sentence. For instance, (1.a) has two readings that, in standard Generalized Quantifier (GQ) approach, can be represented as in (1.b-c). (1.b) iff a particular sound was heard by every man., while (1.c) is true iff each man has heard a (potentially different) sound.

- (1) a. Every man heard a mysterious sound.
b. $\exists_y(\text{mystSound}'(y), \forall_x(\text{man}'(x), \text{heard}'(x, y)))$
c. $\forall_x(\text{man}'(x), \exists_y(\text{mystSound}'(y), \text{heard}'(x, y)))$

In order to deal with quantifier scope ambiguities, two main problems have to be taken into account. First, the number of available readings tends to degenerate into a combinatorial explosion when the number of quantifiers increases. Second, in many real cases the knowledge needed to disambiguate is not fully available during the processing of the sentence. In such cases, all readings must be somehow stored, and, afterwards, when new world knowledge becomes available, sequentially checked in order to remove the inconsistent ones.

In order to provide a flexible solution to semantic ambiguities, *Underspecified* formalisms have been recently proposed. In underspecified semantics, semantic ambiguities are seen as instances of unspecified meaning that can adopt a more restricted sense depending on the preferences grounded in the syntactic structure, world-knowledge, topic/focus distinctions and so forth. underspecified logics allow to encapsulate scope ambiguities in a single structure. Surveys in underspecified semantics may be found in [8] and [11].

The first underspecified logic that has been proposed is perhaps the algorithm of Hobbs and Shieber [13], which has subsequently evolved into Quasi Logical Form [1]. In this proposal, the underspecified representation is a *partial formula* Φ that may contain underspecified terms, called *complex terms*, in the form $\langle q, v, r \rangle$ where q is a GQ, v an individual variable and r , the *restriction* of q , another partial formula. In order to obtain the disambiguated formulae, complex terms must be solved. This is done by “pulling out” and “unstoring” the complex terms one by one. Depending on the order in which they are solved, different readings are obtained. For instance, the underspecified representation of (1.a) is shown in (2.a). By solving the complex term $\langle \forall, x, man'(x) \rangle$, we get the partially solved formula in (2.b), from which, by solving the other complex term $\langle \exists, y, mystSound'(x) \rangle$, we get reading (1.a). Conversely, by solving the two complex terms in the opposite order, we get (1.c).

- (2) a. $heard'(\langle \forall, x, man'(x) \rangle, \langle \exists, y, mystSound'(x) \rangle)$
 b. $\forall_x(man'(x), heard'(x, \langle \exists, y, mystSound'(x) \rangle))$

A more recent approach to underspecified semantics is grounded in dominance constraints between some kind of scope-bearers and some kind of scope-arguments. Underspecified Discourse Representation Theory [19], Hole Semantics [5], the approach of [25], Constraint Language for Lambda Structures (CLLS) [12], and Minimal Recursion Semantics (MRS) [10], belong to this approach. In MRS, the underspecified formula of (1) is

- (3) $\langle h_0, \{h_1: \forall_x(h_{11}, h_{12}), h_2: \exists_y(h_{21}, h_{22}), h_3: man'(x), h_4: mystSound'(y), h_5: heard'(x, y)\}, \{h_0=_q h_5, h_{11}=_q h_3, h_{21}=_q h_4\} \rangle$

This representation is a triple whose second element is a flat set of labelled subformulae called EPs. Each label (h_1, h_2 , etc.) is called an ‘handle’. The EP associated with a quantifier contains a handle for its restriction and one for its body. The third element of the triple is a set of constraints in the

form $h_x =_q h_y$, specifying that the EP handled by h_y must occur in the scope of h_x . Two disambiguations are then possible in (3): $\{h_0 = h_1, h_{12} = h_2, h_{22} = h_5\}$, which leads to (1.b), and $\{h_0 = h_2, h_{22} = h_1, h_{12} = h_5\}$, which leads to (1.c).

A new approach to underspecification of quantifier scope ambiguities has been recently proposed in [20], and implemented into a new formalism called Dependency Tree Semantics (DTS). In DTS, disambiguation is performed by explicitly showing the dependencies between involved sets, i.e. by achieving a sort of underspecified ‘Skolemization’. DTS is presented in detail in section 3.

Before presenting DTS, however, we will discuss in section 2 three main linguistic phenomena that reduce the level of ambiguity of NL sentences, namely Nested Quantification, Island Constraints and logical Redundancy. Those constraints have to be somehow carried out in an underspecified logic, in order to avoid readings not available in NL. We will briefly discuss the alternatives proposed in the aforementioned formalisms and in DTS.

Section 4 contains the core of the research presented here. It proposes to add in DTS two new constructs that implement the three mentioned constraints in terms of allowed/disallowed dependencies, in a uniform and modular way. These constructs, called positive/negative arcs, make DTS *expressively complete*, i.e. able to underspecify any possible subset of available readings. As argued by [11], although expressive completeness is a desirable property for underspecified logics, unfortunately it is in trade-off with spatial/temporal complexity, which, in DTS, turns out to be exponential in the number d of NPs. The last section, however, presents a small experiment on the Turin University Treebank (TUT) [6], whose results seem to indicate that, in real cases, the three examined constraints strongly contribute to keep d , and hence the overall computational time, low. We will then conclude by claiming that the exponential asymptotic behaviour of the complexity function is a price worth paying for devising a disambiguation process based on allowed/disallowed dependencies.

2 Constraints on underspecified formulae

As pointed out in the introduction, not all possible scopings correspond to a possible reading in NL. In this section, we investigate the three main linguistic phenomena involved in the unavailability of certain scope patterns, namely Nested Quantification, Island Constraints, and logical Redundancy.

Nested Quantification: Several unreasonable readings arise from a con-

trast between semantic/syntactic nesting of quantifiers. For instance, it is awkward to take the sentence in (4.a), where the NP *a company* occurs in the syntactic restriction of the quantifier *every*, as describing the situation corresponding to the scoping $\forall most \exists$, in which each representative saw a potentially different set of most samples, and he belongs to a different company for each sample he saw.

- (4) a. [Every_x representative of [a_y company]] saw [most_z samples] [13].
 b. $see'(\langle \forall, x, rep_of'(x, \langle \exists, y, comp'(y) \rangle) \rangle, \langle Most, z, sample'(z) \rangle)$
 c. $\langle h_0, \{h_1: \forall_x(h_{11}, h_{12}), h_2: \exists_y(h_{21}, h_{22}), h_3: Most_z(h_{31}, h_{32}),$
 $h_4: rep_of'(x, y), h_5: comp'(y), h_6: sample'(z), h_7: saw(x, z)\},$
 $\{h_0=_q h_7, h_{11}=_q h_4, h_{21}=_q h_5, h_{31}=_q h_6\} \rangle$
 d. $\exists_y(comp'(y), Most_z(sample'(z), \forall_x(rep'(x) \wedge of'(x, y), see'(x, z))))$
 e. $\forall_x(\exists_y(comp'(y), rep'(x) \wedge of'(x, y)), Most_z(sample'(z), see'(x, z)))$
 f. $Most_z(sample'(z), \forall_x(\exists_y(comp'(y), rep'(x) \wedge of'(x, y)), see'(x, z)))$
 g. $\exists_y(comp'(y), \forall_x(rep'(x) \wedge of'(x, y), Most_z(sample'(z), see'(x, z))))$
 h. $Most_z(sample'(z), \exists_y(comp'(y), \forall_x(rep'(x) \wedge of'(x, y), see'(x, z))))$

In the Hobbs and Shieber algorithm, these readings are forbidden by selecting, at each step, a complex term *not included in the restriction of any other complex term*. With this meta-rule, the algorithm, starting from the underspecified formula in (4.b), computes the five readings in (4.d-h). The same readings are triggered by the dominance constraints in the MRS representation (4.c).

Several authors, e.g. [18], claim that also (4.d) has to be forbidden, in that if q_1 occurs in the restriction of q_2 , no other quantifier can ‘intercalate’ between them in the scope order. In underspecified semantic, this view was accepted by [25] and [15] among others. The former develops a logical theory that identifies intercalating readings by deriving inconsistencies. The latter extends [14], which proposes a syntax-semantics interface from LTAG to MRS-like formulae, with a construct based on quantifier sets that prevents intercalating readings.

Island Constraints: certain syntactic structures are scope-“islands”, i.e. the scope-bearers occurring therein cannot outscope elements outside the island. Island constraints have been introduced by [21]. There are basically

two kinds of islands: finite clauses and coordinated structures. An example of the former is shown in (5.a), where \forall_x cannot outscope \exists_y in that *a student* is outside the finite clause where *every* occurs. In contrast, the scope of NPs can freely rise over non-finite clauses, as shown in (5.b), where $\forall_x \exists_y$ is available. An example of coordinated structures acting as islands is shown in (5.c). (5.c) yields two scopings only, $\forall_x \exists_y \exists_z$ and $\exists_y \exists_z \forall_x$, corresponding, respectively, to a reading where every student reads a own book and a own paper, and a reading where there is a single a book and a single paper that have been read by every student.

- (5) a. A_y student said you met every $_x$ professor. [16]
 b. A_y student wants you to meet every $_x$ professor. [16]
 c. Every $_x$ student reads a $_y$ book and a $_z$ paper. [26]

In underspecified semantic, a standard way to deal with Island constraints of the first kind introduces special ‘blocking’ place holders, one for each tensed clause. The scope of the quantifiers in the clause is required to be always included in the scope of the place holder. In MRS, the place holder may be a handle referring to the ‘top’ of the clause. [14] implements Island constraints in an MRS-like framework. Similar alternatives have been proposed in [12] and [16]. The former implements a mechanism very close to [14], while the latter devises a feature-based framework where a feature **MAXS** blocks the scope of the quantifiers occurring in finite clauses¹.

Island constraints carried by coordinate structures have received less attention in Underspecification, the most recent exception being perhaps [26]. In [26], Hole Semantics is extended to properly handle the meaning of sentences as (5.c).

Logical Redundancy: redundancy may arise when existential and universal quantifiers occur in the sentence. Existential quantifiers range over a singleton of individuals, and so they cannot induce variation of their narrow-scope sets. Analogously, universal quantifiers range over a whole domain of individuals, which do not vary on the entities in their wide-scope sets. Therefore, in (6.a-b) the scopings $\exists_x \exists_y$ and $\forall_x \forall_y$ are respectively equivalent to $\exists_y \exists_x$ and $\forall_y \forall_x$. Therefore, the latter may be taken as redundant. Nevertheless, there is an exception to the rule about universals. It concerns a

¹In [16], also prepositions as *of* in (4) are associated with a feature **MAXS**. This allows to deal with islands and nested quantifiers in terms of the same construct. Redundancy and Coordination are not considered, but the extension of the logic in that direction seems obvious.

universal having a modifier (which includes another quantifier) in its restriction. For example, in the case of (6.c), there are two possibilities, according to the intention of referring to ‘all friends of all participants’ ($\forall_x \forall_y$) or to ‘all friends of any participant’ ($\forall_y \forall_x$).

- (6) a. A_x boy read a_y book.
 b. Every_x man read every_x book.
 c. [Every_x friend of [every_y participant]] arrived.

According to [4], analogous considerations hold for any NP that denotes a *principal filter* (see [2]): proper names, definites, but also some bare indefinites. We agree with [4]; proper names, as well as singular definites, clearly denote constant sets², while definites seem to behave exactly like universals. For example, it is easy to see that (7.b-c) yield the same scopings of (6.b-c)

- (7) a. The_x men read every_y book.
 b. [The_x friends of [every_y participant]] arrived.

In underspecified semantics, logical redundancy has recently been investigated by [9] and [17]. The former devises an algorithm that allows to prevent the generation of redundant readings, while the method presented in the latter generates all readings, but allows to group them into equivalence classes.

3 Dependency Tree Semantics

In [20], a new underspecified semantic formalism called Dependency Tree Semantics (DTS) has been proposed. Well-formed structures in DTS are based on a graph G that represents the predicate-argument relations. The nodes of G are either predicates or individual variables called *discourse referents*. Predicates connect discourse referents via arcs labelled with the argument-position. Each discourse referent is also associated, via a function *quant*, with a GQ, and with a *restriction*, via a function *restr* from discourse referents to subgraphs of G .

In order to make the dependencies among sets of entities explicit, another kind of arcs is introduced, termed *SemDep* arcs, and resemble Skolem dependencies. A discourse referent is taken to depend on all discourse referents it is connected to via a *SemDep* arc. Moreover, G includes a special element

²Note that this seems to be true also for personal pronoun (*you, she, etc.*), singular demonstratives (*this, that, etc.*), and singular possessives (*his, her, etc.*).

called *Ctx*. *Ctx* refers to the context, i.e. the domain wrt which the final structure will be evaluated. All discourse referents are linked to *Ctx* via a *SemDep* arc; however, the ones linked to *Ctx* *only* are assumed to denote fixed sets of entities, i.e. to correspond to Skolem constants. The several readings of a sentence differ in the *SemDep* arcs only. For instance, both readings of (1.a) share the structures shown in fig.1.a³, but they have two different sets of *SemDep* arcs: (1.b) is conveyed by those in fig.1.b; (1.c) by those in fig.1.c.

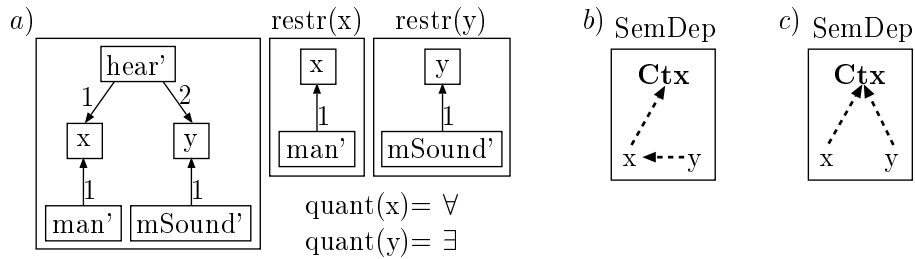


Figure 1: (a) SDG for sentence (1.a). (b-c) The two disambiguations (1.a)

In order to achieve underspecified semantics, *SemDep* arcs have to be somehow added incrementally. In [20], this is done by defining another kind of arcs, termed *SkDep* arcs. This paper makes use of other two kinds of arc, termed *positive* and *negative* arcs, and propose them as an alternative of *SkDep* arcs.

Before proceeding, it is worth stressing the main linguistic advantage of DTS, although this article is not devoted to it. DTS licenses particular readings where two or more sets of entities are introduced at the same level of scope. I refer to such interpretations with the term ‘Independent Set’ (IS) readings. Three basic kinds of IS readings need to be distinguished: distributive, collective and cumulative IS readings (cf. [22]), respectively shown in (8.a-c)

- (8) a. Two examiners marked six scripts.
 b. Three boys lift a piano.

³DTS structures are usually displayed in a graphical notation where the subgraphs corresponding to *restr*'s values are graphically shown separately. *SemDep* arcs are shown as dotted arcs; in order to increase readability, transitive *SemDep* arcs are omitted. Nevertheless, the reader must always keep in mind that they do occur, i.e. that *SemDep* describes a transitive relation. Therefore, in fig.1.b, there is an arc $y \dashrightarrow \text{Ctx}$, even if it is not shown.

- c. Three monkeys ate all our bananas.

If we allow both NPs in (8.a) to receive wide scope and we interpret the main verb *distributively*, we get a reading where there is a set of two examiners and a set of six scripts and each of the two examiners marked each of the six scripts. (8.b) is an archetypal example of the so-called *collective* readings. The sentence may receive an interpretation where the three boys lift a single piano with a joint effort. Finally, (8.c) has a *cumulative* interpretation. The sentence says that the union/cumulation of the bananas singularly eaten by each of the three monkeys includes the set of all our bananas. Currently, DTS deals with distributive IS readings only, despite their controversial existence in NL (see [23]), in that they require less formal effort. The extension of DTS coverage to collective/cumulative IS readings, which is taken as the object of future work, would require the introduction of more complex devices (see [3]).

Since DTS's coverage includes IS readings, the logic accepts more readings than other standard approaches to NL semantics. Every *partial* order between quantifier corresponds to an available reading. Examples are shown below in fig.2.

4 Positive and negative arcs

[20] defines some constraints to prevent undesirable readings arising from Nested Quantification and logical Redundancy (in contrast, Island Constraints are not considered). These constraints are defined in terms of 'meta-rules', like in Hobbs and Shieber's algorithm:

- (9) a. If either d_1 or d_2 (transitively) occurs in the restriction of a discourse referent d , and the other one does not, then the arc $d_1 \dashrightarrow d_2$ can be inserted only if the graph contains an arc $d \dashrightarrow d_2$.
- b. If $\text{quant}(d_1) = \forall$ or $\text{quant}(d_2) = \exists$, the arc $d_1 \dashrightarrow d_2$ cannot be inserted. The only exception is when $\text{quant}(d_1) = \forall$ and d_2 (transitively) occur in the restriction of d_1 .

By applying (9) to the initial DTS representation of (4.a), we get the five readings in fig.2, corresponding to (4.d-h).

This section illustrates an alternative way of managing incremental insertion of semantic dependencies, where allowed/disallowed dependencies are *explicitly* specified in the representation. This is done by inserting additional

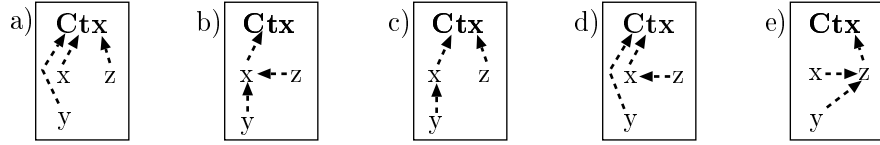


Figure 2: SemDep configurations corresponding to (4.d-h).

arcs termed *positive/negative arcs*. With these constructs, we can easily prevent redundant readings by adding a negative arc $d_1 \dashrightarrow d_2$ for each pair of discourse referents (d_1, d_2) such that $\text{quant}(d_1) = \forall$ or $\text{quant}(d_2) = \exists$ and the exception mentioned in (9.b) does not apply; all other arcs are added as positive arcs.

Nevertheless, how do we achieve (9.a)? Such constraints disallow a set N of dependencies *unless* the representation includes a set E of other dependencies. In order to mirror these concepts, we refine the framework in two ways:

- (10) a. We group positive/negative arcs into (positive/negative) sets, and we impose that all arcs in a set have to be allowed/disallowed *together*.
- b. We allow each negative set N to be associated with a further set of arcs E . Those are the arcs that constitute the exception to the disallowed dependency. In these cases, we will write $\{N|E\}$.

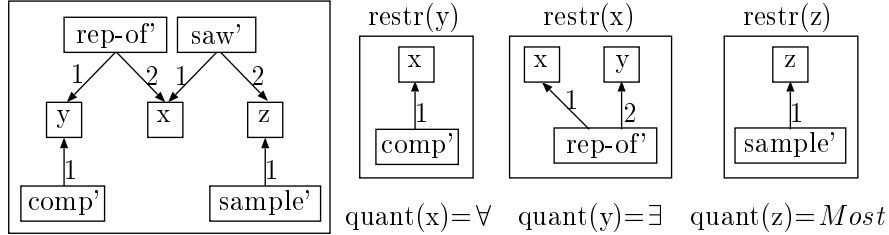
In terms of positive and negative sets, the new DTS representation of (4.a) turns out to be the one in fig.3. Positive and negative sets are separately shown as members of two families P and N in order to avoid verbose graphical representations. The positive set $\{A \dashrightarrow \text{Ctx}\}$ compactly refers to $\{x \dashrightarrow \text{Ctx}, y \dashrightarrow \text{Ctx}, z \dashrightarrow \text{Ctx}\}$. Inserting it amounts to connecting all discourse referent to Ctx ⁴.

$\{y \dashrightarrow x\}$ and $\{z \dashrightarrow x\}$ are allowed and so asserted in P . N contains four negative sets. $\{x \dashrightarrow y\}$ and $\{z \dashrightarrow y\}$ have been disallowed because no arc can enter an existential quantifier, and $\{x \dashrightarrow z\}$ because no universal quantifier can lead to a discourse referent outside its restriction. $\{y \dashrightarrow z\}$ is disallowed unless the structure contains a link from x to z . This is handled by adding the negative set with exception $\{y \dashrightarrow z | x \dashrightarrow z\}$ in N ⁵. The reader can verify

⁴We remind that only graphs where all discourse referents are linked to Ctx can be model-theoretically interpreted.

⁵Negative sets cannot be included in other negative sets, e.g. we cannot add $\{z \dashrightarrow y | x \dashrightarrow y\}$ in N : $\{z \dashrightarrow y\}$ already occurs therein. However, negative sets *can* be

that all possible combinations of the allowed arcs lead to the five readings in fig.2.



$$P = \{ \{ A \dashrightarrow \text{Ctx} \}, \{ y \dashrightarrow x \}, \{ z \dashrightarrow x \} \}$$

$$N = \{ \{ x \dashrightarrow y \}, \{ z \dashrightarrow y \}, \{ x \dashrightarrow z \}, \{ y \dashrightarrow z \mid x \dashrightarrow z \} \}$$

Figure 3: SemDep configurations corresponding to (4.d-h).

In terms of positive/negative arcs, it is rather easy to extend the coverage of DTS to Island constraints. No discourse referent outside the island may depend on a discourse referent inside it. A discourse referent inside the island is, instead, free to depend on any other discourse referent. Hence, if d_1 occurs in an island and d_2 does not, we assert $\{d_1 \dashrightarrow d_2\}$ as positive set and $\{d_2 \dashrightarrow d_1\}$ as negative set. Coordinate NPs are further constrained; if $D \equiv \{d_{11}, \dots, d_{1n}\}$ is the set of discourse referents in a coordination, and $d_1 \in D$ depends on a $d_2 \notin D$, so must any other $d \in D$. This is handled by inserting a *single* positive set $\{d_{11} \dashrightarrow d_2, \dots, d_{1n} \dashrightarrow d_2\}$. In example (5.c), then, P would be $P = \{ \{ y \dashrightarrow x, z \dashrightarrow x \}, \{ A \dashrightarrow \text{Ctx} \} \}$. It is easy to see that P generates the two desired readings only.

5 Expressivity, complexity, and real cases

It is not really necessary to add every positive/negative set, i.e. to specify every possible pattern of allowed/disallowed dependencies in the representation. For instance, disallowing a dependency is clearly equivalent to not allowing it. In fig.3, it is easy to see that the negative sets $\{x \dashrightarrow y\}$, $\{z \dashrightarrow y\}$, and $\{x \dashrightarrow z\}$ are actually useless because those arcs cannot be generated by the sets in P . Therefore, they can be removed from N . Although we defined algorithms implementing such heuristics, lack of space forbids us to provide further details.

exceptions of other negative sets: the occurrence of $\{x \dashrightarrow z\}$ in N does not prevent the insertion of $\{y \dashrightarrow z \mid x \dashrightarrow z\}$.

From the point of view of expressivity, positive/negative arcs make DTS *expressively complete*, i.e. able to underspecify any subset of dependencies. This should be a property of every underspecified logic, as argued by [11], but, unfortunately, it turns out to be in trade-off with computational complexity. In DTS, in order to represent a subset of readings that have nothing in common, the only solution is listing⁶ all corresponding positive sets. For instance, the readings of (5.c) have been “listed” in $P = \{\{y \dashrightarrow x, z \dashrightarrow x\}, \{A \dashrightarrow Ctx\}\}$. Since the number of partial orders is exponential in the number of discourse referents (cf. [7]), in the worst cases P has exponential cardinality.

However, we believe that such cases are rare in reality. To provide evidence for this claim, we analyzed the data in TUT. For each sentence we estimated the number of positive/negative arcs needed to underspecify its readings. (11) is one of the more complex sentences we found in TUT.

- (11) La_x società opera in numerosi $_y$ altri settori commerciali e industriali, annoverando tra le $_z$ sue proprietà una $_k$ catena di 20_w supermercati, (alcuni $_v$) centri turistici e una $_p$ miniera. (ALB-247)

The $_x$ society operates in several $_y$ other commercial and industrial sectors, including among its $_z$ properties a $_k$ chain of 20_w supermarkets, (some $_v$) tourism centers and a $_p$ mine.

(11) contains seven discourse referents (x, y, z, k, w, v, p), so it initially yields $7*6=42$ non-cycling positive arcs, and more than 6 billion partial orders between discourse referents. However:

- x is a singular definite with no restriction and z a plural possessive: both of them must enter Ctx only, and no discourse referent can depend on x .
- k, v, p are existential quantifiers in a coordination: no discourse referent can depend on them, and any triple of positive sets $\{k \dashrightarrow d\}$, $\{v \dashrightarrow d\}$, and $\{p \dashrightarrow d\}$ is replaced by a the positive set $\{k \dashrightarrow d, v \dashrightarrow d, p \dashrightarrow d\}$ ⁷.
- w belongs to the restriction of k : whenever w depends on a discourse referent $d \neq k$ (or viceversa) so must k .

By applying these constraints, we obtain the following sets:

⁶Actually, we must also forbid those positive sets to combine with each other. This may be simply achieved by marking in some way every positive set we do not want to combine.

⁷Note that v is hidden, since *tourism centers* is a bare plural.

$$\begin{aligned}
P = & \{ \{A \dashrightarrow \text{Ctx}\}, \{y \dashrightarrow z\}, \{k \dashrightarrow w, v \dashrightarrow w, p \dashrightarrow w\}, \\
& \{k \dashrightarrow y, v \dashrightarrow y, p \dashrightarrow y\}, \{k \dashrightarrow z, v \dashrightarrow z, p \dashrightarrow z\} \} \\
N = & \{ \{y \dashrightarrow w \mid k \dashrightarrow w, v \dashrightarrow w, p \dashrightarrow w\}, \{w \dashrightarrow y \mid k \dashrightarrow y, v \dashrightarrow y, p \dashrightarrow y\}, \\
& \{w \dashrightarrow z \mid k \dashrightarrow z, v \dashrightarrow z, p \dashrightarrow z\} \}
\end{aligned}$$

The only acceptable reading is the one having all discourse referents linked to Ctx only. However, in order to identify it, we would need a semantic knowledge base from which we can infer, for example, that *several sectors* cannot depend on *his properties*, and so the arc $y \dashrightarrow z$ has to be removed from P .

1715 sentences included in TUT have been analyzed as example (11)⁸. The results are shown in Table 1. Sentences have been divided into four classes, depending on the number D of discourse referents they contain. For each class, the table shows the *average* of the number of positive/negative arcs generated by applying the three constraints discussed above.

$ D $	$0 < D \leq 2$	$2 < D \leq 5$	$5 < D \leq 10$	$10 < D \leq 20$
 Sentences 	462	643	516	94
 Arcs 	1.10	1.85	5.16	12.91

Table 1: TUT - Number of positive/negative arcs per discourse referents

The number of arcs indicated in Table 1 are very low. The reason for this is the frequent occurrence in the sentences of proper names and definites with no restriction. According to the discussion above, such NPs must be linked to Ctx only, but this is already handled by the positive arc $A \dashrightarrow \text{Ctx}$. In other words, proper names and definites with no restriction introduce new discourse referents but do not introduce new arcs, thus decreasing the average values.

More reliable results are reported in Table 2. The table shows the number of arcs per discourse referents without considering such NPs. The reader may see that in the worst cases the number of arcs can be still considered low.

⁸We excluded sentences including verbal ellipsis in that, according to [12], they can engender complex quantifier-scope ambiguities not currently handled in DTS. Anaphora have been ignored, in that those requiring a referent resolution enter Ctx only, while donkey sentences, as shown by [24], simply require copies of the referent, with equal dependencies.

$ D $	$0 < D \leq 2$	$2 < D \leq 5$	$5 < D \leq 10$	$10 < D \leq 20$
 Sentences 	985	574	153	3
 Arcs 	1.38	3.39	10.62	36

Table 2: TUT - Number of positive/negative arcs per discourse referents, without considering proper names and definitives with no restriction.

6 Conclusions

In this paper, we presented an extension of DTS where allowed/disallowed dependencies are explicitly specified by inserting new arcs called positive/negative arcs. We believe that positive/negative arcs provide a scalable and modular solution for the management of constraints coming from heterogeneous sources. For instance, we showed that positive/negative arcs needed to manage Nested Quantification, Island Constraints, and logical Redundancy may be independently defined. Unfortunately, positive/negative arcs lead to a computational complexity which is exponential in the worst cases. Nevertheless, we performed a rough experiment on the Turin University Treebank, whose results indicate that those worst cases are very rare in real sentences. It seems then that the computational complexity is not really a problem. In a real system, in case the number of possible readings is too high, the system may decide to inspect the context in order to detect allowed/disallowed dependencies to be removed or added in the representation, thus reducing the number of readings.

References

- [1] Alshawi, H. *The Core Language Engine*. Mit Press, Cambridge, UK, 1992.
- [2] Barwise, J. & Cooper, R. *Generalized Quantifiers and Natural Language*. *Linguistics and Philosophy*, 4(2), 159–219, 1981.
- [3] Beck, S. & Sauerland, U. *Cumulation is Needed: A Reply to Winter (2000)*. *Natural Language Semantics*, 8(4), 349–371, 2000.
- [4] Beghelli F., Ben-Shalom D., Szabolcsi, A. *Variation, Distributivity, and the Illusion of Branching* In A. Szabolcsi (Eds), *Ways of Scope Taking*, Kluwer:Dordrecht, 29–69, 2001.

- [5] Bos, J. *Predicate Logic Unplugged*. Proceedings of the 10th Amsterdam Colloquium. Amsterdam, The Netherlands, 133–142, 1996.
- [6] Bosco, C. *A grammatical relation system for treebank annotation*. Ph.D. thesis, Department of Computer Science, University of Turin, 2004.
- [7] Brinkmann, G. and McKay, B.D. *Posets on up to 16 Points*. Order, 19(2), 147–179, 2002.
- [8] Bunt, H. *Semantic Underspecification: Which Technique For What Purpose?*. In R. Musken and H. Bunt (Eds.), Computing Meaning. Kluwer, vol. 3, 2003.
- [9] Chaves, R.P. *Non-Redundant Scope Disambiguation in Underspecified Semantics*. Proc. of the 8th ESSLLI Student Session, 47-58, Vienna, 2003.
- [10] Copestake, A, Flickinger, D. and Sag, I.A. *Minimal Recursion Semantics. An introduction*. Research on Language and Computation, 3(2), 2005.
- [11] Ebert, C. *Formal Investigations of Underspecified Representations*. Ph.D thesis, Department of Computer Science, King's College London, 2005.
- [12] Egg, M. and Koller, A. and Niehren, J. *The Constraint Language for Lambda Structures*. Journal of Logic, Language and Information, 10(4), 2001.
- [13] Hobbs, J. R. and Shieber, S. *An Algorithm for Generating Quantifier Scoping*. Computational Linguistics: 13:47–63, 1987.
- [14] Joshi, A. K. and Kallmeyer, L. *Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG*. Research on Language and Computation, 1:3–58, 2003.
- [15] Joshi, A. K. and Kallmeyer, L. and Romero, M. *Flexible Composition in LTAG: Quantifier Scope and Inverse Linking*. In R. Musken and H. Bunt (Eds.), Computing Meaning. Kluwer, vol. 3, 2003.
- [16] Kallmeyer, L. & Romero, M. *Scope and Situation Binding in LTAG using Semantic Unification*. Research on Language and Computation, 6(1), 2008.

- [17] Koller A., Thater S. *Towards a redundancy elimination algorithm for underspecified descriptions*. Proc. of the 5th Int. Workshop on Inference in Computational Semantics (ICoS-5) Buxton, England, 2006.
- [18] Park, J. *Quantifier Scope and Constituency*. Proceedings of the 33rd Annual Meeting of the ACL. pp.205-212, 1995.
- [19] Reyle, U. *Dealing with ambiguities by Underspecification: Construction, Representation and Deduction*. Journal of Semantics, 13, 123–179, 1993.
- [20] Robaldo, L. *Dependency Tree Semantics*. Ph.D thesis, Department of Computer Science, Turin University, Italy, 2007.
- [21] Ross, J. R. *Constraints on Variables in Syntax*. Ph.D thesis, Massachusetts Institute of Technology, 1967.
- [22] Scha, R. *Distributive, Collective and Cumulative Quantification*. In J. Groenendijk, M. Stokhof (Eds.), Formal Methods in the Study of Language, Part 2, pages 483–512. Mathematisch Centrum, Amsterdam, 1981.
- [23] Schein, B. *Plurals and Events*. MIT Press, Cambridge, MA, USA, 1993.
- [24] Steedman, M. *The Grammar of Scope*, forthcoming. See Surface-Compositional Scope-Alternation Without Existential Quantifiers. *Draft 5.2, Sept 2007*. <ftp://ftp.cogsci.ed.ac.uk/pub/steedman/quantifiers/journal6.pdf>.
- [25] Willis, A. *An Efficient Treatment of Quantification in Underspecified Semantic Representations*. Ph.D thesis, University of York, 2000.
- [26] Willis, A. *NP Coordination in Underspecified Scope Representations*. Proc. of the 7th Workshop on Computational Semantics, Tilburg, 2007.