Computing Discourse Semantics: The Predicate–Argument Semantics of Discourse Connectives in D–LTAG

KATHERINE FORBES-RILEY
University of Pennsylvania

BONNIE WEBBER
University of Edinburgh

ARAVIND JOSHI
University of Pennsylvania

Abstract

D–LTAG is a discourse-level extension of lexicalized tree-adjoining grammar (LTAG), in which discourse syntax is projected by different types of discourse connectives and discourse interpretation is a product of compositional rules, anaphora resolution, and inference. In this paper, we present a D–LTAG extension of ongoing work on an LTAG syntax-semantic interface. First, we show how predicate-argument semantics are computed for standard, 'structural' discourse connectives. These are connectives that retrieve their semantic arguments from their D–LTAG syntactic tree. Then we focus on discourse connectives that occur syntactically as (usually) fronted adverbials. These connectives do not retrieve both their semantic arguments from a single D–LTAG syntactic tree. Rather, their predicate-argument structure and interpretation distinguish them from structural connectives as well as from other adverbials that do not function as discourse connectives. The unique contribution of this paper lies in showing how compositional rules and anaphora resolution interact within the D–LTAG syntax-semantic interface to yield their semantic interpretations, with multi-component syntactic trees sometimes being required.

1 INTRODUCTION

The notion that discourse relates to syntax and semantics in a completely different way than does the sentence seems strange when we consider examples such as (1), where an entire discourse is contained within a relative clause, exhibiting the same cohesive and argumentative connections characteristic of multi-sentential discourses such as (2).

(1)   Any farmer who has beaten a donkey but gone home regretting it
       and has then returned and apologized to the beast, deserves
       forgiveness.

(2)   A farmer beat a donkey but he went home regretting it. Then he
       returned and he apologized to the beast. As a result he deserves
       forgiveness.

   Although several discourse models provide a syntax and semantics
for discourse (Schilder 1997; Gardent 1997; Polanyi and van den Berg
1996; Scha and Polanyi 1988), their syntax and semantics differ from
what one finds at the sentence level. D-LTAG presents a model of
low–level discourse structure and interpretation that exploits the *same*
mechanisms used at the sentence level and builds them directly on top
of clause structure and interpretation.
   In this paper, we address the computation of discourse semantics in
D-LTAG. We start by introducing D-LTAG in section 2. Section 3 then
presents a D-LTAG extension of ongoing work on an LTAG syntax-
semantic interface, and shows how it is used to compute the predicate-
argument semantics for D-LTAG structural connectives. In section 4,
we describe how adverbial discourse connectives differ from structural
connectives, and from other adverbials, in terms of their predicate-
argument structure and interpretation. In section 5, we show how
their predicate-argument semantics is computed in D-LTAG's syntax-
semantic interface. In section 6, we compare the computation of
discourse semantics in D-LTAG with that in previous discourse
models. In section 7, we highlight unresolved issues arising due to the
on–going nature of this work, both at the discourse level and the
clause level.

2 D-LTAG: AN INTERMEDIATE LEVEL OF DISCOURSE REPRESENTATION

D-LTAG (Creswell *et al.* 2004; Webber *et al.* 2003; Forbes *et al.* 2003;
Creswell *et al.* 2003; Creswell *et al.* 2002; Forbes *et al.* 2001; Webber
et al. 1999b; Webber *et al.* 1999a; Webber and Joshi 1998) builds an
intermediate level of discourse structure directly on top of the clause.
D-LTAG's syntax is currently modelled using the structures and
structure-building operations of a lexicalized tree-adjoining grammar
(LTAG) (Joshi and Vijay-Shanker 1999; XTAG 2001), which is widely
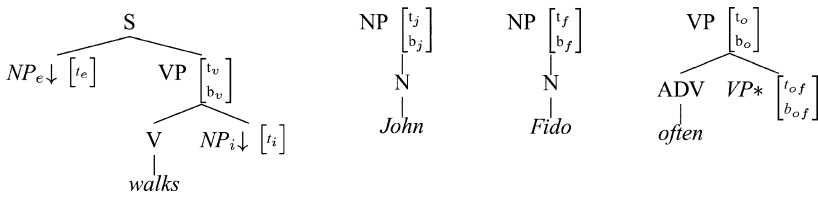used to model the syntax of sentences.

Figure 1   Elementary LTAG trees.

2.1 *An introduction to LTAG*

LTAG is an extension of a tree–adjoining grammar (TAG) (Joshi 1987) that includes lexicalization and unification-based feature structures, and represents the underlying structure of a surface string, as well as the string itself. LTAG consists of (i) a finite set of *elementary* trees, each associated with at least one lexical item (the *anchor*), and (ii) operations for combining them.

There are two types of elementary trees in LTAG: *initial* trees, which encode basic predicate-argument relations, and *auxiliary* trees, which encode recursion and must contain a non-terminal node, called the *foot* node and indicated by *, whose label matches the label of the root. The rightmost tree in Figure 1 is an auxiliary tree, all the others are initial trees.

Each elementary tree represents a domain of locality encoding the syntactic/semantic arguments of its anchor. An anchor may be associated with multiple elementary trees; each tree in this *family* reflects a minimal syntactic construction. For example, the verb *walk* may take the transitive tree shown in Figure 1 or an intransitive tree. Each node in an elementary tree is associated with at least one *feature structure*—a top feature structure in the case of substitution nodes, and both a top and bottom feature structure in the case of all other nodes. These for several nodes are shown in Figure 1. Feature unification then implements the dynamic specification of local constraints concerning how nodes interact during structure-building, including linguistic constraints on agreement, case, and tense (XTAG 2001).

There are two structure-building operations in LTAG that create complex trees, called *derived* trees: *substitution* and *adjunction*. Substitution consists of replacing the node marked ↓, as in the leftmost tree in Figure 1, with the tree being substituted. Only initial trees or trees derived from initial trees can be substituted, and the root node of the tree being substituted must match the label of the node being replaced. For example, we can substitute the NP tree anchored by *Fido* in Figure 1 for the internal NP argument (annotated here with *i*) in the tree
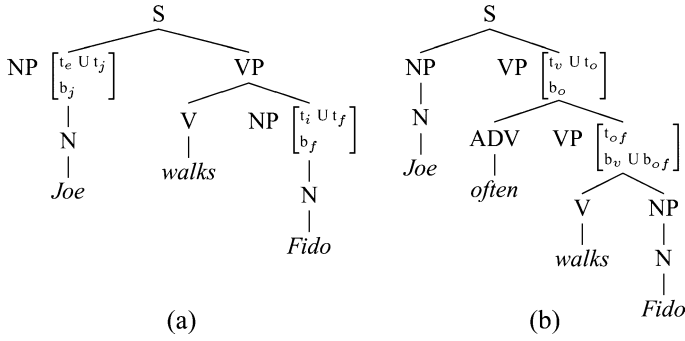
S
NP $\begin{bmatrix} t_e \cup t_j \\ b_j \end{bmatrix}$ VP
N   V   NP $\begin{bmatrix} t_i \cup t_f \\ b_f \end{bmatrix}$
Joe   walks   N
Fido

S
NP   VP $\begin{bmatrix} t_v \cup t_o \\ b_o \end{bmatrix}$
N   ADV   VP $\begin{bmatrix} t_{of} \\ b_v \cup b_{of} \end{bmatrix}$
Joe   often   V   NP
walks   N
Fido

(a)   (b)

Figure 2   LTAG derived trees after substitution and adjunction.

anchored by *walks*, and the NP tree anchored by *John* for the external NP argument (annotated here with *e*). In each case, the resulting node takes the union of the features of the involved nodes. In this union, the features common to both nodes must match. The remaining features are simply assigned. For example, if the top agreement features of the NP_e node are *3rd person, singular*, they must either match the top agreement features of the NP node in the *John* tree or be assigned. The NP node in the *Fido* tree will not carry case features, so accusative case will be assigned by the top features of the NP_i node. The derived tree resulting from these substitutions is shown in Figure 2(a).

Adjunction is restricted to non-terminal nodes not already marked for substitution, and builds a new tree from an auxiliary tree β and any other tree τ. To combine β and τ by adjunction, the root node of β must match the label of the node *n* in τ to which it adjoins. The root node of β is identified with *n*, their top features unify, and the bottom features of β are inherited; the subtree dominated by *n* is attached to the foot node of β, the bottom features of *n* unify with those of the foot node, and the rest of the tree that dominated *n* now dominates the root node of β. For example, the tree anchored by *often* in Figure 1 can adjoin to the VP node in Figure 2(a); the result, with relevant feature unifications, is shown in Figure 2(b).

Because derived trees such as in Figure 2 do not indicate what trees were combined or what operations were used to combine them, this information is recorded in an associated *derivation* tree. A node in a derivation tree is labelled with the elementary tree involved at that point in the derivation; the root of the derivation tree is labelled with the (unique) initial tree that is not substituted into any other tree in this derivation. In general, initial tree labels start with α, and auxiliary tree labels start with β. A tree address is also associated with each node in
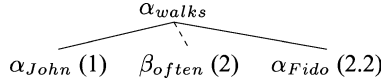
$$\alpha_{walks}$$

$$\alpha_{John}\ (1) \qquad \beta_{often}\ (2) \qquad \alpha_{Fido}\ (2.2)$$

Figure 3 LTAG derivation tree.

the derivation tree except the root. This is the address of the node in the *parent* tree at which the substitution or adjunction operation has occurred. The address of the root node in the parent tree is 0, the address of the kth child of that root is (k), and (p.q) is the address of the qth child of the node whose address is p. The derivation tree corresponding to Figure 2(b) is shown in Figure 3. (Dashed lines indicate adjunction, and solid lines, substitution.) The initial tree that roots this derivation is α_{walks}. α_{John} substitutes into α_{walks} at address (1) (NP_e), β_{often} adjoins into α_{walks} at address (2) (VP), and α_{Fido} substitutes into α_{walks} at address (2.2) (NP_i).

2.2   *An introduction to D-LTAG*

D-LTAG builds on the observation that discourse connectives function syntactically and semantically in the discourse akin to the way verbs function in the clause. In LTAG, verbs are viewed as *predicates*, which supply relations between entity interpretations of noun phrases (NPs) and/or clauses (Ss), producing a sentence interpretation. In D-LTAG, the predicates are discourse connectives, whose arguments are entity interpretations derived from either simple discourse units (e.g. individual clauses) or complex units composed of clauses and discourse connectives. We will use the notation 'D_u' for both, to indicate that these units are being analysed at the discourse level.

As in LTAG, each elementary tree in D-LTAG is anchored by at least one lexical item. Although the research is ongoing, our current view of the set of lexical items that convey discourse–level predication is that it includes subordinating and coordinating conjunctions, an empty connective at clause boundaries that are not marked by an explicit discourse connective, certain adverbs and prepositional phrases called *discourse adverbials*, and even some verb forms such as 'provided that' and imperative 'suppose'.

D-LTAG has both initial trees and auxiliary trees. Initial trees localize the predicate-argument dependencies of discourse connect-ives. For example, the tree family associated with subordinating con-junctions contains the two initial trees shown in Figure 4 with the anchor *although*, representing the basic syntactic alternations shown in

$$D_u$$

$$D_u\downarrow \quad although \quad D_u\downarrow \qquad\qquad although \quad D_u\downarrow \quad D_u\downarrow$$

Figure 4 D-LTAG initial trees for subordinating conjunctions.

(3a)–(3b) that subordinating conjunctions display relative to their arguments (Quirk *et al.* 1985).[1]

(3) a. John is very hard to find, although he is very generous.
     b. Although John is very generous, he is very hard to find.

In LTAG, subordinated clauses are treated as adjuncts because they are not part of the extended projections (e.g. argument structure) of the *verb* of the main clause. In D-LTAG, however, it is the extended projections of discourse connectives that are being modelled: for example, subordinating conjunctions relate two clausal interpretations to form a larger discourse unit and are thus represented as taking two structural arguments. (As such, we call them *structural connectives*.) As is also true of local dependencies at the clause level (e.g. *Apples, Bill says John may like*.), these dependencies can be stretched long-distance via adjunction of additional clause units, as shown in (4).

(4) a. Although John is very generous—
     b. if you need some money,
     c. you only have to ask him for it—
     d. he's very hard to find.

Two types of auxiliary trees are used in D-LTAG. The first is for structural connectives that convey a *continuation* of the prior discourse, e.g. a modification that serves to further a description of a discourse situation or of one or more entities within it (Webber *et al.* 2003). Auxiliary trees for *and* and the empty connective ε are shown in Figure 5. While both arguments in these trees come structurally, one comes via substitution, while the other comes via *adjunction* to a discourse unit in the prior discourse. Continuation is thus akin to the notion of *elaboration* in Knott *et al.* (2001), as exemplified in (5).

(5) John went to the zoo [*and*/.] H/he took his cell phone with him.

---

[1] In displaying D-LTAG trees, we suppress annotation of feature structures, as constraints on how discourse nodes interact during structure building have not yet be exhaustively studied (cf. Section 5.4,7). Empirical evidence for such constraints may come from the Penn Discourse Treebank (Miltsakaki *et al.* 2004; Prasad *et al.* 2004) as well as from theoretical studies.

Figure 5 D-LTAG auxiliary tree for *and* and ε.

More generally, the relations supplied by *and* or ε in these examples are syntactically under-specified, analogous to the syntax of noun–noun compounds, and they can be represented in the lexicon as semantically underspecified.[2] In many instances there may be additional inferred relations between the incoming unit and the prior discourse over and above simple elaboration (indicated here by coordination), which may be represented as semantic features in the final discourse structure. For example, in addition to the simple elaboration relation that ε supplies between the interpretations of *John came home late* and *Mary berated him* in (6), a temporal relation will be inferred between these two clauses and a causal one as well. While these additional inferences are critical to a full account of discourse meaning, because they can be canceled,[3] we do not consider such inferences to be part of compositional semantics and hence do not discuss them within the syntax-semantic interface we present here for D-LTAG.

(6)   Yesterday, John came home late [*and*/.] Mary berated him.

Where conjunctions convey more than continuation, they are represented with initial trees. For example, *so* conveys a *result* relation, as in 7:

(7)   You didn't eat your spinach *so* you won't get dessert.

The second type of auxiliary D-LTAG tree, shown in Figure 6, is used to represent discourse adverbials such as *then, otherwise, (in the) meanwhile*, etc. Like *structural connectives*, discourse adverbials relate the interpretations of two simple or complex discourse units. However, unlike structural connectives, they retrieve only *one* argument structurally—from the discourse unit they modify. Again there are two

---

[2] E.g. Knott (1996) treats the lexical semantics of connectives as feature inheritance.
[3] Inference of a causal relation can be canceled as in (i). The well-known example in (ii) shows that even an inference of temporal order can be cancelled as well.
   (i) John came home late and Mary berated him. But it was because he had forgotten to stop at the grocery store, as he had promised.
   (ii) Mary got married, and she gave birth to a son, but not in that order.
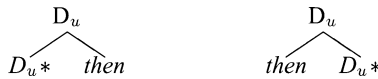
$$D_u$$
$$D_u*\quad then$$

$$D_u$$
$$then\quad D_u*$$

Figure 6   D-LTAG auxiliary trees for discourse adverbials.

trees in this tree family too, due to the fact that discourse adverbials can appear S-initial and S-final.[4]

Except for their node labels, the D-LTAG trees for discourse adverbials are identical to LTAG trees for S-modifying adverbials; adjunction represents the optional continuation that all adverbials make. But since discourse adverbials retrieve only one of their arguments structurally (from the discourse unit at the foot of the adjunction site D_u*), D-LTAG must account for where the other argument comes from. Elsewhere we have argued (Webber *et al.* 2003) that it is provided *anaphorically*.

This distinction between structural and anaphoric connectives in D-LTAG is based on considerations of computational economy and behavioural evidence from cases of multiple connectives (Webber *et al.* 1999b). In (8), *because* encodes a causal relation between the event of Sally raising ire and Sally enjoying cheeseburgers, while *nevertheless* encodes a violated expectation relation between Sally enjoying cheeseburgers and Sally subscribing to the *Vegetarian Times* (Creswell *et al.* 2002).

(8)   a.   Sally subscribes to *Vegetarian Times*.
  b.   Lately, she's raised the ire of her vegan friends
  c.   *because* she *nevertheless* enjoys the occasional bacon cheese-burger.

To model both *because* and *nevertheless* as structural connectives would create a directed acyclic graph with crossing dependencies, which goes beyond the computational power of LTAG and creates a completely unconstrained model of discourse structure (Webber *et al.* 2003). However, investigations into the behaviour of discourse connectives reveal that while subordinating and coordinating conjunctions are constrained to take adjacent discourse segments of like syntactic type as their arguments, discourse adverbials seem to share many properties with anaphora. For example, discourse adverbials can

---

[4] While both discourse adverbials and subordinate clauses can appear clause-internally, D-LTAG currently does not posit separate trees for them. Rather, our D-LTAG parser (Forbes *et al.* 2003) extracts them from clause-internal position, leaving a trace for information-structure analyses and representing them with an S-initial tree.

(like NP anaphora) take their prior argument intra-sententially as in (9), where embedded *nevertheless* relates the interpretation of the matrix clause to the interpretation of the relative clause (Webber *et al.* 2003).

(9) Many people who have developed network software have *nevertheless* never gotten very rich. (i.e. *despite having developed network software*)

Another feature that discourse adverbials share with anaphoric NPs is that they can access inferred entities as well as ones introduced explicitly. For example, in (10a) the discourse adverbial *otherwise* can access the inferred condition of *if the light is not red*. This material is not available to structural connectives: that is, *or* in (10b) can only access what is to its left—the consequent clause (*stop*) or the antecedent–consequent pair.

(10) a. If the light is red, stop. *Otherwise* go straight on.
      b. If the light is red, stop, *or* go straight on.

Furthermore, a property of all English structural predicates is that they do not allow crossing of predicate-argument dependencies.[5] For example, while example (4) above is interpretable as an embedded *if S1, S2* construction within an *although S1', S2'* construction, crossing these dependencies as in (11) is either uninterpretable or at least yields a different interpretation; the dependencies in the original constructions are lost.

(11) a. Although John is very generous—
      b. if you need some money—
      d. he's very hard to find—
      c. you only have to ask him for it.

However, discourse adverbials *do* allow crossing dependencies, as shown in (12) (Webber *et al.* 2003). For *then* in (d) to get its first argument from (b), it must cross the structural connection between the clauses in (c) and (d) that are related by *because*. Of course, as Webber *et al.* (2003) note, it is common for anaphora to show crossing dependencies (e.g. *John$_1$ told Mike$_2$ he$_1$ would meet him$_2$ later.*). Whether *all* discourse adverbials allow crossing dependencies is still an unanswered empirical question. Forbes *et al.* (2003) outline an empirically-based approach to modeling discourse connectives, in which

---

[5] Crossing is possible in Dutch clauses, e.g. the Dutch version of 'The rat the cat the dog chased ate died.' But there is no evidence, even in Dutch, of crossing structural connectives.
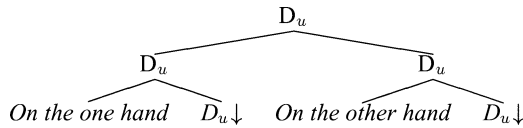
$$D_u$$

$$D_u \qquad\qquad D_u$$

On the one hand $D_u\downarrow$ On the other hand $D_u\downarrow$

Figure 7   D-LTAG initial tree for parallel adverbial constructions.

predicate-argument dependencies are defined structurally until contra-
dicted by corpus examples.

(12)   a.   John loves Barolo.
     b.   So he ordered three cases of the '97.
     c.   But he had to cancel the order
     d.   because then he discovered he was broke.

There is one exception to discourse adverbials being associated with
auxiliary trees, and that is for *parallel* constructions, i.e. paired
connectives which together assert *disjunction* ('either … or'), *addition*
'not only … but also'), *concession* ('admittedly … but'), and *contrast*
('on the one hand … on the other hand'). Because of the interpreted
inter-dependency of the connectives in these constructions, they are
modeled with *initial* trees, as in Figure 7.

Parallel constructions have multiple anchors, but one or the other
anchor may not be realized: for example, 'or' without 'either', 'not
only' without 'but also', or 'on the other hand' without 'on the one
hand'. One might argue in the latter case that an auxiliary tree should
be used instead of an initial tree, accessing the non-structural argument
of 'on the other hand' anaphorically. But this is best viewed as an
empirical question whose answer depends on where the other
argument is found. Data from the annotated corpus that will comprise
the Penn Discourse Treebank (Miltsakaki *et al.* 2004; Prasad *et al.*
2004) will help answer this question.

We conclude this section by making clear our assumptions about
discourse structure. While we assume that discourse is primarily tree-
structured, there are a few simple exceptions that require simple DAGs
(directed acyclic graphs). First, as we note in (Webber *et al.* 2003),
we must allow for examples such as (13) from Bateman (1999):

(13)   … (vi) The first to do that were the German jewelers, (vii) in
       particular Klaus Burie. (viii) And Morris followed very quickly
       after, (ix) using a lacquetry technique to make the brooch, (x)
       and using acrylics, (xi) and exploring the use of colour, (xii)
       and colour is another thing that was new at that time …
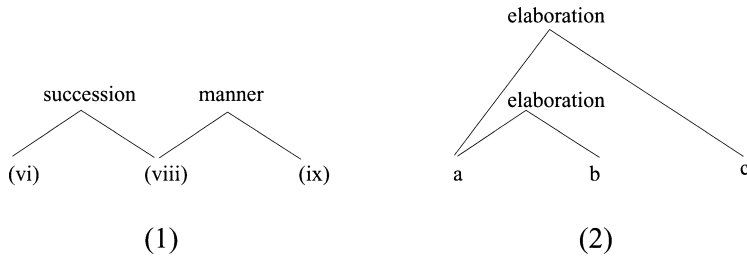
Figure 8 Simple multi–parent structure for (1) 'transitional' clause, (2) clause with independent post-modifiers.

Bateman analyses this example as containing a *transitional clause* (here, clause viii) that stands in one relationship (here *succession*, aka *sequence*) to the clause preceding it (here, clause vi) and another relationship (here, *manner*) to the clause following it (here, clause ix). The result illustrated in Figure 8 (1) is a simple DAG in which a node has more than one parent.

Second, we may want to allow for a discourse structure that corresponds in clause-level syntax to a head with independent post-modifiers, as in (14). In this case, both post-modifiers (b) and (c) relate as independent elaborations of the main clause (a), as illustrated in Figure 8 (2).

(14) a. John loves Barolo.
      b. He first tasted it in 1992.
      c. According to Hugh Johnson, it's one of Italy's supreme reds.

This is again a simple DAG, in which a node has more than one parent. Both cases would require that D–LTAG be extended to allow the same element to substitute into more than one tree (or to allow an element to both substitute into one tree and serve as an adjunction site for an auxiliary tree).

But neither simple DAG (Figure 8 (1) or (2)) has crossing structural dependencies: While the structure of Figure 8 (2) can be drawn in such a way that the arc from (a) to (b) crosses the arc from (a) to (c), simply dropping the arc from (a) to (b) lower down in the figure removes it. So, contra (Danlos 2004), this type of structure has no intrinsic crossing structural dependency, and we continue to believe that the cost of moving to *arbitrary* DAGs for discourse structure is too great to be taken lightly. In fact, the one discourse structure given in (Danlos 2004) of a DAG with real crossing dependencies is one that she herself excludes as an analysis as violating what she calls a *left1-right2 principle*.

While the syntax-semantic interface given in section 3 does not cover the structures given in Figures 8, it is because the adjoining and substitution operators in TAG do not let us produce them. This would have to be addressed in subsequent work.

We now turn to our proposed syntax-semantic interface for D-LTAG.

3   A SYNTAX–SEMANTIC INTERFACE (SSI) FOR D-LTAG: PART 1

3.1   *Starting point: SSIs for LTAG*

Joshi and Vijay-Shanker (1999) were the first to propose an SSI for LTAG. They suggested that if substitution and adjunction are viewed as *attachments* of one tree to another tree, then a syntactic derivation consists of an unordered set of attachments whose semantics can built monotonically and described by a flat semantic representation. For example, the flat semantic representation corresponding to Figure 2(b) might be as in (15), where *i1, i2* denote individuals and *e1* denotes an event.

(15)   *john(i1)* $\wedge$ *Fido(i1)* $\wedge$ *walks(e1, i1, i2)* $\wedge$ *often(e1)*

The problem that Joshi and Vijay-Shanker (1999) note, is that adjunction does not preserve monotonicity with respect to *immediate dominance*.[6] Although syntactic under-specification of immediate dominance would remove this non–monotonicity, the tacit assumption in many LTAG formalisms is that the nodes along the spine of a tree (the path from the root to the anchor) are not distinguished *semantically*. As an alternative to this syntactic under-specification, Joshi and Vijay-Shanker (1999), Candito and Kahane (1998), and Schabes and Shieber (1994) have argued for defining compositional semantics with respect to attachments in the derivation tree (Figure 3), rather than the derived tree (Figure 2). There are two reasons for this. First, the predicate-argument structure of a lexical item is represented only in the derivation tree (not in the derived tree). Second, the derivation tree records the different elementary trees involved in the derivation and distinguishes argument substitution of arguments from optional modification (adjunction).

Computing compositional semantics on the LTAG derivation tree is formalized in Joshi *et al.* (2003), Kallmeyer (2002), Kallmeyer and

---

[6] For example, in Figure 2(a), the VP that is immediately dominated by S immediately dominates V, but in Figure 2(b), after α_{often} is adjoined, this is no longer the case.

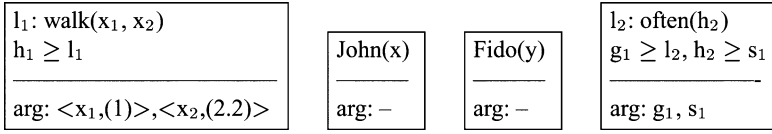| l_1: walk(x_1, x_2) $h_1 \geq l_1$		John(x)	Fido(y)	l_2: often(h_2) $g_1 \geq l_2$, $h_2 \geq s_1$
arg: <x_1,(1)>,<x_2,(2.2)>		arg: –	arg: –	arg: g_1, s_1

Figure 9 Semantic representations of the elementary trees in Figure 1.

Joshi (1999). To represent scope ambiguities in quantifier adjuncts, a restricted form of multi-component TAGs is used. The flat semantics uses ideas from Minimal Recursion Semantics (Copestake *et al.* 1999) and Hole Semantics (Copestake *et al.* 1999; Bos 1995) and consists of typed lambda expressions, scope constraints, and argument variables. We illustrate their approach using the sentence *John often walks Fido*, with semantic representations for α_{walks}, α_{John}, α_{Fido} and β_{often} shown respectively in Figure 9.

As shown in the lower half of each box, substitution argument variables are explicitly linked to addresses in the elementary tree, e.g. the pair $\langle x_1, (1) \rangle$ indicates that the individual argument variable x_1 is linked to address (1), and will obtain the value of whatever free individual variable substitutes in there (i.e. x in *John(x)*). Lambda expressions (formulas) can also contain propositional labels l_i and propositional argument variables s_i (whose values are propositional labels), as well as holes h_i (meta-variables for proposition labels) and hole variables g_i (whose values are holes). Holes and labels underpin an under-specified semantic representation that allows for scope ambiguities whose ordering is constrained by the scope constraints. In α_{walk}, for example, there is a hole above the proposition label l_1 which may produce scope ambiguity constrained as $h_1 \geqslant l_1$. Scope disambiguation occurs when the semantic representations are combined. If no intervening holes or labels are introduced, l_1 would be disambiguated to h_1. However, as shown, quantifiers and adverbs can introduce intervening holes and labels: β_{often} introduces the label l_2, the hole h_2, and the hole variable g_1.

When the derivation tree is built from its constituent trees, it indicates how the corresponding semantic representations combine: when a tree is substituted, its value is applied to the argument variable paired with the position at which it attaches. When a tree is adjoined, however, its argument variables are added to the representation and are assigned the values at the position where it attaches. The derivation tree in Figure 3 indicates how the semantic representations in Figure 9 combine: the value of x_1 comes from α_{John} when it is substituted, and

$$l_1: walk(x, y), John(x), Fido(y), l_2: often(h_2)$$
$$h_1 \geq l_1, h_1 \geq l_2, h_2 \geq l_1$$

Figure 10   Final semantic representation of *John often walks Fido*.

the value of x_2 comes from α_{Fido} when it is substituted. The values of s_1 and g_i come from α_{walk}, where β_{often} adjoins. Because s_1 is a propositional argument variable, it is assigned to l_1, and because g_1 is a hole variable, it is assigned to h_1. These assignments are reflected in Figure 10, which shows the semantic representation for *John often walks Fido*.

Scope disambiguation is then a bijection (one-to-one mapping) from holes to labels that obeys the scope constraints, where: (i) $\geqslant$ is a partial order, (ii) $l_1 \geqslant l_2$ if $l_1 \geq l_2$ is a constraint, (iii) $l_1 \geqslant l_2$ and $l_1 \neq l_2$ if l_2 appears in the formula labelled l_1, (iv) $l_1 \not\geqslant l_2$ and $l_2 \not\geqslant l_1$ if l_1 and l_2 are different arguments of the same predicate (Joshi *et al.* 2003). From Figure 10, we know $l_2 > h_2$, because h_2 appears inside l_2. Because we also see $h_2 \geqslant l_1$, we therefore know $l_2 > l_1$. We further know $h_1 \neq l_1$, because if $h_1 = l_1$ then $l_1 \geqslant l_2$ and we already know $l_2 > l_1$. Therefore, the only possible disambiguation is: $h_1 = l_2$ and $h_2 = l_1$. This yields (16).

(16)   *John(x3)* $\wedge$ *Fido(x_4)* $\wedge$ *walk(x_3, x_4)* $\wedge$ *often(walk(x_3, x_4))*

3.2   *Interlude: SSIs for discourse*

It is only in an approach to discourse which has a syntax that is distinct from its semantics—e.g. Schilder (1997), Gardent (1997), Polanyi and van den Berg (1996), Scha and Polanyi (1988), Danlos (2004)—that the concept of a syntax-semantic interface is relevant. For example, rhetorical structure theory (Mann and Thompson 1988) does not have a semantics that is distinct from its syntax: A (structural) relation is taken to hold between two discourse units because of an intentional-level or an informational-level relationship that holds between them. (The relation that is taken to hold between them may specify that one be treated as the *nucleus* or more important element of the relation and the other, as a *satellite*.) Thus there is no need for a syntax-semantic interface for RST.

As we noted in section 2.2, we take the job of the syntax–semantic interface to be that of providing a basic compositional semantics for a structure. Additional inferences may then be drawn, providing additional elements of semantics—such as the inference from *John often*
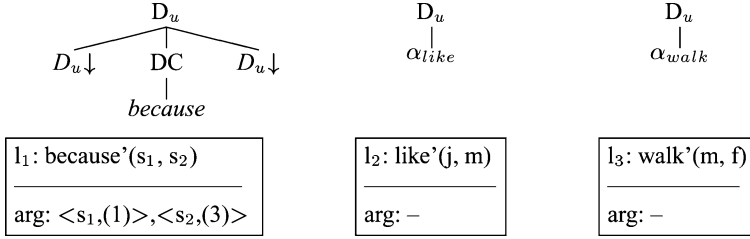
Figure 11   D-LTAG elementary trees and their semantic representations for (17).

walks Fido that John doesn't always walk Fido—but such inferences may not always be drawn, or they may be subject to later cancellation if they are. In any case, we do not take it to be the job of a syntax–semantic interface to project them.

3.3   *An SSI for D-LTAG: Structural connectives*

Because in D-LTAG, discourse connectives are *predicates*, D-LTAG can build their compositional semantics using the *same* semantic mechanisms that are used to build the compositional semantics for clause-level predicates. Here we show how the LTAG syntax-semantic interface can be extended to construct compositional semantics for discourses built from structural connectives. In section 5, we will further extend this approach to discourse containing anaphoric connectives.

Consider first the simple discourse in (17), where two clauses are linked by the subordinating conjunction *because*.

(17)   John likes Mary because she walks Fido.

The elementary D-LTAG trees for (17) are shown in Figure 11; below each tree its semantic representation is shown. Hereafter we represent atomic discourse units (clauses) using the LTAG symbol for initial trees subscripted by the name of the main verb (e.g. α_{like}). In fact, a benefit of D-LTAG is that it parses discourse on top of the clause level parse; thus each atomic discourse units is itself a complex tree (Forbes *et al.* 2003). Recall that D_u is a generic discourse unit root label for both atomic and complex constituents.

We extend to discourse the semantics outlined in section 3.1, by associating propositional labels, e.g. l_i, with the semantics of discourse units, both atomic and complex, and by using propositional variables, e.g. s_i, as discourse unit variables, whose values are these labels. Because
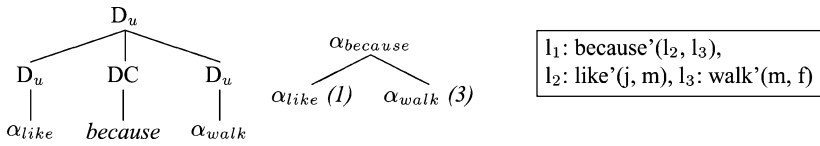
D_u

D_u DC D_u

α_{like} *because* α_{walk}

$\alpha_{because}$

α_{like} *(1)* α_{walk} *(3)*

l_1: because'(l_2, l_3),
l_2: like'(j, m), l_3: walk'(m, f)

Figure 12   D–LTAG derived and derivation trees and semantic representation for (17).

we are only dealing with structural connections here, we will not use holes until section 5. We also employ a simplified representation of clause semantics, akin to the treatment of NPs in Kallmeyer (2002), in which the values of VP arguments have already been computed. For example, we use $l_2 : like'(j, m)$ for the semantic value of *John likes Mary*. As shown, *because* takes two substitution arguments that are associated with addresses in its elementary tree.

Figure 12 shows the derived and derivation trees that result from substituting α_{like} and α_{walk} into $\alpha_{because}$. (Initial trees with discourse connective anchors are labeled with α subscripted by the name of the discourse connective (e.g. $\alpha_{because}$), and auxiliary trees with discourse connective anchors are labeled by β subscripted by the name of the connective (e.g. β_{and}).) The final semantic representation for this discourse is shown beside the trees.

As at the clause level, combining semantic representations at the discourse level consists of building the union of the semantic representations of the elementary trees involved in the derivation and assigning values to argument variables, where the derivation tree indicates how argument variables get values. The derivation tree in Figure 12 indicates that the s_1 argument variable in the *because* elementary tree is assigned the label of α_{like}, which substitutes at position (1). Similarly, s_2 is assigned the label of α_{walk}.

We now analyse the discourse in (18), which is interpreted as simple continuation (using coordination). The elementary D-LTAG trees for (18) are shown in Figure 13 beside their semantic representations. Here only the substitution argument of *and* (s_4) is linked to an address; after adjunction, the argument variable will be added to the representation and assigned the value at the position where it attaches. Figure 14 shows the derived and derivation trees that result from substituting α_{kiss} into β_{and} and adjoining β_{and} to α_{see}, along with the semantic representation that results. The derivation tree shows that s_4 is assigned the label of α_{kiss}, and s_3 is assigned the label of α_{see}.

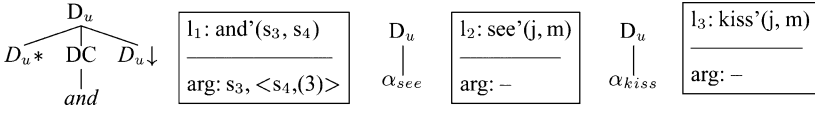(18)   John saw Mary and he kissed her.

Figure 13 D-LTAG Elementary trees and their semantic representations for (18).
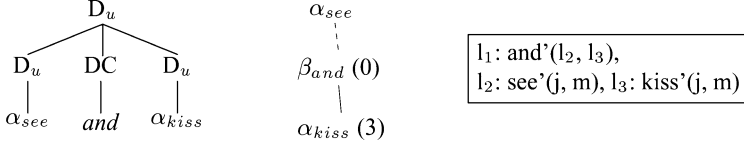

Figure 14 D-LTAG derived and derivation trees and semantic representation for (18).

We now consider the following more complex pair of discourses:

(19)   a.   (Who is happy and who is sad?)
       b.   [*Mary is happy* because *she found a job*] and [John is sad].
(20)   a.   (What are two reasons for Mary being happy?)
       b.   *Mary is happy* because [*she found a job*] and [*she likes it*].

In both, the arguments of *because* are italicized, and the arguments of *and* are bracketed. To make it clear that these are the interpreted semantic arguments of each connective, we include a contextual question in (19a) and (20a).

Figure 15 shows the derived and derivation trees and the semantic representation for (19b) that result from substituting α_{happy} and α_{find} into $\alpha_{because}$, adjoining β_{and} to the *root* of the *because* tree, and substituting α_{sad} into β_{and}. This is the expected interpretation; by adjoining to the root of the *because* tree, the value of the *and* adjunction argument variable becomes the label l_1, corresponding to the entire *because* tree.

However, this process yields an unexpected interpretation for the minimally different discourse in (20b). Figure 16 shows the derived and derivation trees and the semantic representation that result from substituting α_{happy} and α_{find} into $\alpha_{because}$, adjoining β_{and} to the leaf (α_{find}) of the result, and substituting α_{like} into β_{and}.

The natural interpretation of (20) is that Mary being happy is caused by both her finding a job *and* her liking it. However, this is not what is represented in Figure 16, because the *and* adjunction is not retrieved from the argument position (3) of *because* in the derivation tree;
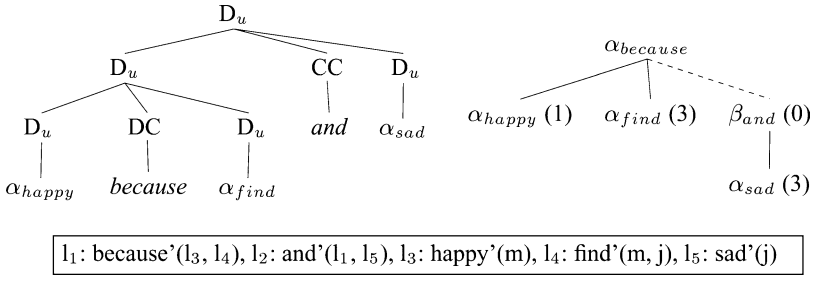
l_1: because'(l_3, l_4), l_2: and'(l_1, l_5), l_3: happy'(m), l_4: find'(m, j), l_5: sad'(j)

Figure 15 D-LTAG derived and derivation trees and semantic representation for (19).

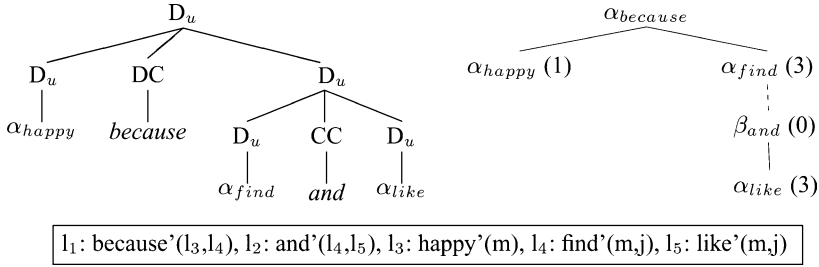l_1: because'(l_3,l_4), l_2: and'(l_4,l_5), l_3: happy'(m), l_4: find'(m,j), l_5: like'(m,j)

Figure 16 D-LTAG derived and derivation trees and semantic representation for (20).

the value of this argument position is, rather, only α_{find}.[7] The assigned interpretation does not even seem coherent.

In fact, the same problem arises in LTAG due to clause-level conjunction, such as exemplified in (21). The unit *Mary and Sue* is built by adjoining an NP-coordination *and* tree to α_{mary} and substituting α_{sue} into the *and* tree. The NP-coordination *and* tree is identical to the auxiliary *and* tree in D-LTAG, except for its non–terminal nodes labels (NP, CC). However, the compositional semantics for (21) would be *likes*(*j*, *m*) because α_{mary} is what substitutes into α_{likes} (at position (2.2)).

(21)   John likes Mary and Sue.

While one solution is to make the *and* tree into an initial tree (in both LTAG and D-LTAG), so that it substitutes directly into the relevant argument position, the problem would remain for adjunctions more generally. Moreover, the *and* tree for VP coordination could not

[7] Note that the problem becomes enormously complex when we consider the current D-LTAG parser's 'lowest adjunction' default for the null connective (Forbes *et al.* 2003).

be an initial tree, because VP trees do not have a root label VP, but rather S.[8]

An alternative solution involves a restricted use of *flexible direction of composition*, such as employed in Joshi *et al.* (2003). Consider the context-free grammar (CFG) rule A → BC. To produce A, we can view B as a function and C as its argument, or C as the function and B as its argument. Because CFGs provide *string* rewriting rules in which function and argument are 'string-adjacent', this use of flexible composition affects neither the weak (set of strings) nor the strong (set of derivation trees) generative capacity of the CFG. A TAG, however, provides *tree* rewriting rules, in which function and argument are complex topological arguments (trees) that are 'tree-adjacent'. Joshi *et al.* (2003) formally define flexible composition in a TAG as follows: if a tree *t* composes with a tree *u* then *u* must be an elementary tree. This assures 'tree-locality' in the composition. If both *t* and *u* are elementary trees, composition can go in either direction. If both *t* and *u* are derived trees, composition is simply blocked. If only one of the trees is elementary, then the other tree can compose into it but not vice versa. As defined, flexible composition allows the derivation tree to be traversed in a flexible manner, starting at any node. Note however that several derived trees can composed into an elementary tree simultaneously, so as not to exclude standard TAG derivations. In general, flexible composition can increase the generative capacity of a TAG. However, Joshi *et al.* (2003) adopt a *restricted use* of flexible composition, which will also be used in this paper. This restricted approach maintains the constraints on composing trees as just defined, but further requires a *bottom up* traversal of the derivation tree. Essentially, this restricted approach provides a specific way of ordering the derivations in a TAG, without increasing its generative capacity.

We return now to our problematic example (20). We have already seen in Figure 16 how traversing this derivation tree from top-down yields an unwanted semantic representation for this discourse, because the right semantic argument of $\alpha_{because}$ is incorrectly assigned to the semantic label of α_{find} (l_4). If, however, we adopt the restricted use of flexible composition in Joshi *et al.* (2003), we get all and only the intended interpretation. Using this approach, the derivation of (20) proceeds as follows: first, α_{like} substitutes into β_{and}, and the semantic label for α_{like} (l_5) is assigned to the right semantic

---

[8] VP coordination can however be viewed as S coordination where two subject nodes are identified, or a VP coordination tree can be constructed dynamically.

argument of β_{and} (s$_4$). This derived tree then adjoins to α_{find}, and the semantic label for α_{find} (l$_4$) is assigned to the left semantic argument of β_{and} (s$_3$). This derived tree substitutes into $\alpha_{because}$, and the semantic label for β_{and} (l$_2$) is assigned to the right semantic argument of $\alpha_{because}$ (s$_2$).[9] α_{happy} also substitutes into $\alpha_{because}$, and the semantic label of α_{happy} (l$_3$) is assigned to the left semantic argument of $\alpha_{because}$ (s$_1$). The resulting (intended) interpretation, shown in (22), has 'l$_1$: because' (l$_3$, l$_2$)', where l$_2$ is the semantic label for the complex tree rooted by β_{and} whose own semantic arguments are the interpretations of α_{like} and α_{find}.

(22) $\boxed{\text{l}_1\text{: because'}(\text{l}_3, \text{l}_2), \text{l}_2\text{: and'}(\text{l}_4, \text{l}_5), \text{l}_3\text{: happy'}(m), \text{l}_4\text{: find'}(m, j), \text{l}_5\text{: like'}(m, j)}$

In section 5, we will illustrate how Joshi *et al.* (2003)'s restricted approach to flexible composition applies to discourses whose representation involves the use of multi-component TAGs (MC-TAGs) (Weir 1988).

4 DISCOURSE ADVERBIALS AS DISCOURSE CONNECTIVES

The syntax–semantic interface we have just considered applies to structural connectives. But other discourse connectives, such as *instead*, *otherwise* and *finally*, are adverbials. Before we discuss how adverbial discourse connectives are treated in D-LTAG's syntactic–semantic interface (section 5), we first address the question of why these *discourse adverbials* (as in example (23b)) are interpreted with respect to the discourse context, while other adverbials, which we call *clausal adverbials* (as in example (23a)), are interpretable with respect to just their matrix sentence.

(23) a. *Probably/In most small towns/Usually*, people are friendly.
      b. *As a result/Consequently/In addition*, people are friendly.

Differences between these types are not attributable to syntax. As shown in Figure 17, both adjoin to the S node and the result is still an S.

This common syntax was used to extract the data for our analysis: using 'tgrep', S-initial, S-adjoined ADVPs and PPs were collected from the Penn Treebank I parses of the Brown and WSJ corpora

---

[9] Note that l$_4$ and l$_5$ are already assigned as the semantic arguments of β_{and} and thus are not candidates for assignment to s$_2$ here, since we assume a one-to-one mapping between propositional variables and labels.

```
              S                                    S
        ┌─────┴─────┐                        ┌─────┴─────┐
       PP           S                       PP           S
    ┌───┴───┐    ┌───┴───┐               ┌───┴───┐    ┌───┴───┐
    P      NP   NP      VP               P      NP   NP      VP
    │       │    │       │               │       │    │       │
  After  a rain  John  smiles            As   a result John  smiles
         shower
```
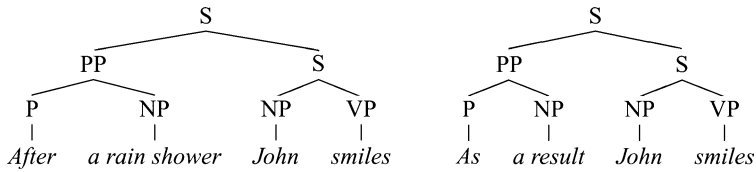
Figure 17   S–Adjoined clausal adverbials and discourse adverbials.

(Penn–Treebank 1993). Although discourse adverbials can occur in other positions, most also occur S–initially. Forbes *et al.* (2003) suggest that these other positions serve an information–structuring purpose. And although other phrasal categories can function as adverbials, ADVP and PP are among the most commonly cited discourse adverbials (Knott 1996).

Total tokens and types in our data–set are shown in Table 1.[10] Counts of those occurring 12 or more times are also shown ('>11'). This comparison shows that while there are more high frequency ADVP tokens than PP tokens, there are many more PP tokens and types. This is due to the wide variability in the internal PP argument; over half of the PP tokens occur only once.

We argue that it is in their semantics that significant differences appear between discourse and clausal adverbials. The semantics of most discourse adverbials has not previously been addressed. Most post–generative investigations of adverbials (Verspoor 1997; Alexiadou 1997; Jackendoff 1990; Gawron 1986; Ernst 1984; McConnell–Ginet 1982) treat clause–level syntactic and semantic issues: Where discourse adverbials are mentioned at all, they are called 'conjunctive adverbials' and are specified as the domain of discourse research. Most discourse research, however, has focused on the semantics or pragmatics of only a few discourse adverbials (Kruijff–Korbayova and Webber 2001; Lagerwerf 1998; Jayez and Rossari 1998; Hovy 1995; Rooth 1995; Redeker 1990; Schiffrin 1987; van Dijk 1979). Although Knott's (1996) lexical semantics can represent all or part of the meaning of a wide variety of discourse adverbials, his experimental methodology for identifying them relies entirely on human judgments and does not control for factors such as intonation and pragmatic implicature.

We argue that the distinguishing features of a discourse adverbial are the number and interpretation of its semantic arguments. These are generally described rather vaguely in the adverbial literature. For example, Ernst (1984) makes informal references to a variety of interpretations of the modified element, including *event, situation, state*

---

[10] Due to annotation errors, these counts are approximate. Also, topicalized non–S modifiers are not excluded because their underlying position is not represented in the Penn Treebank.

Table 1 S-initial, S-adjoined ADVP & PP adverbials in WSJ & Brown Corpora.

Adverbial category	Total tokens	Total types	>11 tokens	>11 types
ADVP adverbial	3528	849	2559	53
PP adverbial	10295	7424	1738	49

of affairs, *proposition*, *mental state*. Moore (1993) is more precise: his formal description of adverbial modification admits events and situations into the semantic domain of entities and represents them as semantic arguments of verbs and true propositions, respectively. However, discourse deixis research provides evidence that a much wider range of interpretations is involved in adverbial modification.

Briefly, the term 'discourse deixis' applies to the use of demonstrative pronouns or adverbs to refer to the interpretation of non–NP constituents, such as shown in (24a)–(24c), where the referent (derived from the bracketed material) is the interpretation of a clause (24a), a sequence of clauses (24b), or a VP (24c). Early studies of this phenomena relate it to another use of demonstratives, shown in (24d), where the referent is not in the discourse at all, but rather is in the spatio-temporal context. This use is called *deictic*, a Greek term meaning 'pointing' or 'indicating' (Lyons 1977).

(24) a. [John took Biology 101.] *That* means he can take Biology 102.
  b. [I woke up. I ate breakfast. I watched TV.] *That*'s all I did today.
  c. John [travels a lot]. Mary does *that* too.
  d. '*That*'s not nice, Billy!', the mother said, when Billy hit her.

The nature of such interpretations goes under the rubric *abstract object* (AO), to distinguish them from *concrete* or *physical objects*. Included among Asher's (1993) taxonomy of AOs are entities such as *events*, *states* and *facts*, which can all be denoted by or inferred from both NP and non–NP constituents. Other researchers (Vendler 1967; Webber 1991; Dahl and Hellman 1995; Forbes 2003) have observed that additional AOs, including *descriptions*, *beliefs*, *speech acts*, *textual objects*, *defeasible rules* and *discourse relations* can also be the referents of discourse deixis.

We argue that AOs provide an appropriate way of understanding the semantic arguments of adverbials. We argue that the range of semantic objects evoked when an adverbial modifies a clause coincides with the

Table 2 AO interpretations via discourse deixis and adverbial modification.

Abstract object	Discourse deictic	PP adverbial	ADVP adverbial
eventuality	that happened	after that,	afterwards,
fact	that's a fact	in fact,	really,
proposition	that's true	in truth,	truly,
belief	that's my belief	in my view,	personally,
textual object	repeat that	to repeat,	again,

range of semantic objects to which a discourse deictic refers. As illustration, the last two columns of Table 2 contain adverbials from our corpus. For comparison, the second column contains comparable discourse deictic reference; the first column specifies the AO type. To see this range of adverbial modification, consider the sequence 'People made mistakes'. One can modify this by 'afterwards', as in 'Afterwards, people made mistakes', treating it as an *eventuality*; by 'in my view', as in 'In my view, people made mistakes', treating it as a belief, etc. Here, the adverbial determines the AO interpretation. In the case of discourse deictics—as in 'People made mistakes. *That* happened because they were tired'.—the predication on the discourse deictic determines the AO. 'Happening', for example, is a property of eventualities.

We claim that the *only* AO involved in the interpretation of clausal adverbials is the AO interpretation of the clause they adjoin to. In contrast, discourse adverbials have at least one other AO argument whose interpretation must be resolved to a non-NP constituent in the discourse, making them uninterpretable with respect to their matrix clause *alone*, even after the resolution of any NP-referring constituents.[11]

Discourse adverbials thus resemble discourse deixis in requiring an AO interpretation from the prior discourse. In fact, we can create a PP discourse adverbial simply by supplying a discourse deictic as the internal argument of an S-modifying PP. Our corpus provides several such examples, including *after that, beyond that*, etc. Moreover, just as a discourse deictic can be replaced by its demonstrative + AO counterpart, e.g. *that text, that speech act*, our corpus provides numerous instances where a demonstrative AO is the internal argument of an S-modifying PP. These illustrate an amazingly wide variety of abstract

---

[11] This judged 'uninterpretability in isolation' (after resolution of NP anaphora) is Knott's (1996) intuitive test for discourse connectives.

Table 3 PP adverbials with demonstrative AO internal arguments.

#	PP	#	PP	#	PP	#	PP
25	in that/ this case	12	in this way	4	for this reason	2	in this sense
21	at that/ this point	4	to that end	3	on this basis	2	in that event
13	by that/ this time	4	in this manner	2	at this stage	1	by that logic

objects, although most occur only once or twice. Some examples are shown, along with their corpus counts, in Table 3.

One source of ambiguity in the AO interpretation of demonstratives is that this interpretation may derive either from a non–NP constituent in the discourse context, as in (25a), or from an NP, as in (25b). Another source of ambiguity is time versus events: in (25c), 'that time' can refer to 'this morning' or to the event of waking. The potential for such ambiguities across all discourse adverbials means that unless the internal argument can *only* resolve to an NP, it may be most efficient to resolve it to the most inclusive AO that makes sense in the context and derive more specific readings from that.

(25) a. John couldn't sleep. *For that reason*, he got out of bed.
      b. Yesterday we discussed reasons for leaving. You gave me a good one. *For that reason*, I thank you.
      c. I woke up early this morning. *Since that time*, I've run many errands.

A PP discourse adverbial does not always make demonstrative AO reference to the previous discourse, however. The PPs in our corpus display a wide variety of internal arguments, including NPs, PPs, and S. The interpretation of these arguments determines whether or not a PP adverbial functions semantically as a discourse adverbial. For example, proper noun internal arguments such as in the PPs *in New York Stock Exchange composite trading* and *in August*, are not dependent for their interpretation on an AO in the discourse context. Nor are the first or second person pronoun internal arguments in *to me*, and *like you*. Definite AO internal arguments, however, such as are shown in Table 4, may or may not be interpreted anaphorically, although they always denote a specific known entity (Heim and Kratzer 1998).

Table 4   PP adverbials with definite AO internal arguments.

#	PP	#	PP	#	PP	#	PP
71	at the same time	11	at the moment	17	at the time	20	in the end
12	on the contrary	6	in the process	1	for the record	18	in the past
14	in the meantime	3	on the surface	4	on the way	2	by the way

In (26a), *in the process* refers to the *suing* event, but in (26b), *the record* simply refers to an abstract notion of right and wrong. Only when they refer to a non-NP constituent do these PP adverbials function as discourse adverbials.

(26)   a.   John sued Mary for a divorce. *In the process*, he cried profusely.
       b.   *For the record*, graduate students don't get paid enough.

The first entry in Table 4, 'at the same time', is always anaphoric, because *same*, like other *alternative phrases* (e.g. *other*, *another*) requires a discourse entity of the appropriate type for it to be the same as (or other to, etc.) (Bierner 2001). There are many such anaphoric modifiers, including determiners (e.g. *both*, *each*, *either*, *such* (Heim and Kratzer 1998)), comparative adjectives (e.g. *similar*, *better* (Bierner 2001)) and spatio-temporal adjectives (e.g. *current*, *present*). When these modifiers modify AOs, the resulting internal argument may be anaphoric to a non-NP constituent in the discourse context, thereby creating a discourse adverbial.

Indefinite or generic AO internal arguments do not in and of themselves refer to discourse context (Heim and Kratzer 1998). However, many of these PPs are in fact discourse adverbials, because the indefinite or generic NP is *relational*, taking a syntactically optional (or *hidden*) AO argument. Examples from our corpus are shown in Table 5. In these examples we can often make the hidden argument overt with a discourse deictic (e.g. *in addition to that*, *as a result of that*, *by comparison with that*, *in return for that*).[12]

---

[12] A German speaker has noted that in German these arguments are not optional; they are explicit or lexically incorporated into the adverbial. See Verspoor (1997) and Hardt (1999) for syntactic and semantic tests to distinguish English arguments and adjuncts.

Table 5 PP adverbials with indefinite/generic relational NP as internal arguments.

#	PP	#	PP	#	PP	#	PP
204	in addition	41	at least	10	by comparison	5	in sum
167	for example	28	by contrast	6	in response	4	in a sense
84	as a result	17	at last	2	as an alternative	3	in essence
70	for instance	16	in particular	2	as a consequence	3	in effect

These syntactically optional arguments have been widely discussed in linguistics, psycholinguistics, and computational linguistics (Mauner *et al*. 1995). In general, hidden arguments fall into two types: definite and indefinite. Indefinite hidden arguments are not necessarily anaphoric—e.g. Mary *ate* (Mary ate something) and Mary talked with *a mother* (Mary talked with a mother of someone). Definite hidden arguments, however, are anaphoric—e.g. the hidden argument of *apply* in (27a) is the earlier-mentioned grant; if the first sentence in this discourse is removed, the second is infelicitous unless the hidden argument is made overt. Mauner *et al.* (1995) note however that some implicit arguments can be definite or indefinite depending on context: in (27b), the hidden argument of *donate* must resolve to PETA, but in (27b) it need not resolve to a specific discourse entity.

(27) a. The due date for the grant has passed. Mary forgot to apply φ.
   b. PETA asked John for a contribution. John donated five dollars φ.
   c. Everyone knows John donates thousands of dollars each year φ.

There can be similar context-dependence in the hidden arguments of adverbials. For example while an *essence* must be an essence *of something*, only in (28a) is this 'something' resolved in the discourse context. In (28b), it is interpreted independently of context. And of course, the same variability in form and resolution displayed by the internal argument of PP adverbials can be displayed by these optional arguments when they are overt. For example, in (28c), the semantic argument of *consequence* is made explicit as a generic NP (*war*), and thus no discourse context is needed to interpret the adverbial.

(28) a. We choose the actors, build the sets, and keep the books. *In essence*, we run the show.

   b.   *In essence*, I am a happy person.
   c.   As a consequence of war, people die.

Sometimes the hidden argument of a relational phrase is interpreted as a *set*. For example, in (29), the hidden argument of *example* resolves to the set of *consequences* of John breaking his arm, as can be made overt by substituting *for an example of the consequences of breaking an arm*. This will become relevant when we discuss how these interpretations are obtained (section 5.3).

(29)   John just broke his arm. So, for example, he can't cycle to work.

We now turn to the S-initial, S-adjoined ADVP adverbials collected from the Penn Treebank I parses of the Brown and WSJ corpora. ADVP adverbials are distinguished morphologically as 'derived' and 'non-derived' (Alexiadou 1997). Most ADVP derive from compounding (e.g. *moreover*) or from ADJP/NP-affixing. In English, the *-ly* suffix predominates;[13] other affixes include *-wise* (e.g. *otherwise*) and *-ically* (e.g. *specifically*). Non-derived forms (e.g. *often*) can be analyzed via synonymy with derived forms.

Many of the ADVP adverbials in our corpus are related to (discourse) deixis, and thus their interpretation may depend on AO reference to a non-NP constituent in the prior discourse. Examples are shown in Table 6. Some of these deictic ADVP (e.g. *here*, *now*) function as discourse deictics in the same way as 'this' and 'that'. Others derive from compounding (e.g. *therefore*, *hereby*) and function as discourse adverbials. Still others are homonyms (e.g. *still*, *then*, *thus*), having both a discourse adverbial use and a (current or archaic) discourse deictic use (Knott 1996).[14] The difference between these uses is illustrated using *then* in (30). In (30b), *then* orders the event it modifies in a temporal sequence relation with the event described in (30a). In (30c), *then* makes discourse deictic reference to the event described in (30) and the it-cleft asserts that the temporal coordinates of this event are the same as the temporal coordinates of the event described in (30c).

(30)   a.   John and Mary had dinner.
       b.   *Then* he asked her to marry him.
       c.   It was *then* that he asked her to marry him.

Many other ADVP adverbials in our corpus derive from comparative adjectives, and thus are dependent on reference to at least

---

[13] in Greek: *-a/-os*, German: *-weise*, French: *-ment*, Italian: *-mente* (Alexiadou 1997).
[14] Miriam Webster Dictionary illustrates some archaic deictic uses: (i) He came *hence* to study English, (ii) He loves her *still*, (iii) He moved wearily and spoke *thus* to them.

Table 6 Deictic ADVP adverbials.

#	ADVP	#	ADVP	#	ADVP	#	ADVP	#	ADVP
292	then	80	yet	41	still	4	thereafter	1	thereupon
189	now	59	here	39	there	1	heretofore	1	therein
114	thus	48	therefore	26	hence	1	hereby	1	nowadays

Table 7 Comparative ADVP adverbials.

#	ADVP	#	ADVP	#	ADVP	#	ADVP
53	moreover	31	furthermore	19	otherwise	4	likewise
36	nevertheless	30	later	13	further	2	earlier

one other object in the context (Bierner 2001; Webber *et al.* 2003)—in some cases, the comparison is made between AOs derived from non–NP constituents. Some examples are shown in Table 7; in our corpus, some of these adverbs also appear with the compared object instantiated (e.g. *later than that*, *earlier than that*).

Comparative adjectives are one type of relational adjective. Many ADVP adverbials in our corpus derive from other relational adjectives or nouns that attribute properties to abstract objects, as shown in Table 8. The hidden arguments of these adverbials are often definite and resolve to non–NP constituents. Removing the suffix can allow the hidden AO argument to be made explicit (e.g. *according to this*). There are also non-derived ADVP relational adverbials in our corpus that appear with and without their explicit argument; for example, we find 18 tokens of *instead* and 14 tokens of *rather*. We also find 44 tokens of *instead of XP*, and 13 tokens of *rather than XP*, where XP appears as a variety of phrase types, and can refer to an AO.

Note that the hidden arguments of most relational spatio-temporal ADVP (and PP) adverbials (e.g. *recently*, *currently*) can resolve to either the discourse or spatio-temporal context. Some can even resolve within the modified clause. For example, in (31a), the events of Mary hearing a noise and Mike hearing a noise are interpreted as *simultaneous*. The coordinating conjunction, which also asserts the two events semantically and takes both arguments structurally, may play a role here (Creswell *et al.* 2002). The same use of *simultaneously* can also resolve its hidden argument to the prior discourse, as in (31b), where Fred seeing a light can be interpreted as the event that is simultaneous with

Table 8   Idiosyncratic relational ADVP adverbials.

#	ADVP	#	ADVP	#	ADVP	#	ADVP
12	similarly	8	simultaneously	5	conversely	2	separately
12	accordingly	8	consequently	3	additionally	1	alternately
12	recently	6	incidentally	2	previously	1	analogously

the event(s) of Mary and Mike hearing a noise. Even VP-modifying adverbials (e.g. *quickly*, *suddenly*) have this quality; in (31c), *quickly* can measure the time between raising the eye dropper and blinking, or between the start and finish of blinking.

(31)   a.   Simultaneously, Mary and Mike heard a noise.
      b.   Fred saw a light. Simultaneously, Mary and Mike heard a noise.
      c.   The doctor raised the eye dropper. John blinked quickly.

As we saw above with the PP adverbials, there are also relational ADVP adverbials in our corpus whose hidden argument resolves to the interpretation of a set. Some examples are shown in Table 9. The ordinals in the first column, for example, indicate the order of the modified element in a larger set. In (4), the relevant set is inter-preted as the reasons why Mary is a space cadet. The frequency adverbials in the second column invoke a particular type of set: a set of *times* (cf. Rooth 1995).[15] The adverbials in the remaining columns say next to nothing about their invoked set; rather, they specify a *comparative* role of the modified element or subset in relation to a larger set.

(32)   Mary is a space cadet. First, she always forgets to buy milk.

This discussion is not meant to imply that the hidden arguments of adverbials *always* resolve to AO interpretations. For example, some ADVP adverbials, including *obviously*, *unfortunately*, *surprisingly*, derive from relational adjectives or nouns whose hidden arguments resolve to concrete objects. In (33), the hidden argument of *obviously* is made explicit with the PP *for John*.

(33)   Unfortunately (for John), vandals covered John's car in graffiti.

---

[15] Epistemic adverbials (e.g. *probably*) have also been analyzed as invoking sets of possible worlds, which are the foundation of intensional semantics (cf. (Heim and Kratzer 1998)).

Table 9 Set-evoking ADVP adverbials.

#	ADVP	#	ADVP	#	ADVP	#	ADVP
47	finally	14	usually	2	mainly	3	significantly
34	first	8	occasionally	2	mostly	3	specifically
5	secondly	4	frequently	2	partly	2	basically

We conclude by noting that it is not only for their semantic argument structure and interpretation that adverbials may appeal to the discourse context. It may be a matter of pragmatic implicature (Forbes 2003). The adverbials *actually*, *in fact* and *surprisingly* have been categorized as discourse connectives (Knott 1996), although they simply assert that the interpretation of the modified clause is 'actual' or 'in the set of facts' or 'surprising (to someone)'. Because we do not normally distinguish those assertions that are 'actual' or 'in the set of facts' or 'surprising', the use of these adverbials creates a conversational implicature (Grice 1989) that the truth of the modified clause may not be expected, given the discourse interpretation so far.

Another reason that clausal adverbials may require a discourse context comes from the use of (topic) focus and Contrastive Theme (Steedman 2000). In the focus literature, for example, adverbials such as *at least* are viewed as being sensitive to focus. Although discussion has been limited to non–clausal modification, such as 'Most of my friends are not educated. John, at least, finished high-school.', the analysis also holds for clausal modification. Moreover, adverbials in first position, as exemplified in (34), can receive the same stress as other topicalized constructions, linking them to what they are being used to illustrate or contrast with in the discourse context (Rooth 1995).

(34)   Nowadays, we eat a lot of vegetables.

5   A SYNTAX-SEMANTIC INTERFACE (SSI) FOR D-LTAG: PART 2

5.1   *Prelude: Discourse adverbials in an LTAG SSI*

Section 4 showed how an S-modifying adverbial's semantic argument structure and interpretation cause it to function semantically as a discourse connective. All of the S-modifying adverbials in (35b), for example, function as discourse connectives because the interpretation of a definite semantic argument of the adverbial resolves to an AO interpretation of (35a).

(35) a.   The company interviewed everyone.
      b.   *In this way/In the process/As a result of that/As a result/*
           Consequently, they considered all their options.

This section shows how a clause-level syntax and semantics for discourse adverbials can be built from LTAG components, but the interpretation of their definite semantic arguments can only be resolved at the discourse level.

Of course, AO reference is not modeled structurally at either level. Definite reference (e.g. *the process*) has been represented semantically using partial functions (Heim and Kratzer 1998), such that the domain of *the* contains nouns that correspond to only one entity in the set of entities, but the definite noun may or may not be *anaphoric* in the sense that it refers to a salient entity in the context. Demonstrative reference (e.g. *this way*) is more akin to pronominal reference, in that it is almost always anaphoric. Pronominal reference (e.g. *that*) has been represented using assignment functions (Heim and Kratzer 1998), i.e. a pronoun denotes an entity e via an index i that is mapped to e relative to an assignment function a, where a is determined by a context c (e.g. $[\text{you}_i]^{a_c}$, where c might yield *Tom, Dick, Harry* for i).

As illustration of the clause-level compositional semantics of discourse adverbials, the LTAG elementary trees that compose to create *in this way* are shown in Figure 18; below each tree is its semantic representation. As shown, α_{way} is labelled with the unary predicate value q_1. α_{this} contains p_1, a unary predicate-argument variable, and supplies the individual value e, which denotes an individual anaphorically, e.g. relative to an assignment function. During bottom–up traversal of the derivation tree, α_{way} substitutes into α_{this} and p_1 takes the value q_1, yielding the interpretation *this'(way'([e_i]^{a_c}))*. This NP substitutes into β_{in} and provides the value of e for its internal argument variable, x_1. β_{in} also contains the external argument variable s_1, which resolves to the interpretation of the clause it modifies.

We extrapolate hereafter from Joshi *et al.*'s (2003) semantics for PPs as NP-adjuncts to PPs as S-adjuncts. We also extrapolate from their semantics for quantificational nouns to referring nouns. Note also that although in LTAG, demonstrative and definite nouns are associated with a +*definite* feature and pronouns are further associated with a +*pronoun* feature (XTAG 2001), our semantic representations use assignment functions to make explicit the anaphoric references to an AO in the discourse context.

The clause-level compositional semantics of PP discourse adverbials containing definite nouns or demonstrative pronouns can be
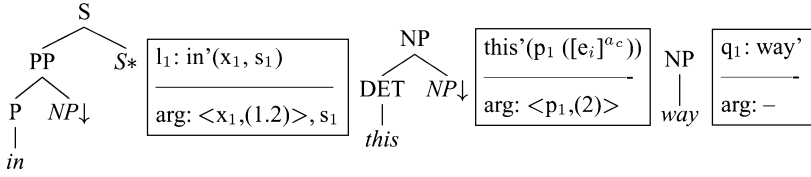
Figure 18 LTAG elementary trees and semantic representations for *In this way*.

extrapolated from Figure 18; PP adverbials taking binary predicates as internal arguments are only slightly more complicated. In Kallmeyer (2002), binary NP predicates (e.g. *winner of NP*) are modelled using NP elementary trees that select for a PP and contain a substitution site for its internal argument; the preposition is treated as semantically void. As illustration, elementary LTAG trees for *as a result of that* are shown in Figure 19. α_{that} denotes an individual e relative to an assignment function. $\alpha_{resultof}$ is represented as a binary predicate value, q_1, containing an argument variable, x_2, whose value will become e after substitution of α_{this} during bottom up traversal. $\alpha_{resultof}$ itself denotes an entity, γ, represented as a bound variable that will be instantiated as the individual variable, z, after substitution into α_a. The predicate variable of α_a, p_1, will resolve to q_1, yielding the interpretation $a'(resultof'(z,that'([e_i]^{a_c})))$. This NP will substitute into β_{as} and provide the value z for its internal argument variable, x_1. β_{as} also contains the external argument variable s_1, which will resolve to the interpretation of the modified clause.

When the internal argument of *result* is not syntactically overt, as in *as a result*, the atomic NP tree, α_{result}, can be employed, with its hidden anaphoric argument, e, represented only in the semantics, as shown in Figure 20.

We saw in section 4 that a wide variety of ADVP adverbials can also contain hidden semantic arguments. Their clause-level semantic representation is exemplified in Figure 21. As shown, *consequently* takes only the clause it modifies as a structural argument (s_1); e represents the hidden anaphoric semantic argument that must be resolved anaphorically at the discourse level.

5.2 *An SSI for D-LTAG: Discourse adverbials*

We now extend the D-LTAG syntax-semantic interface to discourse adverbials, assuming the LTAG-derived semantics given in the previous section. Figure 22 shows the elementary D-LTAG trees for the five
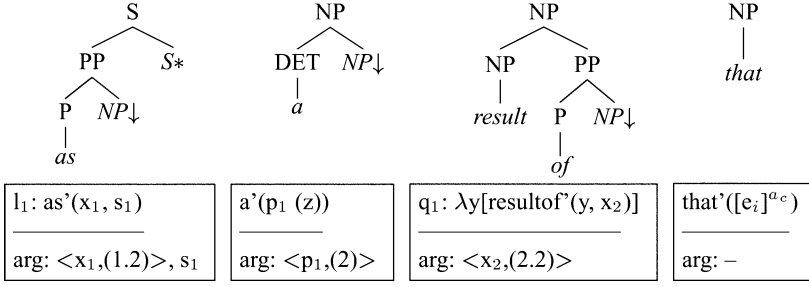
Figure 19 LTAG elementary trees and semantic representations for *as a result of that*.
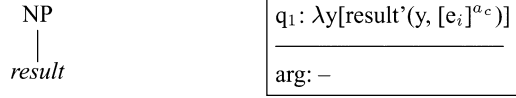

Figure 20 LTAG elementary tree and semantic representation for *result*.

discourse adverbials in (35). As we saw in the previous section, at the *clause level*, these adverbials are represented with complex trees. However, as discussed in section 3, at the *discourse level*, discourse units (D_u) and discourse connectives (DC) are treated as atomic units and thus are represented with elementary trees. This is because D–LTAG uses the same syntax and semantics used at the clause level to build discourse structure and interpretation directly on top of clause structure and interpretation. Therefore, although atomic units at the discourse level may in fact be complex units at the clause level, their clause level syntax and semantics can be retrieved from the D–LTAG parser, but is not necessary for constructing discourse structure and interpretation.[16]

Below each tree is shown its discourse–level semantic representation, where hyphenated forms abbreviate the embedded clause–level semantics. As discussed in section 4, a discourse adverbial contains a structural argument variable, s_1, which will be added to the discourse interpretation after adjunction and will obtain the value of the interpretation of the discourse unit to which it adjoins, and it also contains a definite internal semantic argument, which resolves

---

[16] Distinguishing discourse and clausal adverbials automatically will involve analysis of our manually labeled corpus of discourse adverbials (Miltsakaki *et al*. 2004; Prasad *et al*. 2004).

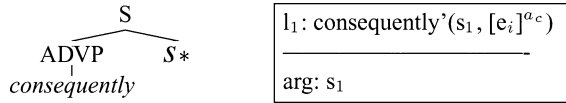$$l_1: \text{consequently'}(s_1, [e_i]^{a_c})$$

$$\text{arg: } s_1$$

Figure 21 LTAG elementary tree and semantic representation for *consequently*.

anaphorically to an AO interpretation derived from a non-NP constituent in the discourse context. We represent this argument, e, using an assignment function, to emphasize its context-dependence.[17]

We now illustrate the construction of the D-LTAG tree and compositional semantics for a simple discourse containing a discourse adverbial, using the discourse from (35) with the discourse adverbial *in this way*, as in (36).

(36)   a.   The company interviewed everyone.
       b.   *In this way*, they considered all their options.

The set of elementary D-LTAG trees required for this discourse are shown in Figure 23 beside their semantic representations. Note that to build this discourse structure, the elementary tree representing the empty connective (anchored by ε) is employed. As discussed in section 3, β_ε takes two structural arguments, one by substitution and the other by adjunction; except for the anchor, its structure is identical to β_{and}. We represent the interpretation of β_ε, e.g. the *continuation* of the discourse, simply as ε here, with the rest of its predicate-argument semantics identical to β_{and}. Note, however, that ε denotes an underspecified semantic discourse relation, whose specification will vary depending on the context in which it appears. As above, atomic discourse units are represented syntactically as initial trees (α) subscripted by the name of the main verb, and their semantic interpretation is associated with a discourse unit label (e.g. l_1).

Figure 24 shows the derived tree and derivation tree for (35), along with the semantic representation that results from a bottom up traversal

---

[17] 'As a result' and 'as a result of that' may express the same discourse relation. However, more research is needed. First, the complete set of discourse relations is still an open question. Second, the different forms may indicate different functions. For example, 'as a result' could also be glossed as 'as a result of *this*'. Without the explicit PP, one makes no commitment to the speaker's perspective on the AO (close/towards, or distant/away). 'As a result of this/that' can also be used with a physical gesture to link a situation in the world to a speaker-specified consequence (e.g. 'As a result of this ⟨speaker gestures at clothes lying around the hearer's bedroom⟩, I'm going to have to do three extra washes this week.'). Plain 'as a result' can't be used here. Currently, we thus give each discourse adverbial a separate interpretation.
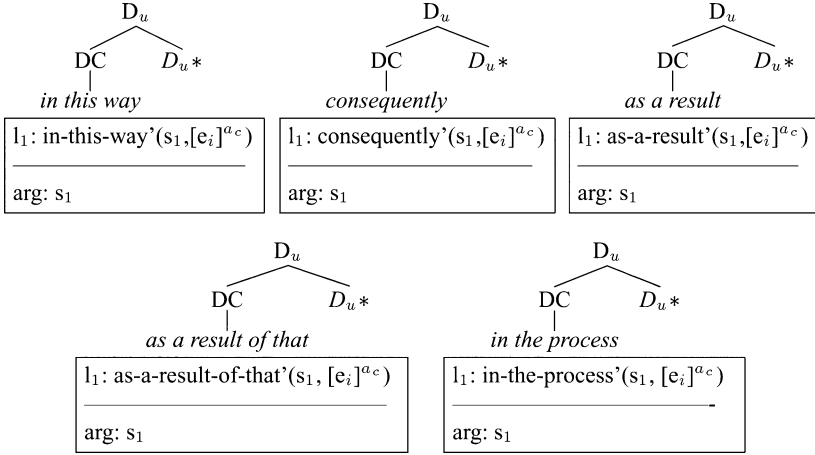
D_u
DC D_u*
in this way
l_1: in-this-way'$(s_1,[e_i]^{ac})$
arg: s_1

D_u
DC D_u*
consequently
l_1: consequently'$(s_1,[e_i]^{ac})$
arg: s_1

D_u
DC D_u*
as a result
l_1: as-a-result'$(s_1,[e_i]^{ac})$
arg: s_1

D_u
DC D_u*
as a result of that
l_1: as-a-result-of-that'$(s_1, [e_i]^{ac})$
arg: s_1

D_u
DC D_u*
in the process
l_1: in-the-process'$(s_1, [e_i]^{ac})$
arg: s_1

Figure 22 D-LTAG elementary trees and semantic representations for adverbials in (35).

D_u
DC D_u*
in this way
l_1: in-this-way'$(s_1,[e_i]^{ac})$
arg: s_1

D_u
D_u* DC $D_u\downarrow$
ϵ
l_2: ϵ'(s_2,s_3)
arg: s_2, $<s_3,(3)>$

D_u
$\alpha_{interview}$
l_3: interview'(c,e)
arg: $-$

D_u
$\alpha_{consider}$
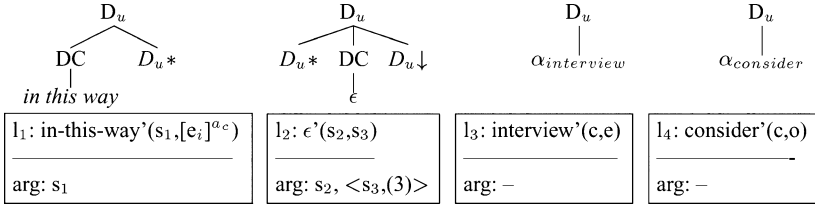l_4: consider'(c,o)
arg: $-$

Figure 23 D-LTAG elementary trees for example (36).

of the derivation tree. As shown, $\beta_{in-this-way}$ adjoins to $\alpha_{consider}$ and its adjunction argument variable, s_1 is identified with l_4. The resulting derived tree then substitutes into β_ϵ at position (3) and the substitution argument variable associated with this position (s_3) is identified with l_1. This derived tree then adjoins to $\alpha_{interview}$, whose label, l_3, is identified with the adjunction argument variable, s_2.

Note that the anaphoric argument of *in this way* is not resolved by compositional semantics. That it resolves to l_3: *interview'(c, e)* must be determined by anaphora resolution. Clearly, however, bottom–up traversal of the derivation tree yields the intended compositional semantics. If instead we use a top-down traversal, the interpretation produced would be that of two *separate* discourse relations between the two discourse units: the ϵ continuation relation would relate the first and second sentence only—the fact that *in this way* adjoins to the second sentence would *not* be reflected in the ϵ relation. The relation
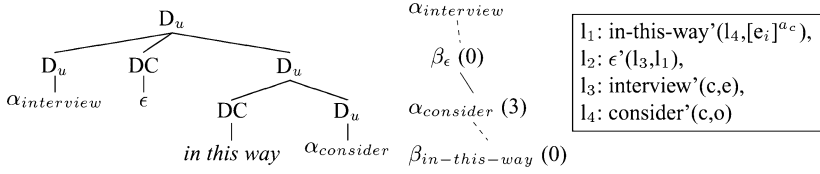
D_u

D_u DC D_u

$\alpha_{interview}$ ϵ DC D_u

in this way $\alpha_{consider}$

$\alpha_{interview}$

β_ϵ (0)

$\alpha_{consider}$ (3)

$\beta_{in-this-way}$ (0)

l_1: in-this-way'(l_4,$[e_i]^{ac}$),
l_2: ϵ'(l_3,l_1),
l_3: interview'(c,e),
l_4: consider'(c,o)

Figure 24 D-LTAG derived and derivation trees, and semantic representation for (35).

conveyed by *in this way* between the first and second sentence would be a separate formula.

5.3 An SSI for D-LTAG: Special case of 'for example'

The adverbial *for example* appears to behave differently from many other discourse adverbials, contributing a relation that is *parasitic on* another predication (Webber *et al.* 2003). Consider the following example:

(37) John just broke his arm. *So, for example,* he can't cycle to work now.

The interpretation of (37) is that John not being able to cycle to work is *one example of the consequences of John breaking his arm*. In other words, the interpretation of the left argument of *for example* is dependent on the 'consequence' relation supplied by the structural connective *so* between the first and second sentences. Webber *et al.* (2003) argue that this interpretation involves first abstracting the meaning of the left argument with respect to the the meaning of the right argument, and then making an assertion with respect to this abstraction. More precisely, if γ represents the interpretation of *John just broke his arm*, and δ represents the interpretation of *he can't cycle to work now*, and *result(γ, δ)* represents the interpretation of the relation *so* applied to γ and δ, then the interpretation of *for example* is as in (38), where *exemplify* represents the relation that *for example* supplies. Webber *et al.* (2003) note that such an interpretation resembles that of an English quantifier, in that the scope of its interpretation is wider than is explained by its syntactic position.

(38) exemplify(δ, λX. explanation(γ, X))

English quantifiers have been treated syntactically as initial NP trees into which a generic noun substitutes (Kallmeyer 2002). However, to get the right scope interpretations, Joshi *et al.* (2003) use the

Figure 25 MC-TAG for {\em for example} and derivation tree for (37).

restricted approach to flexible composition that was introduced in section 3.3, along with tree-local multi-component TAGs (MC-TAGs) (Weir 1988). In a tree-local MC-TAG, a set of trees represents a lexical item; all trees in the set are added to the derivation simultaneously, and they are added to nodes belonging to the same elementary tree. Joshi *et al.* (2003) also allow limited non-tree-local attachments, but these are restricted to trees that do not carry any syntactic material. In this way, the generative capacity is not extended. In Joshi *et al.* (2003), a quantifier MC-TAG contains two elementary trees that represent the quantifier: an auxiliary tree consisting of a single adjunction node represents the scope part of the quantifier semantics, and an initial tree containing a substitution site represents the predicate-argument part.

Based on its similarity with quantifiers, Webber *et al.* (2003) extend this approach to *for example*, associating it with the MC-TAG shown in brackets in Figure 25, which consists of two *auxiliary* trees. The second auxiliary tree adjoins to the root of the discourse unit it modifies, while the first auxiliary tree adjoins to the root of the *higher* discourse unit—i.e., the one being abstracted. As illustration, derivation tree for (37) is also shown in Figure 25, where β_{ex1} signifies the first tree in the MC-TAG for *for example*, and β_{ex2} signifies the second tree in the MC-TAG. The syntactic tree for *so* (α_{so}) is identical to that of $\alpha_{because}$ (see Figure 11), taking two substitution arguments. α_{break} signifies the tree for the first clause in the discourse, and α_{cycle} signifies the tree for the second clause in the discourse.

The derivation of (37) then proceeds as follows: β_{ex2} adjoins to α_{cycle}. The result is a derived tree whose scope part is still β_{ex1}. Both compose (tree-locally) into α_{so}. The interpretation that results from this composition step both embeds and abstracts the interpretation of α_{so}, which, after the resulting tree composes with α_{break}, yields e.g. (39):

(39)   exemplify'($\neg$cycle'(j, w), λX.so'(break'(j', a'), X))

We now show how this semantics can be computed using the D-LTAG syntax–semantic interface proposed in this paper. Figure 26

l_1:exemplify'$(h_1, \lambda s.h_2)$ $h_2 \geq s_1$ —— arg: s_1	l_2: s $g_1 \geq l_2$ $h_1 \geq s_2$ —— arg: s_2	l_3: so'(s_3,s_4) $h_3 \geq l_3$ —— arg: $<s_3,(1)>$ $<s_4,(3)>$	l_4:break'(j,a) $h_4 \geq l_4$ —— arg: −	l_5:¬cycle'(j,w) $h_5 \geq l_5$ —— arg: −

Figure 26   D-LTAG semantic representation of the elementary trees used in (37).

shows the semantic representations of the elementary trees for the discourse in (37). The first two representations are associated with β_{ex1} and β_{ex2}, respectively. As shown, like the 'predicate-argument' and 'scope' parts of *every* above, the representation for β_{ex2} supplies and embeds the variable over which the representation of β_{ex1} completes the abstraction. (We use the propositional variable *s* to avoid confusion with clause-level entity variables.) Unlike *every*, however, the predication supplied by *for example* scopes over the application of the variable to the abstraction, and thus is part of the representation of β_{ex1}. The third representation is associated with α_{so}. The last two representations are associated with α_{break} and α_{cycle}, respectively.[18] As in Joshi *et al.* (2003), the holes and scope constraints in these representations are crucial for capturing the semantic interpretation represented in (39).

The derivation of (37) now proceeds as above: β_{ex2} adjoins to α_{cycle}. The semantic representation of β_{ex2} contains a hole variable, g_1[19], and an argument variable, s_2, both of which must be associated with values from the adjoined argument. g_1 thus obtains the value of the hole in α_{cycle}: h_5, and s_2 obtains the value of the label of α_{cycle}: l_5. The resulting derived tree substitutes into α_{so} at position (3), yielding the value l_2 for the argument variable at that position in α_{so}: s_4. β_{ex1} simultaneously adjoins to the root of α_{so}, and its argument variable, s_1, is thus associated with the label l_3. Finally, α_{break} substitutes into position (1) of this derived tree, yielding the value l_4 for the argument variable, s_3. The resulting semantics is shown in Figure 27.

Scope disambiguation then consists of matching each hole to a unique label such that the scope constraints are obeyed. We can visualize the process by viewing labels along an ordered plane of 'label boxes', and holes along a parallel plane of 'hole boxes', as in Figure 28. The formulas in Figure 27 tell us that $l_3 > l_4$ and $l_3 > l_2$, because l_4 and l_2 appear inside the formula labeled l_3. Thus, we need at least two label boxes below l_3. The scope constraints in Figure 27, $h_2 \geq l_3$ and $h_3 \geq l_3$,

---

[18] For the sake of simplicity, we have incorporated the negation into the representation here.
[19] See also the representation of β_{often} in Figure 9.

l_1: exemplify'$(h_1, \lambda s.h_2)$, l_2: s, l_3: so'(l_4, l_2), l_4: break'(j,a), l_5: ¬cycle'(j,w)
$h_2 \geq l_3$, $h_5 \geq l_2$, $h_1 \geq l_5$, $h_3 \geq l_3$, $h_4 \geq l_4$, $h_5 \geq l_5$

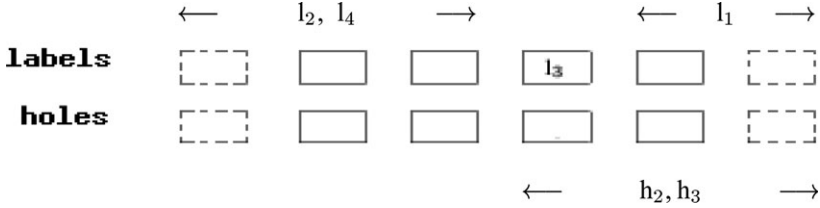**Figure 27** D-LTAG semantic representation for (37).

Figure 28 Partial matching of holes and labels from Figure 27.

tell us that we also need at least one label box above l_3, because both of these holes can't be matched with l_3. We further know that $l_1 > h_1$ and $l_1 > h_2$, and that these holes cannot match l_1, because they appear inside l_1. Thus, because $l_1 > h_2$ and $h_2 \geqslant l_3$, we know $l_1 > l_3$. This information is visualized in Figure 28. Because we don't yet know where l_5 lies, we show a dotted box on either edge of the figure.

We can however deduce the position of l_5: If $l_5 > l_3$, then because every hole matches a unique label, the following scope constraints tell us that we need three holes above l_3: $h_2 \geqslant l_3$ and $h_3 \geqslant l_3$, $h_1 \geqslant l_5$, $h_5 \geqslant l_5$. But three is more labels than can exist above l_3. Therefore, $l_5 < l_3$. This result leaves only l_1 and l_3 as possible matches for h_3 and h_2, but we already saw that $l_1 > h_2$, therefore $h_2 = l_3$ and $h_3 = l_1$.

Now we have h_5, h_4 and h_1 still to be matched with l_5, l_4 and l_2. We know: $h_4 \geqslant l_4$, $h_1 \geqslant l_5$, $h_5 \geqslant l_5$, and $h_5 \geqslant l_2$. If $h_5 = l_5$, then there would have to be a label to match the hole h_1 above l_5, as well as a hole to match the label l_2 below l_5. The label above l_5 would have to be l_4, yielding $h_1 = l_4$ and $h_4 = l_2$. But the scope constraints told us that $h_4 \geqslant l_4$. Thus, $h_5 \neq l_5$. If $h_4 = l_5$, then again l_4 would match the hole h_1 above l_5, and the hole h_5 would thus match the label l_2 below l_5. But the scope constraints told us that $h_4 \geqslant l_4$. Thus, $h_4 \neq l_5$. Therefore, $h_1 = l_5$. Finally, we have h_4 and h_5 to match with l_4 and l_2. We know $h_4 \geqslant l_4$ and $h_5 \geqslant l_2$. If $h_4 = l_2$ then $h_5 = l_4$ and we get $h_2 > l_4$ and $h_4 > l_2$, which is a contradiction. Thus, $h_4 = l_4$ then $h_5 = l_2$.

Replacing holes with labels in Figure 27, we get: *exemplify'*$(l_5, \lambda s.l_3)$, and replacing labels with formulas yields the same formula given in (39), namely:

(40) exemplify'$(¬$cycle'$(j,w), \lambda s.$so'$($break'$(j', a'), s))$

It seems natural to consider how Webber *et al.* (2003)'s analysis of scope effects in *for example* extends to the interpretation of other 'set-taking' discourse adverbials (i.e. whose left argument resolves to a set). In addition to *for example*, we saw a variety of other adverbials in Table 5 and Table 9 whose hidden arguments can resolve to a set of AOs. Two examples are *specifically* and *in particular*. In (41), we see that the relation supplied by these discourse adverbials is similarly *parasitic on* the relation that the structural connective supplies between the first and second sentence.

(41) John broke his arm. *So, specifically/in particular*, he can't cycle to work.

The interpretation of the discourse with *specifically* is that John not being able to cycle to work is *one specific consequences of John breaking his arm*. The interpretation of the discourse with *in particular* is that John not being able to cycle to work is *one particular consequence of John breaking his arm*. In other words, the left argument of these 'set-taking' discourse adverbials are similarly dependent on the 'consequence' relation supplied by the structural connective *so* between the first and second sentences. Thus, a semantic treatment similar to that discussed here for *for example* is likely warranted for other PP and ADVP discourse adverbials whose hidden argument must also be a set or an abstraction.

Of course, we saw above that the kinds of 'set' interpretations available to these adverbials can vary. For example, frequency adverbials such as *usually* and *at times* take sets of *times*; therefore their hidden arguments cannot resolve to the set of *consequences* made available by the structural connective *so*. The possible interactions between structural connectives and discourse adverbials are complex indeed, and will play an important role in deciding the most efficient way to model discourse structure and interpretation. We leave further details of these interactions for future work.

5.4 *An SSI for D-LTAG: Larger discourses*

We end this discussion by highlighting other issues that arise when considering how the compositional semantics of more complex discourses are computed in D-LTAG; these issues will impact future work on the D-LTAG syntax-semantic interface. One such issue concerns the ambiguity that can arise in a longer discourse with multiple discourse adverbials, such as (42).
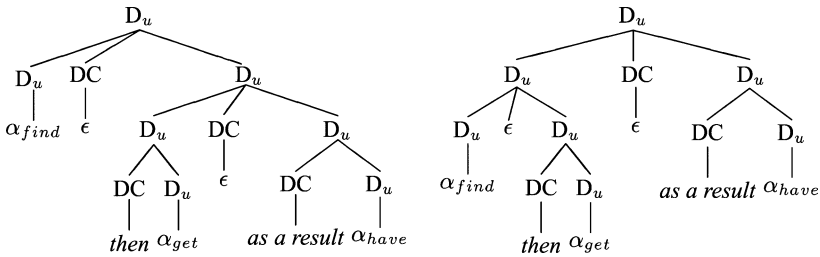
Figure 29 Two possible D-LTAG derived trees for (42).

(42)   Mary found a job. Then Mike got a raise. As a result, they had
       enough money to buy a house.

One interpretation of (42) is that Mike and Mary having enough money to buy a house is the result of *both* Mike getting a raise *and* Mary finding a job. Another interpretation is that it is the result *only* of Mike getting a raise.[20] If we limit the resolution of the internal (anaphoric) argument of *as a result* to a structural constituent, then these two different interpretations can be associated with the two derived trees shown in Figure 29.

If the anaphoric argument of *as a result* is restricted to a structural constituent, then the first tree admits only 'Mike got a raise' as the argument, while the second tree admits both possibilities. If the anaphoric argument is further restricted to a sister constituent (Stone 1994), then the second tree admits only the joint interpretation. This is similar to the problem of interpreting discourse deixis, which also appears to interact with the structural description of the discourse (Webber 1991; Passonneau 1991; Stone 1994).

Another issue is whether or not our approach will always yield the intended discourse interpretation. In (43), we can only interpret the relation supplied by *otherwise* as embedded in the relation supplied by *because*. In other words, the interpretation of (43) is paraphrasable as: *if the light is red, stop, because if you do something other than stop, you'll get a ticket.*

(43)   a.   If the light is red,
       b.   stop
       c.   *because otherwise* you'll get a ticket.

However, in (44), we may want to allow an interpretation where the relation supplied by *because* is interpreted *separately* from the relation

---

[20] A third potential interpretation does not seem possible: that Mike and Mary's having enough money to buy a house is the result *only* of Mary finding a job.

supplied by *then*. This would yield two possible interpretations of (44), which differ in terms of the following paraphrases: (1) *He had to cancel the order because after he ordered three cases he discovered he was broke*, (2) *he had to cancel the order because he discovered he was broke & he ordered three cases then he discovered he was broke*.

(44)   a.   John loves Barolo.
        b.   So he ordered three cases of the '97.
        c.   But he had to cancel the order
        d.   *because then* he discovered he was broke.

A bottom–up traversal of the derivation trees for these examples will produce the first (embedded) interpretation. We need a top-down traversal of the derivation tree to produce the second (separate) interpretation of (44). Allowing both traversal orders, however, introduces semantic ambiguity. Whether or not these two interpretations should be distinguished is still an open question; at least in (44) it can be argued that the second interpretation does not 'buy us' anything new, i.e., the two interpretations are equatable.

Finally, computing compositional semantics is still ongoing work in LTAG (i.e. at the clause level). Thus, because D-LTAG uses the LTAG syntax and semantics to build discourse structure and interpretation directly on top of the clause, innovations in computing clause-level semantics can impact the discourse level, particularly as more and more discourses are considered. Recently, for example, Romero and Kallmeyer (2005)[21] have proposed to combine Joshi *et al.*'s (2003) approach with the approach in Gardent and Kallmeyer (2003), which uses LTAG's syntactic feature unification mechanism to compute semantics on the derived tree. Romero and Kallmeyer (2005) argue however that the *derivation tree* is the preferred structure for computing semantics, and the feature structures needed for semantics have different formal properties than those needed for syntax: namely, the former contain semantic variables and labels as feature values, so their number is not finite. Romero and Kallmeyer (2005) thus propose to first incorporate semantic feature structures into the derivation tree and then use feature structure unification on the derivation tree to compute semantics.

We briefly illustrate how Romero and Kallmeyer's (2005) approach extends to the discourse level, using example (43) above. Figure 30 shows the derivation tree for (43), where the elementary D-LTAG trees are replaced by their semantic representations and linked semantic feature structures. As shown, these semantic representations are

---

[21] This approach is also discussed in Kallmeyer and Romero (2004).

l_1: because'($\boxed{1}$, $\boxed{2}$)

$$\begin{bmatrix} DU_1 & \left[T \quad \left[P \quad \boxed{1} \right] \right] \\ DU_2 & \left[T \quad \left[P \quad \boxed{2} \right] \right] \end{bmatrix}$$

(1)   (3)

l_2: if'($\boxed{3}$, $\boxed{4}$)

$$\begin{bmatrix} R & \left[T \quad \left[P \quad l_2 \right] \right] \\ DU_1 & \left[T \quad \left[P \quad \boxed{3} \right] \right] \\ DU_2 & \left[T \quad \left[P \quad \boxed{4} \right] \right] \end{bmatrix}$$

l_5: get'(y, t)

$$\begin{bmatrix} R & \left[B \quad \left[P \quad l_5 \right] \right] \end{bmatrix}$$

(1)   (3)   (0)

l_3: is-red'(l)

$$\begin{bmatrix} R & \left[B \quad \left[P \quad l_3 \right] \right] \end{bmatrix}$$

l_4: stop'(y)

$$\begin{bmatrix} R & \left[B \quad \left[P \quad l_4 \right] \right] \end{bmatrix}$$

l_6: otherwise'($\boxed{5}$)

$$\begin{bmatrix} R & \left[B \quad \left[P \quad l_6 \right] \right] \\ F & \left[T \quad \left[P \quad \boxed{5} \right] \right] \end{bmatrix}$$
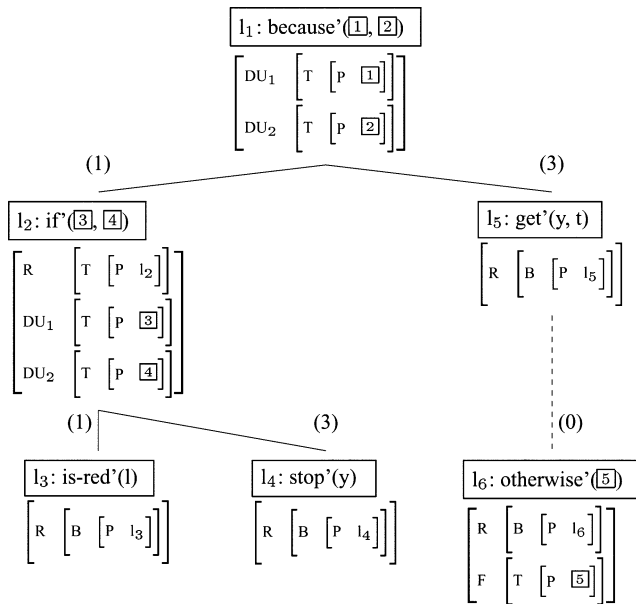
Figure 30   Semantic representation for example (43).

essentially unchanged from Joshi *et al.* (2003) (consisting of labelled formulas and scope constraints); however, they are linked to complex semantic feature structures, which have type *sem* and consist of features for each node position in an elementary tree whose values are also feature structures, of type *tb*. Feature structures of type *tb* consist of a *T* and *B* feature (top and bottom), whose values are also feature structures of type *bindings*, which consist of a feature *I*, whose values are individual variables, a feature *P* whose values are propositional labels, etc.

In Figure 30[22], $\alpha_{because}$ roots the derivation tree, is labelled by l_1, and takes two (substitution) arguments. Note that feature and propositional variables are represented in this approach using boxed numbers (e.g. $\boxed{1}$). In the linked semantic feature structure, there are features shown for each substitution node, DU_1 and DU_2, whose value is a feature structure of type *tb*. Because these are substitution sites, these feature structures consist only of a top (T) semantic feature, whose value is a feature structure of type *bindings* that consists of a feature P, whose value is a propositional variable representing the substitution argument, e.g. $\boxed{1}$, which will eventually resolve to a propositional label through feature unification.

---

[22] Although there are (possibly empty) features for every node in an elementary tree, as in Romero and Kallmeyer (2005), we only show selected, non-empty node features in Figure 30.

α_{if} is labelled by l_2, and takes two (substitution) arguments. There are features shown for the two substitution nodes, DU_1 and DU_2, and the root node (R). The value of the substitution node features is akin to those for $\alpha_{because}$. The R node has a T feature structure consisting of a feature P whose value is the semantic label of α_{if}, to allow this label to percolate up the tree during unification. The R node's B feature is empty, and not shown here.

$\beta_{otherwise}$ is labelled by l_6, and takes one (adjunction) argument. There are features shown for the R node and the foot node (F). The R node has a B feature structure consisting of a feature P whose value is the semantic label of $\beta_{otherwise}$, to allow this label to percolate up the tree during unification. The F node has a T feature structure consisting of a feature P whose value is a propositional variable representing the adjunction argument.

α_{is-red}, α_{stop} and α_{get} are elementary (clausal) discourse units, labelled by l_3, l_4 and l_5, respectively. Only their R node feature is shown, whose value is a B feature structure consisting of a feature P whose value is the semantic label of the discourse unit. The R node's empty T feature is not shown.

Semantic composition then consists *only* of feature unifications: the feature unifications in the syntax performed during substitutions and adjunctions, as well as the final top–bottom unifications in the derived tree. In the derivation tree, elementary trees are replaced by their semantic representations plus the corresponding feature structures, as was shown in Figure 30. Then, for each edge in the derivation tree from γ_1 to γ_2 with position p, the unifications in (45) are performed, which yields values for variables in the semantic representation. Finally, the union of all semantic representations is built.

(45) a. The T (top) feature of position p in γ_1 and the T feature of the R (root) position in γ_2 are identified.
      b. If γ_2 is an auxiliary tree, then the B (bottom) feature of the F (foot) node of γ_2 and the B feature of position p in γ_1 are identified.
      c. For all positions p for all γ in the derivation tree such that there is no edge from p to another tree, the T and B features of p are identified.

For our discourse in (43), these unifications lead to the feature value identities in Figure 31. Starting at the root, α_{if} substitutes into the left argument position (1) of $\alpha_{because}$. By (45a), the T feature of DU_1 in $\alpha_{because}$ and the T feature of the R position in α_{if} are identified as $\boxed{10}$. In the same way, α_{get} substitutes into the right argument position (3) of
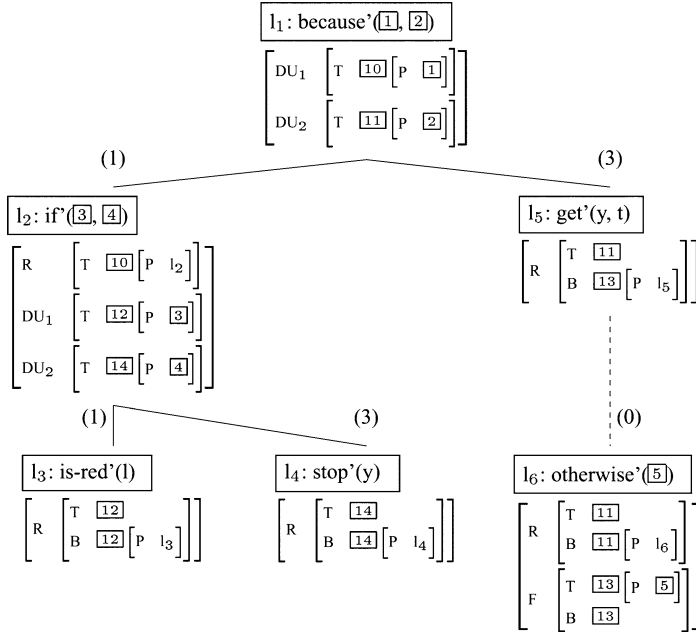
Figure 31 Semantic unifications for example (43).

$\alpha_{because}$, and the T feature of DU_2 in $\alpha_{because}$ and the T (empty) feature of the R position in α_{get} are identified as $\boxed{11}$. Similarly, α_{is-red} substitutes into the left argument position (1) of α_{if}; thus the T feature of DU_1 in α_{if} and the T (empty) feature of the R position in α_{is-red} are identified as $\boxed{12}$. In the same way, α_{stop} substitutes into the right argument position (3) of α_{if}; thus the T feature of DU_2 in α_{if} and the T (empty) feature of the R position in α_{stop} are identified as $\boxed{14}$. Finally, $\beta_{otherwise}$ adjoins to the root position (0) of α_{get}, and the T feature of R in α_{get} and the T (empty) feature of the R position in $\beta_{otherwise}$ are identified as $\boxed{11}$. Furthermore, by (45b), the B feature of the F position in $\beta_{otherwise}$ and the B feature of the R position in α_{get} are identified as $\boxed{13}$.

With all unifications at the edges in the derivation tree completed, final top-bottom unifications then occur. These unifications give the identities $\boxed{1} = l_2$ (via $\boxed{10}$), $\boxed{2} = l_6$ (via $\boxed{11}$), $\boxed{3} = l_3$ (via $\boxed{12}$), $\boxed{4} = l_4$ (via $\boxed{14}$), and $\boxed{5} = l_5$ (via $\boxed{13}$), which results in the semantic representation in (46).

(46)

| l_1: because'(l_2, l_6), l_2: if'(l_3, l_4), l_3: is-red'(l), l_4: stop(y), l_6: otherwise'(l_5), l_5: get'(y, t) |

With respect to the simple discourses discussed herein, there is no great advantage to computing discourse semantics using Romero and Kallmeyer (2005)'s approach over Joshi *et al.*'s (2003) approach. We saw that if we compute discourse semantics directly from node attachments in the derivation tree, then we need the bottom up traversal achieved by flexible composition to produce the correct semantic value of a complex embedded argument resulting from adjunction of a discourse adverbial. If we instead compute discourse semantics using semantic feature unification in the derivation tree, then we can use feature unification to pass the semantic value of the complex embedded argument up through the feature structures in the derivation tree, and thus we don't need flexible composition for our examples.

More generally, however, flexible composition and semantic feature unification have different intended benefits. Joshi *et al.* (2003) show that by allowing node attachment to proceed in different directions, flexible composition can license syntactic derivations that are not allowed with standard local (MC-)TAGs, and computation of semantics (using either approach) on these new structures yields new semantic interpretations. Romero and Kallmeyer (2005) show that although flexible composition and semantic feature unification both allow semantic features of attached nodes to be passed in different directions, semantic feature unification also allows semantic feature values to be passed from one node of an elementary tree set to another node in the same elementary tree set (by structure sharing inside the elementary tree set). As a consequence, values can be shared between trees attaching to these two nodes. These trees could come from different elementary tree sets, and in this case these trees are not linked in the derivation tree, even with a flexible composition approach. So semantic feature unification allows a relation to be established that cannot be obtained with flexible composition (Laura Kallmeyer, p.c.). These different benefits of flexible composition and semantic feature unification will become more important in D-LTAG as we consider discourses of greater complexity.

6   COMPARISON WITH RELATED APPROACHES

There already exist tree-based approaches to incremental discourse construction. For example, Webber (1991) outlines a tree-based model of discourse to account for 'right frontier' constraints on discourse deixis interpretation; Scha and Polanyi (1988) and Polanyi and van den Berg (1996) go further and propose a discourse model that accounts

for structural constraints on the antecedents of other NP anaphora in local discourse spans. Gardent (1997), on the other hand, uses a variant of feature-based tree-adjoining grammars to construct discourse structure and semantics, while Schilder (1997) extends Gardent's formalism to handle world and contextual knowledge using a non-monotonic reasoning system. Marcu's (2000) approach to 'discourse parsing' is more empirically oriented. His system identifies discourse relations after training on a corpus annotated with RST-like relations (Mann and Thompson 1988), and then utilizes correlations between surface-based features and discourse relations to assign discourse relations, and thereby discourse structure, to an unseen corpus.

Despite the structural similarity of the above works with D–LTAG's approach, all of this prior work differ significantly in that the semantics of discourse is built wholly compositionally. Only D–LTAG accounts for the anaphoric properties of discourse adverbials. Moreover, in much of this work discourse connectives are treated as 'signals' of more abstract discourse relations. D–LTAG only attempts to address how their semantic role as discourse connectives derives naturally from their internal semantic argument structure and interpretation. Thus D–LTAG does not require a pre-defined set of abstract discourse relations, thereby relegating inference based on world knowledge and pre-supposition to a separable process. Furthermore, these other models all assume the prior identification of the discourse units and/or relations involved in discourse structure-building. The D–LTAG model does not assume pre-processing or segmentation of the textual input. Rather, the input to D–LTAG is the output of the clause-level parse, and discourse structure and semantics are built compositionally directly on top of the clause.

7 CONCLUSIONS AND FUTURE WORK

One issue that we have touched on but not resolved here is true discourse ambiguity—that alternative structural analyses are possible, each of which will be assigned a different compositional semantics in D–LTAG. For example, we showed in section 3 that the structural analyses of (19b) and (20b) differ, and that corresponding semantics can be computed on these different structures. However, an automated system would first have to decide between those structural analyses and apply the correct one to both (19b) and (20b). In current wide-coverage parsing systems, disambiguation is handled as a matter of statistics, but gathering statistics at the discourse level is a bigger challenge (but cf. Marcu and Echihabi 2002; Lapata 2003).

Similarly, we have not dealt with the issue of how the hidden AO argument of discourse adverbials can be automatically resolved. Understanding both the mechanisms that determine which elements are accessible to function as antecedents of the anaphoric arguments of discourse adverbials, and the elements which allow an AO interpretation to be derived, will require the production and analysis of annotated corpora such as the one described in Miltsakaki *et al.* (2004) and Prasad *et al.* (2004), and should consider analyses already proposed for discourse deixis. Note that as non-NP constituents, the arguments of *all* discourse connectives are interpreted as abstract objects, including the arguments of structural connectives. Of course, while the AO interpretation of the arguments of some structural connectives is easy to see, others are less obvious. For example, in 'S$_1$ because S$_2$', S$_2$ is interpreted as a *cause*, and S$_1$ is interpreted as a *result* or *effect*, but the AO interpretations in 'S$_1$ nevertheless S$_2$' are less clear. We hope to exploit AO interpretations via the implementation of semantic features in D-LTAG. For example, it may be possible to implement feature-based constraints concerning the AO types that discourse connectives require from their arguments. Such semantic features can also be used, as at the clause level, to handle scope ambiguities that can arise due to multiple (discourse) adverbial modifiers (cf. Ernst 1984).

Finally, as noted above, there are several ongoing issues in the LTAG syntax–semantic interface that are also ongoing in the D-LTAG syntax–semantic interface. For one, the interaction between LTAG feature structures and the LTAG syntax–semantic interface is not yet fully spelled out. For another, that interface has not yet been extended to fully compute the semantics of all syntactic constructions, including the DAGs discussed in section 2.2, in which a node may have more than one parent (provided there are no crossing structural dependencies).

Although there is much work still to be done, the D-LTAG syntax-semantic interface presented here nevertheless shows that discourse interpretation can be analysed modularly: certain functions can be attributed to the semantic domain, others to the pragmatic domain, and still others to larger issues of discourse structure. There are numerous benefits of this approach. First, it is economical, making use of pre-existing mechanisms to build the interpretation of discourse, thereby reducing the load on inference to account for discourse interpretation (cf. Kehler 1995). Secondly, it provides a theoretical grounding for Knott's (1996) empirical approach to discourse connectives, showing why certain adverbials function semantically as discourse connectives, while others do not. Thirdly, it shows how discourse interpretations

can be built directly on top of clause interpretations, thereby bridging the gap between clause-level interpretations and the shallow discourse interpretations produced by other models such as RST (Mann and Thompson 1988).

Acknowledgements

KATHERINE FORBES-RILEY
3 Campbell Street
Apartment 3F
Lebanon
NH 03766
e-mail: forbesk@babel.ling.upenn.edu

REFERENCES

Alexiadou, A. (1997) *Adverb Placement: A Case Study in Antisymmetric Syntax*. John Benjamins Publishing Company. Amsterdam/Philadelphia.

Asher, N. (1993) *Reference to Abstract Objects*. Kluwer. Dordrecht.

Bateman, J. (1999) 'The dynamics of "surfacing": An initial exploration'. In *Proceedings of the International Workshop on Levels of Representation in Discourse*. Edinburgh. Scotland, 127–133.

Bierner, G. (2001) *Alternative Phrases: Theoretical Analysis and Practical Application*. Ph.D. dissertation. University of Edinburgh, Edinburgh, Scotland.

Bos, Johan (1995) 'Predicate logic unplugged'. In P. Dekker & M. Stokhof (eds), *Proceeding of the 10th Amsterdam Colloquium*. University of Trient. Amsterdam, 133–142.

Candito, M.-H. & Kahane, S. (1998) 'Can the TAG derivation tree represent a semantic graph? An answer in the light of meaning-text theory'. In *Proceedings of the TAG+4 Workshop*. University of Pennsylvania. Philadelphia, 25–28.

Copestake, A., Flickinger, D. & Sag, I. (1999) *Minimal Recursion Semantics. An introduction*. Unpublished MS, Stanford University. California.

Creswell, C., Forbes, K., Miltsakaki, E., Prasad, R., Webber, B. & Joshi, A. (2002) 'The discourse anaphoric properties of connectives'. In *Proceedings of the 4th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC2002)*. Lisbon.

Creswell, Cassandre, Forbes, K., Miltsakaki, E., Prasad, R., Webber, B. & Joshi, A. (2004) 'The predicate-argument structure of connectives: A corpus-based study'. In A. Branco, T. McEnery & R. Mitkov (eds), *Anaphora Processing: Linguistic, Cognitive and Computational Modelling*. John Benjamins Publishing Company. Amsterdam/Philadelphia.

Creswell, C., Forbes, K., Miltsakaki, E., Webber, B. & Joshi, A. (2003)

'Anaphoric arguments of discourse connectives: Semantic properties of antecedents versus non-antecedents'. In *Proceedings of the EACL03 Computational Treatment of Anaphora Workshop*. Budapest.

Dahl, Ö. & Hellman, C. (1995) 'What happens when we use an anaphor'. In *Presentation at the XVth Scandinavian Conference of Linguistics*. Oslo, Norway.

Danlos, L. (2004) 'Sentences with two subordinate clauses: syntactic and semantic analyses, underspecified semantic representation'. In *Proceedings of the TAG+7 Workshop*. Simon Fraser University. Vancouver.

Ernst, T. (1984) *Toward an Integrated Theory of Adverb Position in English*. Ph.D. dissertation. Indiana University, Indiana.

Forbes, K. (2003) *Discourse Semantics of S-Modifying Adverbials*. Ph.D. dissertation. University of Pennsylvania, Philadelphia, PA.

Forbes, K., Miltsakaki, E., Prasad, R., Sarkar, A., Webber, B. & Joshi, A. (2001) 'D-LTAG System: Discourse parsing with a lexicalized tree-adjoining grammar'. In *ESSLLI 2001 Workshop on Information Structure, Discourse Structure and Discourse Semantics*. Helsinki, Finland.

Forbes, K., Miltsakaki, E., Prasad, R., Sarkar, A., Webber, B. & Joshi, A. (2003) 'D-LTAG System: Discourse parsing with a lexicalized tree-adjoining grammar'. *Journal of Language, Logic and Information* **12**(3):261–279.

Gardent, C. (1997) *Discourse TAG*. University of Saarland, Saarbrucken, Claus report 89.

Gardent, C. & Kallmeyer, L. (2003) 'Semantic construction in feature-based TAG'. In *Proceedings of the 10th Meeting of the European Chapter of the Association for Computational Linguistics*. Budapest, Hungary.

Gawron, J. (1986) 'Situations and prepositions'. *Linguistics and Philosophy* **9**(3):327–382.

Grice, H. P. (1989) *Studies in the Way of Words*. Harvard University Press. Cambridge, MA.

Hardt, Daniel (1999) 'Dynamic interpretation of verb phrase ellipsis'. *Linguistics and Philisophy* **22**:187–221.

Heim, I. & Kratzer, A. (1998) *Semantics in Generative Grammar*. Blackwell. Malden, MA/Oxford.

Hovy, E. (1995) 'The multifunctionality of discourse markers'. In *Proceedings of the Workshop on Discourse Markers*. Egmond-aan-Zee. The Netherlands.

Jackendoff, R. (1990) 'Semantic structures'. In *Current Studies in Linguistics Series*. MIT Press. Cambridge, MA.

Jayez, J. & Rossari, C. (1998) 'Discourse relations versus discourse marker relations'. In *Proceedings of the ACL Workshop on Discourse Relations and Discourse Markers*. Montreal, Canada, 72–78.

Joshi, A. (1987) 'An introduction to tree adjoining grammar'. In A. Manaster-Ramer (ed.), *Mathematics of Language*. John Benjamins. Amsterdam, 87–114.

Joshi, A., Kallmeyer, L. & Romero, M. (2003) 'Flexible composition in LTAG: Quantifier scope and inverse linking'. In *Proceedings of the International Workshop on Compositional Semantics*. Tilburg, The Netherlands.

Joshi, A. & Vijay-Shanker, K. (1999) 'Compositional semantics with lexicalized tree-adjoining grammar (LTAG): How much underspecification is necessary?' In *Proceedings of the 3rd International Workshop on Computational Semantics*. Tilburg, The Netherlands.

Kallmeyer, L. (2002) 'Using an enriched TAG derivation structure as basis for semantics'. In *Proceedings of the TAG+6 Workshop*. University of Venice, 101–110.

Kallmeyer, L. & Joshi, A. (1999) 'Factoring predicate argument and scope semantics: Underspecified semantics with LTAG'. In *Proceedings of the 12th Amsterdam Colloquium*. Amsterdam, 169–174.

Kallmeyer, L. & Romero, M. (2004) 'LTAG semantics with semantic unification'. In *Proceedings of the TAG+7 Workshop*. Vancouver.

Kehler, A. (1995) *Interpreting Cohesive Forms in the Context of Discourse Inference*. Ph.D. dissertation. Harvard University, Boston, MA.

Knott, A. (1996) *A Data-Driven Methodology For Motivating A Set of Coherence Relations*. Ph.D. dissertation. University of Edinburgh, Edinburgh, Scotland.

Knott, A., Oberlander, J., O'Donnell, M. & Mellish, C. (2001) 'Beyond elaboration: The interaction of relations and focus in coherent text'. In J. Schilperoord T. Sanders & W. Spooren (eds), *Text representation: linguistic and psycholinguistic aspects*. John Benjamins Publishing Company. Amsterdam/Philadelphia. 181–196.

Kruijff-Korbayova, I. & Webber, B. (2001) 'Information structure and the interpretation of "otherwise" '. In *ESSLLI 2001 Worshop on Information Structure, Discourse Structure, and Discourse Semantics*. Helsinki, Finland, 61–78.

Lagerwerf, L. (1998) *Causal Connectives Have Presuppositions*. Holland Academic Graphics. The Hague.

Lapata, M. (2003) 'Probabilistic text structuring: Experiments with sentence ordering'. In *Proceedings of the 41st Meeting of the Association of Computational Linguistics*. Japan, 545–552.

Lyons, J. (1977) *Semantics*. Cambridge University Press. Cambridge, MA.

Mann, W. & Thompson, S. (1988) 'Rhetorical structure theory: Toward a functional theory of text organization'. *Text* **8**(3):243–281.

Marcu, D. (2000) 'The rhetorical parsing of unrestricted texts: A surface-based approach'. *Computational Linguistics* **26**(3):395–448.

Marcu, D. & Echihabi, A. (2002) 'An unsupervised approach to recognizing discourse relations'. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*. Philadelphia, PA.

Mauner, G., Tanenhaus, M. & Carlson, G. (1995) 'Implicit arguments in sentence processing'. *Journal of Memory and Language* **34**:357–382.

McConnell-Ginet, S. (1982) 'Adverbs and logical form'. *Language* **58**:144–184.

Miltsakaki, E., Prasad, R., Joshi, A. & Webber, B. (2004) 'Annotating discourse connectives and their arguments'. In *Proceedings of the NAACL/HLT Workshop on Frontiers in Corpus Annotation*. Boston, MA.

Moore, R. C. (1993) 'Events, situations and adverbs'. In R. Weischedel and M. Bates (eds), *Challenges in Natural Language Processing*. Cambridge University Press. Cambridge, UK.

Passonneau, R. J. (1991) 'Some facts about centers, indexicals, and demonstratives'. *Proceedings of the 29th Annual Meeting of the Assoc. for Computational Linguistics*. Berkeley. California, 63–70.

Penn-Treebank (1993) See www.ldc.upenn.edu/ldc/online/treebank/ for documentation.

Polanyi, L. & van den Berg, M. (1996) 'Discourse structure and discourse interpretation'. In *Proceedings of the 10th Amsterdam Colloquium*. University of Amsterdam. The Netherlands, 113–132.

Prasad, R., Miltsakaki, E., Joshi, A. & Webber, B. (2004) 'Annotation and data mining of the Penn Discourse

TreeBank'. In *ACL Workshop on Discourse Annotation*. Barcelona, Spain.

Quirk, R., Greenbaum, S., Leech, G. & Svartvik, J. (1985) *A Comprehensive Grammar of the English Language*. Longman, London.

Redeker, G. (1990) 'Ideational and pragmatic markers of discourse structure'. *Journal of Pragmatics* **14**:367–381.

Romero, M. & Kallmeyer, L. (2005) 'Scope and situation binding in LTAG using semantic unification'. In *Proceedings of the 7th International Workshop on Computational Semantics (IWCS)*. Tilburg.

Rooth, M. (1995) 'Indefinites, adverbs of quantification, and focus semantics'. In G.N. Carlson and F. J. Pelletier (eds), *The Generic Book*. University of Chicago Press.

Scha, R. & Polanyi, L. (1988) 'An augmented context free grammar for disocurse'. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING)*. Budapest, 573–577.

Schabes, Y. & Shieber, S. M. (1994) 'An alternative conception of tree-adjoining derivation'. *Computational Linguistics* **20**(1):91–124.

Schiffrin, D. (1987) *Discourse Markers*. Cambridge University Press. Cambridge, MA.

Schilder, F. (1997) 'Discourse tree grammar, or how to get attached to a discourse'. In *International Workshop on Computational Semantics*. The Netherlands, 261–273.

Steedman, M. (2000) 'Information Structure and the syntax-phonology interface'. *Linguistic Inquiry* **31**(4):649–689.

Stone, M. (1994) *Discourse Deixis, Discourse Structure, and the Semantics of Subordination*. Unpublished MS.

van Dijk, T. A. (1979) 'Pragmatic connectives'. *Journal of Pragmatics* **3**: 447–456.

Vendler, Z. (1967) *Linguistics in Philosophy*. Cornell University Press. Cornell, NY.

Verspoor, C. M. (1997) *Contextually-Dependent Lexical Semantics*. Ph.D. dissertation. University of Edinburgh, Edinburgh, Scotland.

Webber, B. (1991) 'Structure and ostension in the interpretation of discourse deixis'. *Language and Cognitive Processes* **6**(2):107–135.

Webber, B. & Joshi, A. (1998) 'Anchoring a lexicalized tree-adjoining grammar for discourse'. In *COLING/ACL Workshop on Discourse Relations and Discourse Markers*. Montreal, 86–92.

Webber, B., Joshi, A., Stone, M. & Knott, A. (1999a) 'Discourse relations: A structural and presuppositional account using lexicalised TAG'. In *Proceedings of the 36th Meeting of the Assoc. for Computational Linguistics (ACL)*. College Park, MD, 41–48.

Webber, B., Joshi, A., Stone, M. & Knott, A. (2003) 'Anaphora and discourse structure'. *Computational Linguistics* **29**(4):547–588.

Webber, B., Knott, A. and Joshi, A. (1999b) 'Multiple discourse connectives in a lexicalised grammar for discourse'. In H. Bunt, R. Muskens & E. Thijsse (eds), *Computing Meaning*, volume 2. Kluwer Academic Publishers. London, 229–245.

Weir, D. J. (1988) *Characterizing Mildly Context-Sensitive Grammar Formalisms*. Ph.D. dissertation. University of Pennsylvania. Philadelphia, PA.

XTAG, Research Group (2001) 'A lexicalized tree adjoining grammar for English'. Technical Report IRCS-01-03, IRCS, University of Pennsylvania.