Multi-Component Tree Adjoining Grammars, Dependency Graph Models, and Linguistic Analyses

Joan Chen-Main* and Aravind K. Joshi*+
*Institute for Research in Cognitive Science, and
+Dept of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104-6228
{chenmain,joshi}@seas.upenn.edu

Abstract

Recent work identifies two properties that appear particularly relevant to the characterization of graph-based dependency models of syntactic structure[1]: the absence of interleaving substructures (*well-nestedness*) and a bound on a type of discontinuity (*gap-degree* ≤ 1) successfully describe more than 99% of the structures in two dependency treebanks (Kuhlmann and Nivre 2006).[2] Bodirsky et al. (2005) establish that every dependency structure with these two properties can be recast as a lexicalized Tree Adjoining Grammar (LTAG) derivation and vice versa. However, multi-component extensions of TAG (MC-TAG), argued to be necessary on linguistic grounds, induce dependency structures that do not conform to these two properties (Kuhlmann and Möhl 2006). In this paper, we observe that several types of MC-TAG as used for linguistic analysis are more restrictive than the formal system is in principle. In particular, tree-local MC-TAG, tree-local MC-TAG with *flexible composition* (Kallmeyer and Joshi 2003), and special cases of set-local TAG as used to describe certain linguistic phenomena satisfy the well-nested and gap degree ≤ 1 criteria. We also observe that gap degree can distinguish between prohibited and allowed *wh*-extractions in English, and report some preliminary work comparing the predictions of the graph approach and the MC-TAG approach to scrambling.

1 Introduction

Bodirsky et al. (2005) introduce a class of graphical dependency models, called *graph drawings* (which differ from standard dependency structures), that are equivalent to lexicalized Tree Adjoining Grammar (LTAG) derivations (Joshi and Schabes 1997). Whereas TAG is a generative framework in which each well-formed expression corresponds with a legitimate derivation in that system, the graph drawing approach provides a set of structures and a set of constraints on well-formedness. Bodirsky et al. offer the class of graph drawings that satisfy these constraints as a model-based perspective on TAG. Section 2 summarizes this relationship between TAG derivations and these graph drawings.

In related work, Kuhlmann and Nivre (2006) evaluate a number of constraints that have been proposed to restrict the class of dependency structures characterizing natural language with respect to two dependency treebanks: the Prague Dependency Treebank (PDT) (Hajič et al., 2001) and the Danish Dependency Treebank (DDT) (Kromann, 2003). The results indicate that two properties provide good coverage of the structures in both

[1] Whereas weak equivalence of grammar classes is only concerned with string sets and fails to shed light on equivalence at the structural level, our work involves the equivalence of derivations and graph based models of dependencies. Thus, our work is relevant to certain aspects of grammar engineering that weak equivalence does not speak to.

[2] These properties hold for many of the so-called non-projective dependency structures and the corresponding non-context free structures associated with TAG, further allowing CKY type dynamic programming approaches to parsing to these dependency graphs.

Proceedings of the 5th Workshop on Important Unresolved Matters, pages 1–8,

treebanks.[3] The first is a binary *well-nestedness* constraint.[4] The other is a bound on *gap degree*, a graded measure of discontinuity. These results are given in Table 1. What is noteworthy is that the graph drawings which correspond to LTAG derivations share these two properties: LTAG induced graph drawings are both well-nested and have gap degree ≤ 1, and for every graph drawing that is both well-nested and gap degree ≤ 1, there exists a corresponding LTAG derivation (Möhl 2006). In section 3, these two properties are defined.

property	Danish Dep. Treebank	Prague Dep. Tree-bank
all structures	n = 4393	n = 73088
well-nested	99.89%	99.89%
gap degree 0	84.95%	76.85%
gap degree 1	14.89%	22.72%
gap degree ≤ 1	99.84%	99.57%

Table 1. Relevant results from Kuhlmann and Nivre (2006).

In section 4, we show that gap degree can be used to distinguish between strong island violations and weak island violations in English. This supports the notion that gap-degree is a linguistically relevant measure.

Although TAG is a linguistically expressive formalism, a closer look at the linguistic data has motivated extensions of TAG.[5] One of the most widely used extensions for handling cases that are difficult for classic TAG is Multi-Component TAG (Weir 1988). Like basic TAG, MC-TAG is a formalism for rewriting nodes of a tree as other trees. The set of underived trees are called *elementary trees*. The rewriting is accomplished via two operations: *substitution*, which rewrites a leaf node labeled X with a tree rooted in a node labeled X, and *adjoining*, which rewrites a node X with a tree that labels both its root and a distinguished leaf node, the *foot*, with X. The observation that linguistic dependencies typically occur within some sort of local domain is expressed in the TAG hypothesis that all such dependencies occur within the basic building blocks of the grammar. Recursive structure is "factored out," which allows apparent non-local dependencies to be recast as local ones. Whereas basic TAG takes the basic unit to be a single elementary tree, MC-TAG extends the domain of locality to encompass a set of elementary trees. That is, these sets are the objects over which the combinatory operations apply. The MC-extension allows for linguistically satisfying accounts for a number of attested phenomena, such as: English extraposition (Kroch and Joshi 1986), subj-aux inversion in combination with raising verbs (Frank 1992), anaphoric binding (Ryant and Scheffler 2006), quantifier scope ambiguity (Joshi et al. 2003), clitic climbing in Romance (Bleam 1994), and Japanese causatives (Heycock 1986).

The primary concern of this paper is the reconciliation of the observation noted above, that MC-TAG appears to be on the right track for a good generative characterization of natural language, with a second observation: The graph drawings that correspond to MC-TAG derivations, are not guaranteed to retain the properties of basic-TAG induced graph drawings. Kuhlmann and Möhl (2006) report that if an entire MC set is anchored by a single lexical element (the natural extension of "lexicalization" of TAGs to MC-TAGs), then the class of dependency structures is expanded with respect to both conditions that characterized the TAG-induced graph drawings: MC-TAG induced graph drawings include structures that are not well-nested, have gap degree > 1, or both. As Kuhlmann and Möhl point out, the gap degree increases with the number of components, which we will elaborate in section 6. This is true even if we require that all components of a set combine with a single elementary tree (i.e. tree-local MC-TAG, which is known to allow more derivation structures (i.e. derivation trees) than TAG, although they generate the same set of derived trees). If we suppose that the characterization of dependency structures as reported by Kuhlmann and Nivre (2006) for Czech and Danish extends cross-linguistically, i.e. the dependency structures for natural language falls within the class of well-nested and gap degree ≤ 1 dependency structures, then MC-TAG appears to correspond to the wrong class of model-theoretic dependency structures. It is desirable to account for the apparent mismatch.

One possibility is that the linguistic analyses that depend on a multi-component approach are ex-

[3] A third property based on *edge degree* also characterizes the structures, but has no clear relationship to TAG-derivations. Thus, reference to it is omitted in the remaining text. See Kuhlmann and Nivre (2006) for the definition of edge degree.
[4] Well-nestedness differs from *projectivity*. (See section 3.)
[5] For a readable introduction, see Chapter 1 of Frank (2002).

tremely infrequent, and that this is reflected in the small proportion (< 1%) of data in the PDT and DDT that are not both well-nested and gap degree ≤ 1. A second possibility is that the structures in the PDT and DDT are actually not good representatives of the structures needed to characterize natural languages in general. However, a look at the cases in which MC-TAG is employed reveals that these particular analyses yield derivations that correspond to graph drawings that do satisfy well-nestedness and have gap degree ≤ 1. In practice, MC-TAG seems to be used more restrictively than what the formal system allows in principle. This keeps the corresponding graph drawings within the class of structures identified by Bodirsky et al. (2005) as a model of TAG derivations, and by Kuhlmann and Nivre (2006) as empirically relevant. Lastly, we compare the scrambling patterns that are possible in an MC-TAG extension with those that conform to the well-nestedness and gap degree ≤ 1 properties of the graph approach.

2 TAG-induced Graph Dependencies

The next two sections are intended to provide an intuition for the terms defined more formally in Bodirsky et al. (2005) and Kuhlmann and Nivre (2006). In the former, the authors define their dependency structures of interest, called *graph drawings*, as a three-tuple: a set of nodes, a dominance relation, and a (total) precedence relation. These dependency structures are based on information from both a TAG-derivation and that derivation's final phrase structure. The anchor of each elementary tree of a strictly lexicalized TAG (LTAG) is used as a node label in the induced dependency structure. E.g. suppose tree A is anchored by lexical item *a* in the LTAG grammar. Then *a* will be a node label in any dependency structure induced by an LTAG derivation involving tree A.

To see how the dominance relation and precedence relation mirror the derivation and the final derived phrase structure, let us further suppose that LTAG tree B is anchored by lexical item *b*. Node *a* dominates node *b* in the dependency structure iff Tree A dominates tree B in the derivation structure. (I.e. tree B must substitute or adjoin into tree A during the TAG-derivation.[6]) Node *a* precedes node *b* in the dependency structure iff *a* linearly precedes *b* in the derived phrase structure tree.

An example based on the cross-serial dependencies seen in Dutch subordinate clauses is given in Figure 1. In the graph drawing in (4), the four nodes names, {*Jan, de kinderen, zag, zwemmen*}, are the same set as the anchors of the elementary trees in (1), which is the same as the set of terminals in (3), the derived phrase structure. The ordering of these nodes is exactly the ordering of the terminals in (3). The directed edges between the nodes mirrors the immediate dominance relation represented in (2), the derivation structure showing how the trees in (1) combine. E.g. Just as the *zwemmen* node has the *zag* and *de kinderen* nodes as its two children in (2), so does the *zwemmen* node dominate *zag* and *de kinderen* in (4).

Möhl (2006) provides the formal details showing that such LTAG-induced dependency structures have the properties of being 1) well-nested and 2) gap degree ≤ 1, and, conversely, that any structures with these properties have a corresponding LTAG derivation.[7] These properties are defined in the next section.

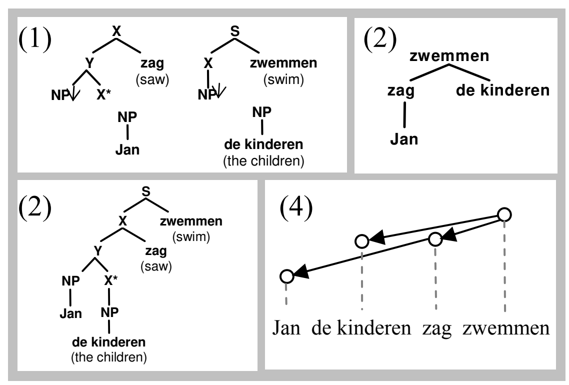


Figure 1. Derivation for *Jan de kinderen zag zwemmen* and corresponding graph drawing

3 Properties of Dependency Graphs

3.1 Gap-Degree

It will be useful to first define the term *projection*.

Definition: The projection of a node x is the set of nodes dominated by x (including x). (E.g. in (4), the projection of *zag* = {*Jan*, *zag*}.)

[6] Whereas in standard dependency graphs, adjunction of t2 to t1 generally corresponds to a dependency directed from t2 to t1, in a TAG-induced dependency graph, adjoining t2 to t1 corresponds to the reverse dependency.

[7] This result refers to single graph drawings and particular LTAG derivation. See Kuhlmann and Möhl (2007) on the relationship between sets of graph drawings and LTAGs.

Recall that the nodes of a graph drawing are in a precedence relation, and that this precedence relation is total.

Definition: A gap is a discontinuity with respect to precedence in the projection of a node in the drawing. (E.g. in (4), *de kinderen* is the gap preventing *Jan* and *zag* from forming a contiguous interval.)

Definition: The gap degree of a node is the number of gaps in its projection. (E.g. the gap degree of node *zag* = 1.)

Definition: The gap degree of a drawing is the maximum among the gap degrees of its nodes. (E.g. in (4), only the projection of *zag* is interrupted by a gap. Thus, the gap degree of the graph drawing in (4) = 1.)

In TAG drawings, a gap arises from an interruption of the dependencies in an auxiliary tree. If B is adjoined into A, the gap is the material in A that is below the foot node of B. E.g. in figure 1, *De kinderen* is substituted into the *zwemmen* tree below the node into which the *zag* tree adjoins into the *zwemmen* tree. Thus, *de kinderen* interrupts the pronounced material on the left of the *zag* tree's foot node, *Jan*, from the pronounced material on the right of the foot node, *zag*.

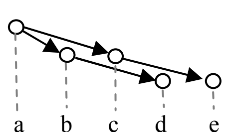
3.2 Well-Nestedness

Definition: If the roots of two subtrees in the drawing are not in a dominance relation, then the trees are *disjoint*. (E.g. in (5), the subtrees rooted in *b* and *c* are disjoint, while the subtrees rooted in *a* and *b* are not.)

Definition: If nodes x_1, x_2 belong to tree X, nodes y_1, y_2 belong to tree Y, precedence orders these nodes: $x_1 > y_1 > x_2 > y_2$, and X and Y are disjoint, then trees X and Y *interleave*. (E.g. in (5), *b* and *d* belong to the subtree rooted in *b*, while *c* and *e* belong to the subtree rooted in *c*. These two subtrees are disjoint. Since the nodes are ordered $b > c > d > e$, the two trees interleave.)

Definition: If there is no interleaving between disjoint subtrees, then a graph drawing is *well-nested*. (e.g. (4) is well-nested, but (5) is not)

(5)



Non-well nested graph drawing

4 Island Effects and Gap-Degree

When standard TAG analyses of island effects are adopted (see Frank 2002), we observe that differences in gap degree align with the division between *wh*-extractions that are attested in natural language (grammatical *wh*-movement and weak island effects) and those claimed to be prohibited (strong island effects). Specifically, four strong island violations, extraction from an adverbial modifier, relative clause, complex NP, or subject, correspond to structures of gap degree 1, while cyclic *wh*-movement and a weak island violation (extraction from a *wh*-island) are gap degree 0 in English. Interestingly, while it is clear that weak islands vary in their island status from language to language, strong islands have been claimed to block extraction cross-linguistically. We tentatively postulate that gap degree is useful for characterizing strong islands cross-linguistically.

An example is given in (6), a standard TAG derivation for adverbial modification: the *after*-tree adjoins into the *buy*-tree (the matrix clause), the *got*-tree substitutes into the *after*-tree, and the two arguments *who* and *a-raise* substitute into the *got*-tree. In (7), the corresponding dependency structure, the projection of *got* includes *who*, which is separated from *got* by the string comprising the matrix clause and adverbial. Clearly, we do not want to claim that any gap degree of 1 is a sure source of ungrammaticality. However, it is possible that a gap degree of 1 in conjunction with a *wh*-element yields ungrammaticality. For the particular set of islands we examined, we postulate that the projection of the node immediately dominating the *wh*-element is prohibited from containing gaps.

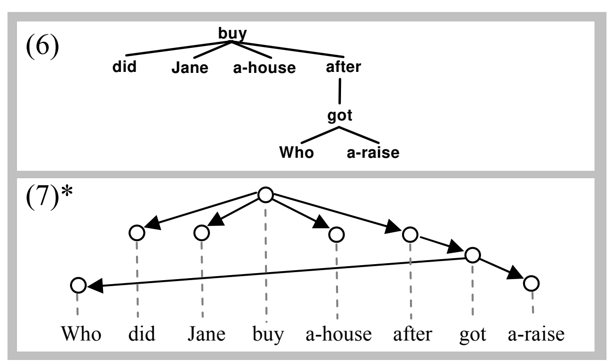


Figure 2. LTAG derivation and graph drawing for **Who did Jane buy a house after got a raise?*

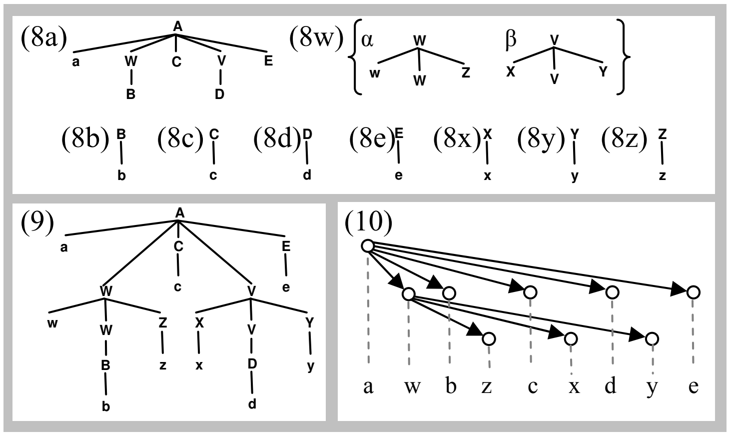


Figure 3. MC-TAG induced graph drawing of gap degree 3

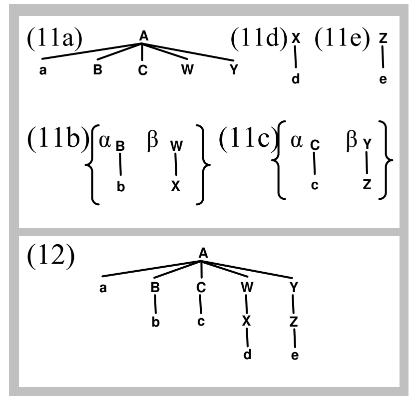


Figure 4. Non-well-nested MC-TAG induced graph drawing

5 MC-TAG-induced Dependency Graphs

5.1 Gap-Degree Beyond 1

As reviewed in section 3, the source of every gap in a TAG drawing comes from an interruption of the dependencies in an auxiliary tree. Since the auxiliary tree only has one foot, it only has a slot for a single gap. A MC-set, however, could be comprised of two auxiliary trees. This means there are slots for two gaps, one associated with each foot. Furthermore, a gap may arise as a result of any pronounced material between the two components. Thus, when we already have at least one foot, adding an additional foot increases the maximum gap degree by 2. The maximum gap degree = 1 + 2(n – 1) = 2n – 1, where n is the max # of foot nodes in any elementary tree set.

As an example, consider the composition of the trees in (8), Figure 3 (Kuhlmann, p.c.) The tree set in (8w) is comprised of two auxiliary trees. One tree, (8wα), adjoins into (8a), and a gap is created by the material in (8a) that falls below the foot node of (8wα), namely *b*. When (8wβ) is adjoined into (8α) at node V, a second gap is created below (8wβ) by *d*. A third gap is created by the material between the two components. (9) shows the derived phrase structure, and (10), the corresponding graph drawing. The projection of node *w*, {*w*, *x*, *y*, *z*} has three discontinuities, nodes *b*, *c*, and *d*.

5.2 Non-Well-Nestedness

Kuhlmann and Möhl (2006) show that even a tree-local MC-TAG that allows only substitution can induce a non-well-nested graph drawing. Figure 4 replicates their example. This derivation involves two MC-sets, (11b) and (11c). The tree anchored by d, (11d), substitutes into the second component of the set anchored by b, (11b). Similarly, the tree anchored by e, (11e), substitutes into the second component of the set anchored by c, (11c). Both MC-sets compose into the tree anchored by a, yielding the derived phrase structure in (12). The corresponding graph drawing is exactly our earlier example of non-well-nestedness in (5).

6 MC-TAG in Practice

We now turn to cases in which linguists have used MC-TAGs to account for cases argued to have no satisfying solution in basic TAG. Unlike the examples in 5.1 and 5.2, these particular MC-derivations correspond to dependency structures that are well-nested and have gap degree ≤ 1. Table 2 summarizes these cases. The last column indicates the type of MC-extension assumed by the analysis: tree-local MC-TAGs, tree-local MC-TAGs with *flexible composition*, the mirror operation to adjoining; if tree α adjoins into tree β, the combination can be alternatively viewed as tree β "flexibly" composing with tree α (Joshi et al. 2003, Kallmeyer and Joshi 2003)[8], and set-local MC-TAGs. Set-local MC-TAGs are generally more powerful than TAGs, but since these particular cases induce well-nested graph drawings of gap degree ≤ 1, we can conclude that set-local MC-TAG as used in

[8] I.e. When composing A and B, we can take A as the function and B as the argument or vice versa. For CFGs, such flexibility has no added benefit. For categorical type grammars, this kind of flexibility is accomplished via type raising, which allows for some new types of constituents but does not give rise to any new word orders. For tree local MC-TAGs, such flexibility does allow more word orders (permutations) to be generated than are possible without flexible composition.

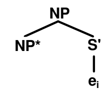
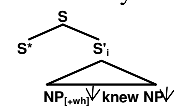
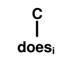
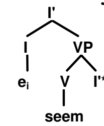
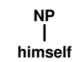
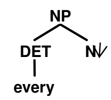
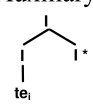
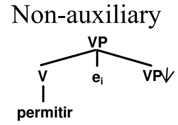
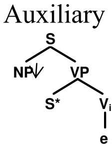
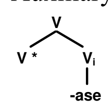
analysis source	phenomenon	first component	second component	MC-type
Kroch and Joshi 1986	English extraposition *A man arrived who knew Mary.*	Auxiliary 	Auxiliary 	Tree-local
Frank 1992	subj-aux inversion with raising verb constructions *Does Gabriel seem to like gnocchi?*	Non-auxiliary 	Auxiliary 	Tree-local
Ryant and Scheffler 2006	anaphoric binding *John$_i$ likes himself$_i$.*	Auxiliary NP*	Non-auxiliary 	Tree-local + flexible composition
Joshi, Kallmeyer, & Romero 2003	quantifier scope ambiguity *An FBI agent is spying on every professor.* $(\forall y\ [\mathrm{prof}(y) \rightarrow \exists x\ [\mathrm{agent}(x) \wedge \mathrm{spy}\ (x, y)]\ ])$ OR $(\exists x\ [\mathrm{agent}(x) \wedge \forall y\ [\mathrm{prof}(y) \rightarrow \mathrm{spy}\ (x, y)]\ ])$	Auxiliary S*	Non-auxiliary 	Tree-local + flexible composition
Bleam 1994	clitic climbing in Romance *Mari telo quiere permitir ver.* Mari you-it wants to permit to see "Mari wants to permit you to see it."	Auxiliary 	Non-auxiliary 	Set-local
Heycock 1986	Japanese causatives *Watasi-wa Mitiko-ni Taroo-o ik –ase (–sase) –ta.* I TOP DAT ACC go –CS –CS –PST "I made Mitiko make Taroo go."	Auxiliary 	Auxiliary 	Set-local

Table 2. Canonical tree sets used in MC-TAG analyses of several phenomena

these cases is weakly equivalent to TAG.

From Table 2, we can draw two generalizations. First, in an MC-TAG analysis, a two-component set is typically used. One of the trees is often a very small piece of structure that corresponds to the "base position," surface position, or scope position of a single element. Second, the auxiliary tree components typically have elements with phonological content only on one side of the foot.

At this point, we make explicit an assumption that we believe aligns with Bodirsky et al. (2005). Since silent elements, such as traces, do not anchor an elementary tree, they do not correspond to a node in the dependency structure.

6.1 Why the Gap-Degree Remains ≤ 1

Recall that in example (8), each of the two components in the example MC-TAG has a foot with phonological material on both sides, giving rise to two gaps, and a third gap is created via the material between the two components. In contrast, in the MC-TAG sets shown in Table 2, the auxiliary trees have pronounced material only on one side of the foot node. This eliminates the gap that would have arisen due to the interruption of material on the left side of the foot from the right side of the foot as a result of the pronounced material beneath the foot. The only way to obtain pronounced material on both sides of the foot node is to adjoin a component into one of these auxiliary trees. Interestingly, the set-local analyses (in which all components of a set must combine with components of a single set vs. tree-local MC-TAG) for clitic climbing and Japanese causatives do posit recursive components adjoining into other recursive components, but only while maintaining all pronounced material on one side of the foot. In the absence of a derivational step resulting in pronounced material on both sides of a foot, the only remaining possible gap is that which arises from pronounced material that appears between the two components.

Note that the observation about the position of pronounced material applies only to auxiliary trees in sets with multiple components. That is, auxiliary trees that comprise a singleton set may still have pronounced material on both sides of the foot.

6.2 Why the Structures Remain Well-Nested

Since Kuhlmann and Möhl (2006) show that even a MC-TAG that allows only non-auxiliary trees in MC-sets will expand the drawings to include non-well-nested drawings, there is no way to pare back the MC-TAG via restrictions on the types of trees allowed in MC-sets so as to avoid interleaving.

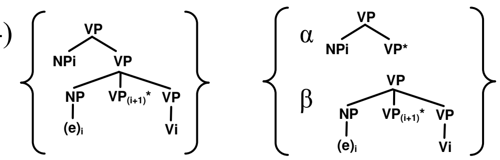
Recall that to satisfy the definition of interleaving, it is necessary that the two MC-sets are not in any dominance relation in the derivation structure. In Kuhlmann and Möhl's example, this is satisfied because the two MC-sets are sisters in the derivation; they combine into the same tree. In the linguistic analyses considered here, no more than one MC-set combines into the same tree. For tree-local MC-TAG, it appears to be sufficient to bar more than one MC-set from combining into a single tree.

7 MC-TAG and Scrambling

In subordinate clauses in Standard German, the canonical order of verbs and their subject arguments is a nested dependency order. However, other orderings are also possible. For example, in the case of a clause-final cluster of three verbs, the canonical order is as given in (13), $NP_1NP_2NP_3$ $V_3V_2V_1$, but all the other permutations of the NP arguments are also possible orderings. All six permutations of the NPs can be derived via tree-local MC-TAG. From the graph-model perspective adopted here, this is unsurprising: All the sequences are well-nested and have gap degree ≤ 1.

(13) NP_1 NP_2 NP_3 V_3 V_2 V_1
. . . Hans Peter Marie schwimmen lassen sah
. . . Hans Peter Marie swim make saw
“ . . . Hans saw Peter make Marie swim.”

However, with an additional level of embedding, i.e. four NPs and four verbs, the situation is different, both linguistically and formally. Our focus is on making the formal predictions of a linguistically informed system precise. We start with a tree-local MC-TAG that is restricted to linguistically motivated tree-sets and to semantically coherent derivations. The former linguistic restriction is illustrated in (14), the possible tree-sets anchored by a verb that takes a VP argument. The latter linguistic restriction is that there is no semantic feature clash at any stages of the derivation: the VP argument of V_i must be associated with V_{i+1}.

(14)


Single and two-component sets for V_i

As MC-TAG is enriched in various ways (by allowing flexible composition, multiple adjoining at the same node, and/or components from the same MC-set to target the same node), all 24 orderings where the nouns permute while the verbs remain fixed can be derived. (We are aware that German also allows verbs to scramble.) Taking the dependency structures of these sequences to consist of an edge from each verb V_i to its subject NP and to the head of its argument VP, V_{i+1}, we can compare the predictions of the graph drawing approach and the MC-TAG approach. It turns out that the permutations of gap degree ≤ 1 and those of gap-degree 2 do not align in an obvious way with particular enrichments. For example, $NP_4NP_2NP_3NP_1V_4V_3V_2V_1$ (gap degree 2) is derivable via basic tree-local MC-TAG, but $NP_3NP_1NP_4NP_2V_4V_3V_2V_1$ and $NP_3NP_2NP_4NP_1V_4V_3V_2V_1$ (also gap degree 2) appear to require both flexible composition and allowing components from the same MC-set to target the same node.

8 Conclusion and Future Work

This paper reviews the connection established in previous work between TAG derivations and model-theoretic graph drawings, i.e. well-nested dependency structures of gap degree ≤ 1, and reports several observations that build on this work. First, additional evidence of the linguistic relevance of the gap degree measure is given. The gap degree measure can distinguish *wh*-movement that is assumed to be generally disallowed from *wh*-movement that is permitted in natural language. Second, we observe that the graph drawings induced by MC-TAGs used in linguistic analyses continue to fall within the class of well-nested, gap degree $\leq$ 1 dependency structures. While Kuhlmann and Möhl (2006) show that MC-TAGs in which each set has a single lexical anchor induce graph drawings that are outside this class, this extra complexity in the dependency structures does not appear to be utilized. Even for the crucial cases used to argue for MC-extensions, MC-TAG is used in a manner requiring less complexity than the formal system allows. Examining these particular grammars lays the groundwork for identifying a natural class of MC-TAG grammars whose derivations correspond to well-nested graph drawings of gap degree $\leq$ 1. Specifically, the observations suggest the class to be MC-TAGs in which 1) component sets have up to two members, 2) auxiliary trees that are members of non-singleton MC-sets have pronounced material on only one side of the foot, whether the auxiliary member is derived or not, and 3) up to one MC-set may combine into each tree. Though these constraints appears stipulative from a formal perspective, a preliminary look suggests that natural language will not require their violation. That is, we may find linguistic justification for these constraints. Lastly, in ongoing work, we explore how allowing flexible composition and multiple adjoining enables MC-TAGs to derive a range of scrambling patterns.

References

Tonia Bleam. 2000. Clitic climbing and the power of Tree Adjoining Grammar. In A. Abeillé and O. Rambow (eds.), *Tree Adjoining Grammars: formalisms, linguistic analysis and processing*. Stanford: CSLI Publications, 193-220. (written in 1994).

Manuel Bodirsky, Marco Kuhlmann, and Mathias Möhl. 2005. Well-nested drawings as models of syntactic structure. In *10th Conference of Formal Grammar and 9th Meeting on Mathematics of Language (FG-MoL)*, Edinburgh, UK.

Robert Frank. 1992. *Syntactic Locality and Tree Adjoining Grammar: grammatical, acquisition, and processing perspectives*. PhD dissertation, University of Pennsylvania, Philadelphia, USA.

Robert Frank. 2002. *Phrase Structure Composition and Syntactic Dependencies*. MIT Press.

Jan Hajič, Barbora Vidova Hladka, Jarmila Panevová, Eva Hajičová, Petr Sgall, and Petr Pajas. 2001. Prague Dependency Treebank 1.0. LDC, 2001T10.

Caroline Heycock. 1986. The structure of the Japanese causative. Technical Report MS-CIS-87-55, University of Pennsylvania.

Aravind K. Joshi, Laura Kallmeyer, and Maribel Romero. 2003. Flexible composition in LTAG: quantifier scope and inverse linking. In H. Bunt and R. Muskens (eds.), *Computing Meaning* 3. Dordrecht: Kluwer.

Aravind K. Joshi and Y. Schabes. 1997. Tree-Adjoining Grammars. In G. Rozenberg and A. Salomaa (eds.): *Handbook of Formal Languages*. Berlin: Springer, 69–123.

Laura Kallmeyer and Aravind K. Joshi. 2003. Factoring predicate argument and scope semantics: underspecified semantics with LTAG. *Research on Language and Computation* 1(1-2), 3-58.

Anthony Kroch and Aravind K. Joshi. 1990. Extraposition in a Tree Adjoining Grammar. In G. Huck and A. Ojeda, eds., *Syntax and Semantics: Discontinuous Constituents*, 107-149.

Matthias Trautner Kromann. 2003. The Danish Dependency Treebank and the DTAG treebank tool. In *2nd Workshop on Treebanks and Linguistic Theories (TLT)*, 217-220.

Marco Kuhlmann and Mathias Möhl. 2006. Extended cross-serial dependencies in Tree Adjoining Grammars. In *Proceedings of the 8th International Workshop on Tree Adjoining Grammar and Related Formalisms*, Sydney, Australia, 121-126.

Marco Kuhlmann and Mathias Möhl. 2007. Mildly context-sensitive dependency languages. In *45th Annual Meeting of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic.

Marco Kuhlmann and Joakim Nivre. 2006. Mildly non-projective dependency structures. In *21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL), Companion Volume*, Sydney, Australia.

Mathias Möhl. 2006. *Drawings as Models of Syntactic Structure: Theory and Algorithms*, Masters thesis, Saarland University, Saarbrücken, Germany.

Neville Ryant and Tatjana Scheffler. 2006. Binding of anaphors in LTAG. In *Proceedings of the 8th International Workshop on Tree Adjoining Grammar and Related Formalisms*, Sydney, Australia, 65-72.

David Weir. 1988. *Characterizing mildly context-sensitive grammar formalisms*. PhD dissertation, University of Pennsylvania, Philadelphia, USA.