An alternative description of extractions in TAG

KAHANE Sylvain, CANDITO Marie-Hélène & DE KERCADIO Yannick

LaTTiCe - TALaNa
sk@ccr.jussieu.fr

LexiQuest
marie-helene.candito@lexiquest.fr

LIMSI
kercadio@limsi.fr

Abstract

The aim of the paper is to propose a new description of extraction in plain TAG. Contrary to Kroch 1987's analysis, our description is based on the fact that the power of a relative clause to adjoin on a noun can be attached to the wh-word rather than to a verb. This analysis solves some problems of the previous analysis, notably by giving the right semantic dependency in case of pied-piping.

We are thankful to our two reviewers for many valuable comments.

Introduction

The only description of extractions in TAG we know has been developed by Kroch & Joshi (1986), Kroch (1987) and implemented in the developed grammars of English (XTAG 1995) and French (Abeillé 1991, Candito 1999). This implementation solves the unboundedness of extractions with predicative adjoining, but the pied-piping is solved using a special feature. We think that this solution of pied-piping is not absolutely convenient, because some edge of the derivation tree cannot be interpreted as semantic dependency (Candito & Kahane 1998). Our assumption is based on the fact that a TAG derivation tree can be interpreted as a semantic graph, that is a predicate-argument structure. Moreover this implementation fails to describe some cases of extraction, such as some French *dont*-relatives. We propose a new description of extraction in TAG which solve most of these problems. Nevertheless, our study must rather be appreciated as an investigation of the limits of the TAG formalism, because we think that TAG is not the most appropriate framework for the implementation of our description of extractions. The same analysis is more suitably implemented in GAG/DTG (Candito & Kahane 1998).

1. Semantic dependencies

The meaning of a sentence comes from the combination of the meaning of the lexical units of the sentence. A lexical meaning or **semanteme** can be considered as a semantic functor or predicate. For instance, consider:

(1) *Peter often saw black cats.*

In (1), the meaning 'see' is a binary functor whose argument are 'Peter' and 'cat', whereas 'often' and 'black' are unary functors with respectively 'see' and 'cat' as arguments. This predicate-argument structure can be represented by a graph (Fig. 1), called a **semantic graph** (Zolkovski & Mel'çuk 1967, Mel'çuk 1988). An edge of such a graph is called a **semantic dependency**. The two extremities of a semantic dependency are called the **semantic governor** and **the semantic argument**. A semantic graph can be converted into a logical formula by reification : for each semanteme a variable is introduced as first argument of the predicate; this variable is used by other predicates pointing on it in the semantic graph. The semantic graph of Fig. 1 is thus converted in the formula:

'Peter'(x) & 'cat'(y) & 'black'(p,y) & 'see'(e,x,y) & 'often'(q,e)

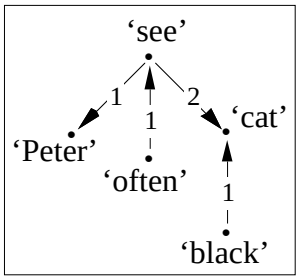

Fig. 1. The semantic graph of (1)

2. Principles for our TAG

We assume the following linguistic properties for elementary trees. The elementary trees correspond to exactly one semantic unit (Abeillé 1991), and respect the predicate-argument co-occurrence principle (PACP), with a semantic interpretation (Candito & Kahane 1998, Candito 1999): semantic predicates anchor trees with positions for the syntactic expression of *all and only* their semantic arguments.[1] It is important to note that the PACP concerns any position to extend, whether substitution or foot node.

Therefore, **the arcs of a TAG derivation tree can be interpreted as semantic dependencies**. In the following, substitution arcs will be represented by down arrows and adjoining arcs, by up arrows. The label on an arrow indicates the position of the semantic argument in the predication (first, second...). A last word about complementizers: as noted by Tesnière (1959), which called them *translatifs*, they are grammatical words that mark a link between two words. Contrary to Franck 1992, we think that complementizers must be attached to the SEMANTIC governor, that is the word that controls the link. For instance, in *Peter thinks that Mary likes beans*, *that* will be a co-anchor of the elementary tree anchored by *thinks*—the semantic governor of *likes*—, while in *the beans that Mary likes*, *that* will be a co-anchor of *likes*—the semantic governor of *beans*. See our solution of *qui*/*que* alternation of the complementizer in French for an illustration of this principle (Fig.14).

The plain TAG formalism constrains adjoining in the following manner: the root and foot nodes of an auxiliary tree β must be of same categories. It follows that, in a predicative adjunction, the anchor of β and the semantic argument on which β adjoins must be of same categories. In order to allow predicative adjunction on a semantic argument of a different category this constraint must be relaxed. Although it is well known that it does not modify the generative power (Vijay-Shanker 1987, 1992), we do not think that it was really used for linguistic descriptions in TAG.[2] The solution simply consists in considering categories as top and bottom features. In this case, all nodes will have a same transparent category X and real syntactic categories will only appear in top and bottom features. The following notation will be adopted: [A|B] := [X,t:A,b:B]. For the sake of simplicity, a node with same top and bottom categories A will be noted A: A := [A|A]. Note that a node that has different top and bottom categories has to receive an adjunction. This little change in the formalism (which does not change the generative power) allows new linguistic descriptions. Before going to the extraction, we will study the case of determiners, predicative adjectives and *tough*-movement.

---

[1] This counts for expressed semantic arguments only, so not for the agent in agentless passive constructions for instance. Moreover this principle cannot be respected to handle control cases, for which there is a cycle in the semantic graph, as in *Bill wants to sleep*. Nevertheless different formal devices can be developed to recover both semantic dependencies between *want* and *Bill* and between *sleep* and *Bill*.

[2] It can be noted that it was done in other formalisms of the TAG family such as DTG (Rambow *et al*. 1995).

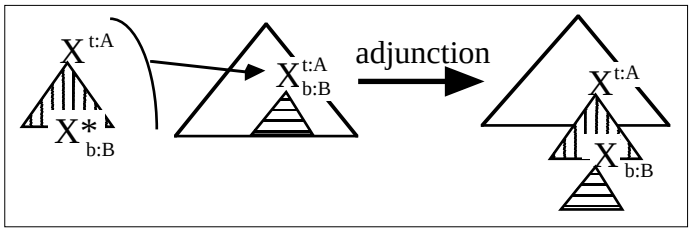

Fig. 2. Adjunction and top and bottom features

Determiner. In TAG, it is usual to consider that the determiner adjoins on the noun, which gives us the right semantic dependencies. Nevertheless, in usual TAG, this analysis needs to attribute the same categories to a phrase with and without a determiner and to distinguish them by a special feature (generally called [det]). It is now possible to use different categories (Fig. 3).

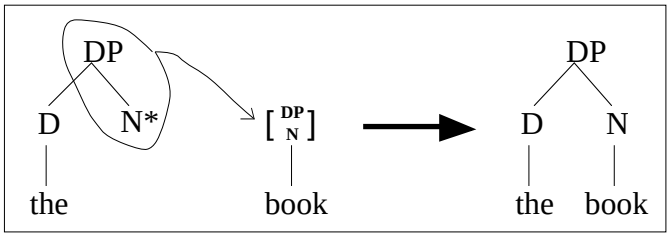

Fig. 3. Determiner's adjunction

Note that it does not change anything here if we use a NP label rather a DP label. In the following, determiners are no longer considered, and a N label will be used for noun phrases (as in Abeillé 1991).

Predicative adjective. Basic adjectives are considered as unary predicates, which adjoin on their semantic argument when they are attributive. Conversely, when they are predicative, their semantic argument substitutes. So in *Peter seems happy*, *Peter*, which is a semantic argument of *happy* and not of *seems*, will substitute in *happy* and *seems* will adjoin in *happy*. The tree α*happy* will thus contain a [VP|A] node on which β*seem* will adjoin. Note that such a category forces the adjunction of a verb. The verb *be* will be treated, in this case, as *seem*, although it is semantically empty.[3]

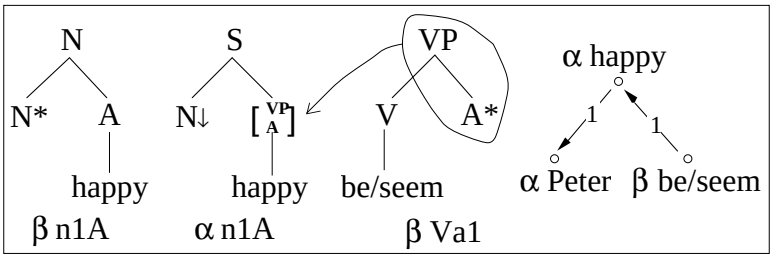

Fig. 4. Derivation tree for *Peter is/seems happy*

Tough-movement. Tough-movement is described in the same way as predicative adjective and the same trees are used for the copulative verb *be* and the raising verb *seem* (Fig. 5 and 6).[4] The

[3] The verbs *be* and *seem* differ not only semantically but syntactically: *Is Peter happy?* / *Does Peter seem happy?* Even if they share the tree of Fig. 4, they do not share the same family of trees.

[4] We have represented the complement of *easy* as a small clause labeled S. Phrase such *as easy for Mary to read* are described in the same way. The treatment of unbounded tough-movement *(This book is easy for me to believe that John would ever read*, adapted from Bresnan 1982: 255) can also be analyzed; it requires a tree β*for…to believe that* which will adjoin on a special tree α*read* (similar to the tree of Fig. 5, but with a finite S) and on which the tree β*easy* of Fig. 5 will adjoin. To avoid overgeneration, the tree *easy* must specify explicitly that its foot node is a S[(*for*)…*to*]

derivation tree can again be interpreted as a correct semantic graph. Note that *easy* needs different trees in the two constructions considered, which is avoided in GAG/DTG (Candito & Kahane 1998).

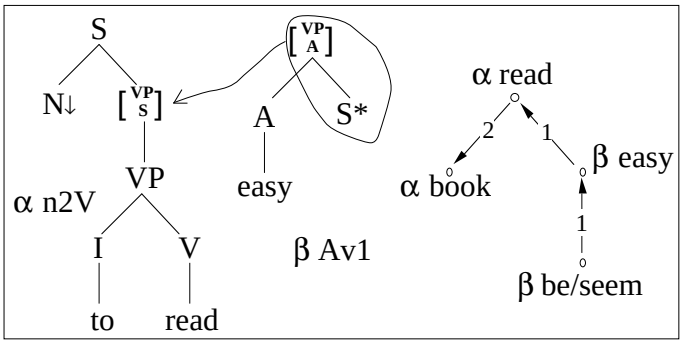

Fig. 5. The derivation of *the book is easy to read*

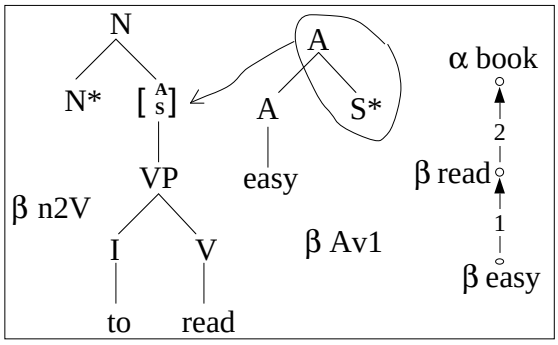

Fig. 6. The deriv. of *a book easy to read*

3. Extractions

We will consider a case of pied-piping in French:

(2) *Marie connaît la fille à la mère de qui Pierre parle.*
M. knows the girl to the mother of which P. talks.

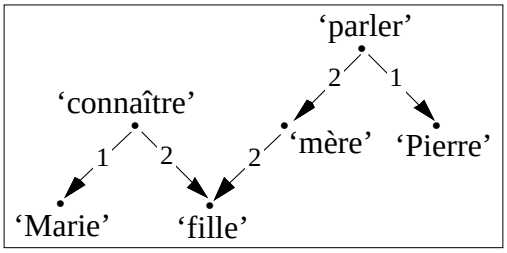

Fig. 7. The semantic graph of (2)

Three solutions will be considered. In the first one (Fig. 8), the verb *parle* ‘talk’ and the wh-word *qui* ‘which’ co-anchor a tree β*à qui-parle*, which adjoins on the antecedent *fille* ‘daughter’. To obtain (2), β*mère* must adjoin on β*à qui-parle*. In this case, the derivation tree cannot be satisfactorily interpreted as a semantic graph, because *parle* ‘talk’ is not the semantic argument of *mère* ‘mother’. Nevertheless, this is a good solution from a weak generative capacity viewpoint.

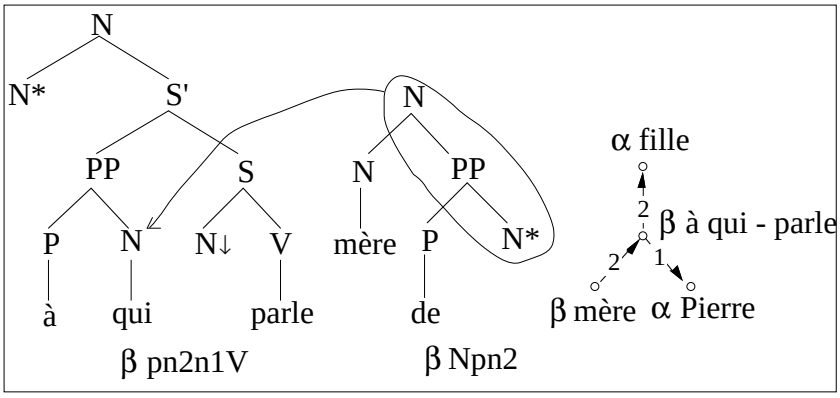

Fig. 8. A first (non suitable) derivation for (2)

The second solution (Fig. 9) is adapted from Kroch 1987 and is adopted by all the studies we know in TAG. The tree β*à qui-parle* of the first solution is broken in two trees: a tree β*parle*, which still adjoins on the antecedent, and a tree α*qui*, which substitutes in it.

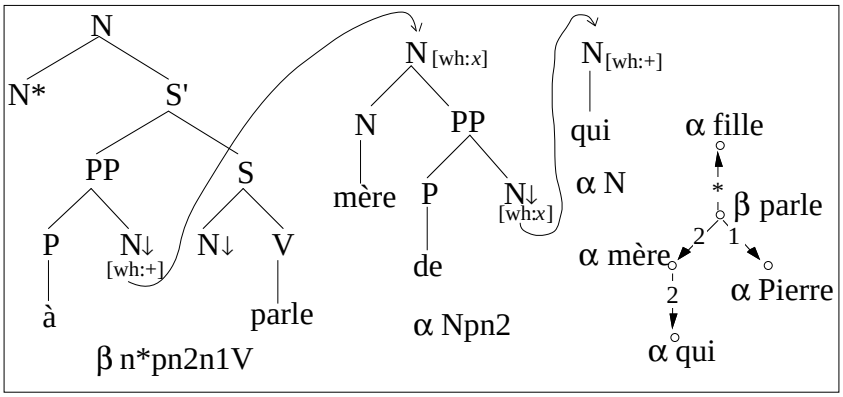

Fig. 9. A second possible derivation for (2)

In this solution, *mère* is the semantic argument of *parle*, but there is also an adjunction arc between β*parle* and the antecedent that cannot be interpreted as a semantic dependency. Moreover, a feature [wh] is necessary to ensure that the noun phrase that substitutes in the extracted position of β*parle* contains a wh-word. So a wh-word must be [wh:+] and a tree such as α*mère* must have two coreferent features [wh:*x*]. To avoid that a noun phrase without a wh-word substitute on a [wh:+] position, a noun must be [wh:-].

The idea of the third solution (Fig. 10) is to break the tree β*à qui-parle* of the first solution in another way. Following Tesnière 1959, we consider that the wh-word plays two roles: on one hand, it fills a position in the relative as pronoun and on the other hand it controls the distribution of the relative. If we follow this idea, it is more natural to attach the power to adjoin on a noun to the wh-word than to the verb of the relative. The adjoining arc between β*qui* and the antecedent (labeled =) can be interpreted as a link of coreference which can be collapsed to keep only the semantic dependencies.

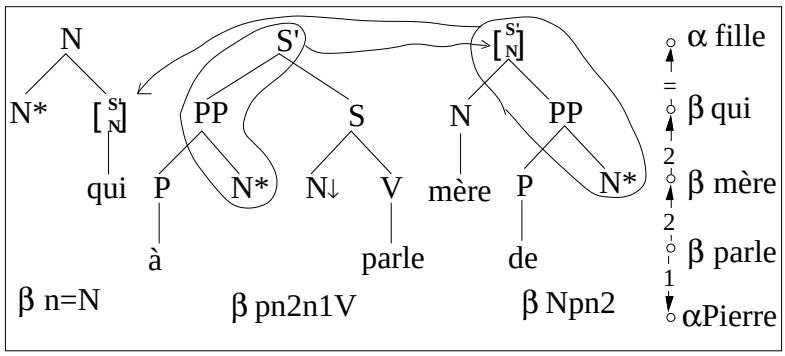

Fig. 10. A third (more suitable) derivation for (2)

As we see, β*parle*, which have a top node of top category S' and a foot node of bottom category N, can adjoin on the node of category [S'|N] of β*qui*. In addition to the fact that this analysis gives us the right semantic dependencies, there is another advantage: the same trees β*parle* and β*mère* can be used for other extractions, such as topicalization and direct or indirect interrogatives:

(3) **a.** *A la mère de Marie, Pierre parle.*
To the mother of Mary, Peter talks.
b. *Marie sait à la mère de qui Pierre parle.*
M. knows to the mother of which P. talks.

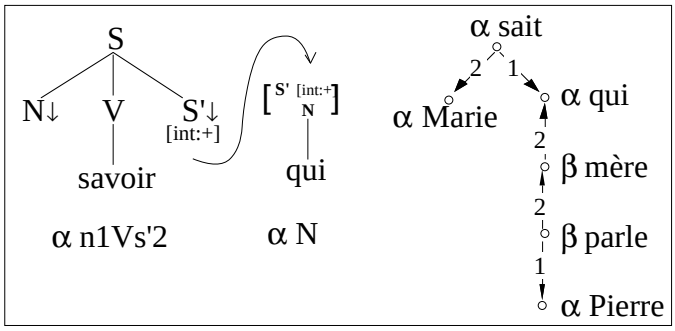

Fig. 11. Derivation of (3b)

This solution makes it possible to handle constructions that cannot be described in the Kroch 1987 analysis, without using multi-component TAG. That is the case of French *dont*-relative where a noun complement of a subject or a direct object is extracted:

(4) *le livre dont Pierre aime la fin*
the book of-which Peter likes the end
'The book whose end Peter likes'

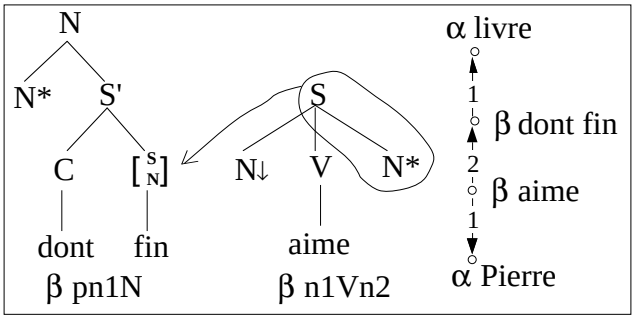

Fig. 12. Derivation of (4)

English sentences with extraction out of a noun complement can be analyzed in the same way:

(5) **a.** *the girl who Peter painted (a copy of) a picture of*
b. *Peter painted (a copy) of a picture of this girl*

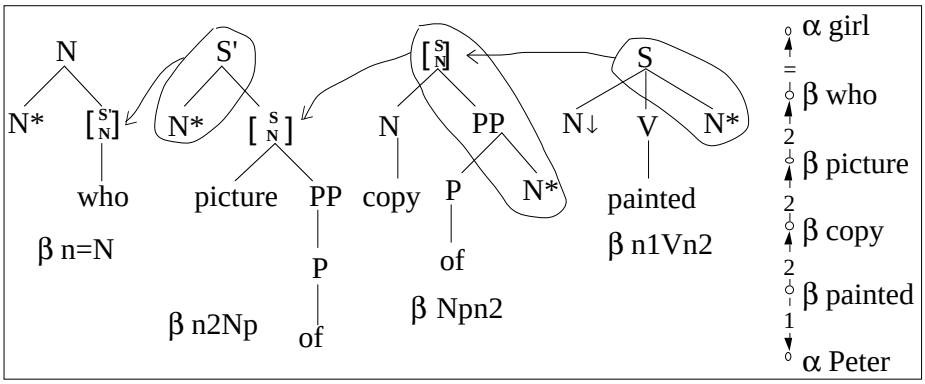

Fig. 13. Derivation of (5a)

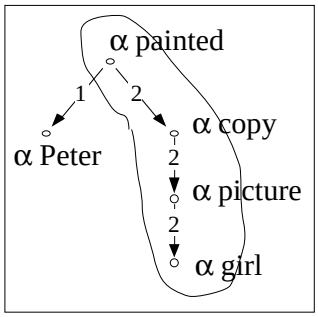

Fig. 14. Derivation of (5b)

We will now give an analysis of a well known and puzzling construction in French (Kayne 1975). As it can be seen in (6), the extraction of a subject phrase out of subordinate clause is possible, but only with a strange alternation of the complementizers:

(6) **a.** *le type qui dort*

the guy who is-sleeping

b. *Je pense que ce type dort*
I think that this guy is-sleeping

c. *le type que je pense qui dort*
the guy that I think [that] is-sleeping

d. ** le type qui je pense que dort*

Our analysis is based on the following assumptions:

1) *que* and *qui* are two forms of a same lexeme *qu-* : *qui* = *qu-*$_{[nom:+]}$ and *que* = *qu-*$_{[nom:-]}$.
2) A phrase of category S' must contain one and only one term in the nominative case: it is either the subject of the verb or, if the subject is extracted, the complementizer. For this reason, the two subconstituents of an S' must bear [nom] features with opposite values.

In other words, our analysis supposes that a subject can be extracted, but not the nominative case borne by it. In conformity with our assumption that a complementizer is attached to the semantic governor of the link that it marks, the wh-word *qu-* introducing the relative clause co-anchors the tree of a verb whose subject has been extracted (tree βn1qu-V, Fig. 14), which is the semantic governor of the antecedent noun. If no bridge verb is inserted, as in sentence (6a), *qu-* becomes [nom:+] and is realized by *qui*, else it becomes [nom:-] and is realized by *que*, as in sentence (6c). Conversely, the complementizer *qu-* that introduces the subordinate clause subcategorized by the bridge verb *pense* 'think' co-anchors the tree β*pense*. If the bridge verb adjoins on a verb with a subject, as in (6b), *qu-* becomes [nom:-] and is realized by *que*, while it becomes [nom:-] and is realized by *qui* if it adjoins on a verb whose subject has been extracted, as in (6c). Our solution differs from Franck 1992:173, where the complementizers are not attached to the semantic governors and it is not possible to use the same elementary trees to derive the sentences (6a-c).

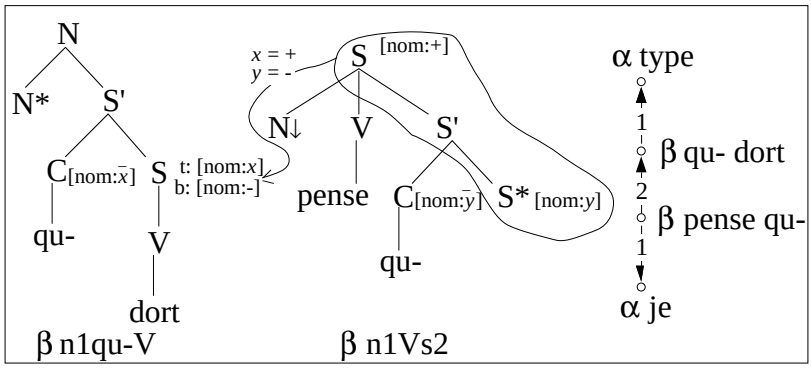

Fig. 14. Derivation of (6a) and (6c)

4. Conclusion

The main attraction of Kroch's analysis is its ability to derive a variety of constraints on extraction. Our analysis retains this particularity and even extends it to pied-piping cases. Extractions are a case of mismatch between syntactic and semantic dependencies: the syntactic head of a relative clause—the main verb of the clause—, which syntactically depends on the antecedent, is generally not semantically linked to the antecedent (e.g. *parle* in (2), *aime* in (4) or *pense* in (6c)). As proposed in Kahane & Mel'çuk 1999, the constraints on extraction can be expressed on the string of syntactic dependencies between the syntactic head of the clause following the extracted element and the gap. One particularity of the TAG description concerns this string: in case of extraction, the hierarchy induced by the derivation tree on this string is the converse of the hierarchy in the syntactic dependency tree, which is also the hierarchy generally adopted for a derivation without extraction (compare Fig 13 and 14). For this reason, all the string of nodes between the syntactic head and the gap is realized by predicative trees. Moreover, these trees have the following characteristics: the nodes that have been piped and are in COMP (*mère* in (2), Fig 10 will receive a

predicative tree rooted by S' without S node, while the node which is linked to COMP (e.g. *parle* in (2)/Fig. 10; *picture* in (5a)/Fig. 13) will receive a predicative tree rooted by S' with a S node. The nodes that are between the node linked to COMP and the syntactic head of the relative will receive a predicative tree rooted by S.[5] And the converse is true. In other words, a lexical unit can be in one of the three positions considered in the string between the syntactic head and the gap if it has a tree of one of three types proposed.

Although our analysis handles more extractions than Kroch 1987's analysis, some constructions still cannot be suitably described. For instance, problems arise when one of the dependencies between the syntactic head and the gap is a substitution arc: it is the case for extractions outside an interrogative clause (*le livre* ***que*** *je sais à qui* ***donner*** 'the book that I know to which to-give': αlivre <-2- βque donner **–3->** αqui <-2- βsais) or extractions where the wh-word is a modifier in the relative and might be both adjoined in the relative and on the antecedent (*the guy* ***whose*** *car I borrowed*: αguy <-1- βwhose **–2->** αcar <-1- βborrowed).[6] In both cases, the tree which substitutes (α*qui* or α*car*) is not in an adequate position for the tree that might adjoin on it. All these problems can be avoided in GAG/DTG where multiple adjoining and substitution of a same elementary tree are possible (Candito & Kahane, 1998). For instance, the wh-word *where* will receive an elementary structure which can adjoin simultaneously on the antecedent *bed* and on the verb *slept* it modifies. Similarly, the wh-word *qui* in (2) will receive an elementary structure that can simultaneously adjoin on its antecedent and substitute in the relative clause. But contrary to Kroch's analysis and our analysis, constraints on extraction are not directly assumed by the categorial features of nodes and special features must be added for not overgenerating.

References

ABEILLE A. (1991). *Une grammaire lexicalisée d'arbres adjoints pour le français*. Ph.D. Thesis. Univ. Paris 7.

CANDITO M.-H. (1999) *Organisation modulaire et paramétrable de grammaires électroniques lexicalisées*. Ph.D. Thesis. Univ. Paris 7.

CANDITO M.-H. & KAHANE S. (1998). "Defining DTG derivations to get semantic graphs". *TAG+4*, Philadelphia, 25-28.

FRANCK R. (1992). *Syntactic Locality and Tree Adjoining Grammar : Grammatical, Acquisition and Processing Perspectives.* Ph.D. Thesis. Univ. of Pennsylvania.

KAHANE S. & MEL'CUK I. (1999). "La synthèse sémantique ou la correspondance entre graphes sémantiques et arbres syntaxiques – Le cas des phrases à extraction", *T.A.L.*, **40**:2, 25-85.

KAYNE R. (1975). *French Syntax : The Transformational Cycle*, MIT Press.

KROCH A. (1987). "Subjacency in a Tree-Adjoining Grammar". In Manaster-Ramer A. (ed), *Mathematics of Language*, Benjamins, 143-71.

KROCH A. & JOSHI A. (1986). "Analysing Extrapositions in a TAG". In Huck G. & Ojeda A. (eds), *Discontinuous Constituents, Syntax and Semantic 20,* Academic Press, 107-49.

MEL'CUK I. (1988). *Dependency Syntax : Theory and Practice*. State Univ. of NY Press.

RAMBOW O., VIJAY-SHANKER K. & WEIR D. (1995). "D-tree Grammars". *ACL'95* .

TESNIERE L. (1959). *Eléments de syntaxe structurale*, Klincksieck, Paris.

VIJAY-SHANKER K. (1987). *A Study of TAG*. PhD Thesis, Univ. Pennsylvania, Philadelphia.

VIJAY-SHANKER K. (1992). "Using descriptions of trees in TAG". *Computational Linguistics*, **18**:4, 481-517.

XTAG Research Group (1995). "A Lexicalized TAG for English". Technical Report IRCS 95-03, Univ. of Pennsylvania. (On line updated version).

ZOLKOVSKI A. & MEL'CUK I. (1967). O semanticeskom sinteze [On semantic synthesis]. *Problemy kybernetiki*, **19**, 177-238.

[5] The only exception to this principle concerns the raising verbs, such as *seem*, which receive a tree rooted by VP; but as noted in Candito & Kahane 1998, it is not always possible to obtain the right dependencies with such trees.

[6] Note that the phrase *the bed* ***where*** *I slept* (αbed <-1- βwhere **–2->** αslept) can be analyzed, but with a rather strange tree for *where*, where the clause modified by *where* must substitute in the tree β*where*.