An Interaction Grammar of interrogative and relative clauses in French

Guy Perrier
LORIA - université Nancy 2
BP 239
54506 Vandœuvre-lès-Nancy cedex - France
guy.perrier@loria.fr

Abstract

We present a fairly complete grammar of interrogative and relative clauses in French, written in the formalism of Interaction Grammars. Interaction Grammars combine two key ideas: a grammar is viewed as a constraint system which is expressed through the notion of tree description, and the resource sensitivity of natural languages is used as a syntactic composition principle by means of a system of polarities.

Keywords

Syntax, grammatical formalism, tree description, polarity, interrogative clause , relative clause, interaction grammar

1 Introduction

This article is a contribution to the construction of formal grammars from linguistic knowledge. This task is motivated by both applicative and scientific considerations. From an applicative point of view, it is essential for NLP systems requiring a fine and complete syntactic analysis of natural languages. From a scientific point of view, formalization can be very helpful for linguists, who aim at capturing the complexity of a natural languages with relevant generalizations. In this task, one of the most difficult challenges is to get the largest possible coverage of these formal grammars.

Regarding this challenge, relative and interrogative clauses in French are a good test because they illustrate the complexity of natural languages in a very obvious manner. They give rise to interference between several phenomena, which are present in both types of clauses and justify a common study. In the following, a clause which is a relative clause or an interrogative clause is called a *wh-clause*. In this paper, we have highlighted four phenomena occurring with wh-clauses:

- *Wh-extraction*: a particular constituent is extracted from its canonical position in the wh-clause to be put in front of it: if the clause is interrogative, it contains the requested information and if the clause is relative, it contains a reference to the antecedent of the relative clause; in both cases, they are represented with grammatical words, which we call *wh-words*. The difficulty lies in the fact that extraction can occur through a chain of a possibly indeterminate number of embedded clauses. This gives rise to an unbounded dependency which is subject to special constraints, called *island constraints*.
- *Pied-piping*: in some cases, wh-words drag a complex phrase along with them in the extraction movement. Another unbounded dependency may be generated from the fact the wh-word can be embedded less or more deeply in the extracted phrase.
- *Subject inversion*: contrary to canonical constructions of clauses where the subject precedes the verb, wh-clauses allow subject inversion under some conditions. These conditions depend on various factors.
- *Interrogative and declarative marking*: in French, relative clauses and interrogative clauses often use the same wh-words but they differ in the fact that the first ones are marked declaratively, whereas the second ones are marked interrogatively. If we consider only written texts, there are four main ways of marking clauses interrogatively or declaratively: the punctuation, the position of subject clitics with respect to the verb, the construction of clauses as objects of verbs expecting questions, and special terms like *est-ce que*.

If we aim at capturing all these phenomena most fairly, we need a rich formalism but which is at the same time simple enough to keep the formal representations readable. We have chosen the formalism of Interaction Grammar (IG) [8], for two main reasons:

- The basic objects of the formalism are pieces of underspecified syntactic trees, which can combine very freely by superposition. Such a flexibility is used at the same time in the construction of modular grammars and then in the process of syntactic composition. In our application, underspecification will be used to represent unbounded dependencies related to wh-extraction and pied-piping.
- The resource sensitivity of natural languages is used as a principle of syntactic composition under the form of a system of polarities. In this way, syntactic composition consists in superposing pieces of syntactic trees under the control of polarities with the goal of saturating them. The

use of polarities for managing the interrogative and declarative marking of clauses is an elegant illustration of this principle.

In section 2, we give an informal presentation of IG with the help of an example. Then, we show how to build an IG of wh-clauses focusing on four phenomena: wh-extraction (section 3.1), pied-piping (section 3.2), subject inversion (section 3.3), interrogative and declarative marking (section 3.4). We end with an evaluation of the grammar.

2 Presentation of Interaction Grammars

2.1 Tree descriptions and polarities

The basic objects of IG [8] are *tree descriptions*, which can be viewed as partially specified syntactic trees. Their nodes represent syntactic constituents and they are labelled with feature structures representing the morpho-syntactic properties of constituents. The nodes are structured by two kinds of relations: dominance and precedence. Both can be underspecified.

Tree descriptions combine by superposition under the constraints of polarities expressing their saturation state. Polarities are attached to features, so that features are triples (name, polarity, value), which are called *polarized features*. Tree descriptions labelled with polarized feature structures are called *polarized tree descriptions (PTD)*.

The superposition of two PTDs is realized by merging some of their nodes. When two nodes merge, their feature structures are composed according to an operation, which reduces to classical unification if we forget polarities.

There are 5 polarities: neutral (=), positive (→), negative (←), virtual (∼), saturated (↔). Polarities are composed according to an operation, denoted ⊕, defined by the following table .

⊕	=	→	←	∼	↔
=	=				
→			↔	→	
←		↔		←	
∼		→	←	∼	↔
↔				↔	

In this table, an empty entry means that the corresponding polarities fail to be composed. The neutral polarity applies to features that behave as non consumable resources, agreement features for instance. Other polarities interact together with the aim of being saturated. Thus, according to the table, there are two kinds of interactions:

- *Linear interactions*: a linear interaction occurs between exactly one positive feature $f \rightarrow v_1$ and one negative feature $f \leftarrow v_2$ to combine in a saturated feature $f \leftrightarrow v_1 \wedge v_2$[1]; in this way, both features become saturated. Then, they can only combine with virtual features; they cannot combine any more with another positive, negative or saturated feature. This mainly expresses interaction between predicates and arguments, in which one predicate requires exactly one argument for each function.
- *Non linear interactions*: a non linear interaction occurs between one saturated feature $f \leftrightarrow v$ and n (n being possibly equal to 0) virtual features $f \sim v_1, \ldots, f \sim v_n$ to saturate these virtual features into a feature $f \leftrightarrow v \wedge v_1 \cdots \wedge v_n$. This interaction can be viewed as an absorption of any number of virtual polarities by a saturated polarity. It mainly models two types of interactions: context requirements and applications of modifiers to constituents.[2]

The system of polarities presented here is not the only possible one. It is important to understand that the polarity system is a parameter of any IG. For instance, the system used in the initial presentation of IG [8] differs from the present system on one point: neutral features have the property of absorbing virtual features.

2.2 Syntactic composition and parsing

In IG, a *syntactic composition process* is defined as a sequence of PTD superpositions controlled by interactions between polarities. One superposition is composed of elementary operations of *node merging.* When two nodes merge, their feature structures are composed, which can give rise to some interactions between their polarized features.

A particular IG is defined by a finite set of PTDs, the *elementary PTDs (EPTDs)* of the grammar. In practice, the actual IGs are totally lexicalized: each EPTD has a special node that is linked to a word of the language; this node is unique and it is called the *anchor* of the EPTD.

All *valid syntactic trees* generated from the grammar are the *saturated trees* resulting from a syntactic composition process of a finite set of EPTDs. A saturated tree is a PTD that is a completely specified tree in which all polarities are neutral or saturated The language generated by the grammar is the set of the *yields* of the valid syntactic trees, that is the sequences of words attached to the leaves of the trees.

To parse a sentence with a particular IG, we first have to select an EPTD for each word of the sentence: the anchor of the PTD must be linked with the corresponding word. Then, we have to perform the syntactic composition of the selected EPTDs to find a valid syntactic tree, the yield of which is the parsed sentence.

The parsing problem for IG in its whole generality is NP-hard, which can be shown with an encoding of

[1] Feature values v_1 and v_2 are disjunctions of atoms and $v_1 \wedge v_2$ represents their conjunction.

[2] If we look at the polarity composition table carefully, we remark that a virtual feature can also be absorbed by a positive or a negative feature but these features must then combine with a dual feature to become saturated. Apart from the order, this leads to the same result as a linear interaction followed by a non linear interaction. From this consideration, it is logical to extend the notion of non linear interaction to any interaction between a virtual feature and a positive or negative feature.

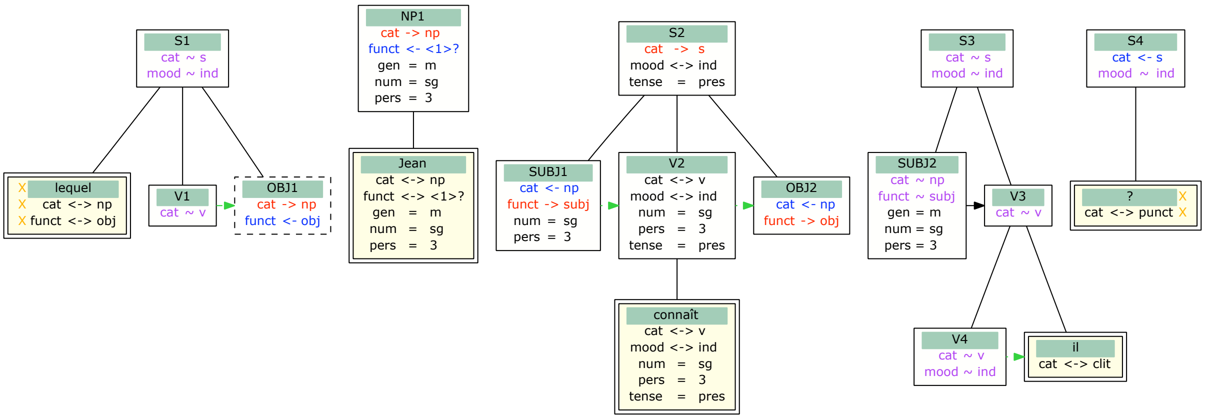

Fig. 1: *EPTDs selected from an IG for sentence 1*

the Intuitionistic Implicative fragment of linear logic. Nevertheless, grammars aiming at modelling real natural languages allow the use of original methods for parsing, which are based on polarities and make parsing tractable. This is not the concern of this article and the reader can refer to [3, 4, 2] for an in-depth study of parsing with IG.

2.3 An example

Consider the following sentence to parse with an IG:

(1)	*Lequel*	*Jean*	*connaît-il*	*?*
	which one	Jean	does he know	?

‘Which one does Jean know ?’

Figure 1 represents a possible selection from the grammar for the words of sentence 1. The EPTD associated with *connaît* represents a syntactic construction where the transitive verb is used in the active voice. Node S_2 represents the clause with *connaît* as its head. It is composed of three constituents: the verbal kernel, the subject and the object of the verb, which are respectively represented by nodes V_2, $SUBJ_1$ and OBJ_2. The basic constituent of the verbal kernel is the bare verb, the anchor of the EPTD, represented with a double frame. In $SUBJ_1$, the negative feature $cat \leftarrow np$ and the positive feature $funct \rightarrow subj$ mean that *connaît* expects a noun phrase to provide it with the subject function. Underspecified strict precedence between the three nodes is represented with dashed arrows.

The EPTD associated with *il* represents its construction as a subject clitic pronoun, put just after the verb as a duplication of the actual subject to make the clause interrogative. Node V_4 represents the bare verb, which is concatenated with *il* to constitute the cliticized verb V_3. The *cat* features of nodes V_3 and V_4 are virtual to express that *il* acts as a modifier of an actual verb: nodes V_3 and V_4 have to merge with actual verbs. Nodes S_3 and $SUBJ_2$ represent a required context: a finite clause with a third person singular male subject. This is expressed with virtual features.

The EPTD associated with *lequel* represents its construction as an object interrogative pronoun. The interrogative clause is represented by the node S_1. The pronoun *lequel* is put in front of the clause to represent an extracted object[3] and the canonical position of the object is occupied by a trace, the node $OBJ1$ which is represented with a dashed box to express that the trace has an empty phonological form.

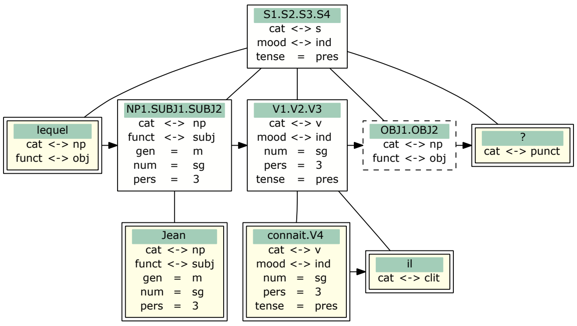

Fig. 2: *Parse tree of sentence 1*

The EPTD associated with *Jean* contains two *funct* features with an undetermined value, which is expressed with a question mark. It means that the noun phrase can receive any syntactic function in the sentence. The index $<1>$ that is put before the value indicates that the *funct* features of nodes *NP1* and *Jean* share the same value.

Punctuation signs are considered as ordinary words and they are also associated with EPTDs. So, the question mark is associated with an EPTD, the root of which represents the interrogative sentence.

The parsing succeeds because the syntactic composition of the EPTDs from figure 1 ends with the valid syntactic tree given by figure 2. On the figure, the head of the box representing each node contains the names of the nodes from the initial EPTDs that were merged into this node. The parsing of the sentence is composed of 9 mergings, including themselves 5 linear interactions and 12 non linear interactions. Among the 12 non linear interactions, only one realizes the action of a modifier; the others realize context requirements.

[3] The crosses on the left of the box representing *lequel* mark that the node is the leftmost daughter of node *S1*.

3 Modelling interrogative and relative clauses in French

3.1 Wh-extraction

Wh-extraction is a common property to relative and interrogative clauses in French: wh-words appear at the beginning of the clause and they play the role of a constituent that is lacking in the clause. The empty position in the clause is usually marked with a trace.

(2)

Où	*Pierre*	*pense-t-il*	*que*	*Marie*
where	Pierre	does he believe	that	Marie
veut	*aller*	□	*?*	
wants	to go		?	

'Where does Pierre believe that Marie wants to go ?'

In sentence 2, the trace is indicated with a □ symbol. The wh-word is bold. Contrary to sentence 1, in sentence 2, the trace is located in a clause which is not the interrogative clause introduced by the wh-word but an embedded object clause. The clause *aller* □ is an infinitive clause, which is included in the finite clause *que Marie veut aller* □ , which is itself included in the interrogative clause *où Pierre pense-t-il que Marie veut aller* □. The number of embedded clauses between the trace and the wh-word is undetermined, hence an unbounded dependency between the wh-word and the verb that has the trace as its complement.

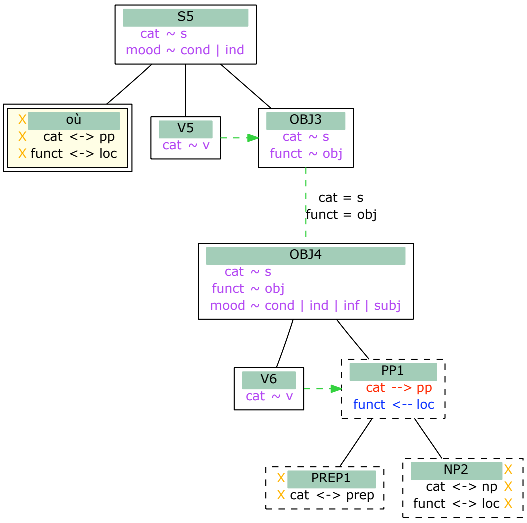

Fig. 3: *EPTD for the interrogative pronoun* où

IG uses an underspecified dominance relation to model this unbounded dependency. Figure 3 shows the EPTD used in sentence 2 to represent the interrogative pronoun *où*. Node *OBJ3* represents the object clause that is immediately included in the main clause, *que Marie veut aller* □ in our example. Node *OBJ4* represents the last embedded object clause, which has the trace as an immediate constituent, the infinitive clause *aller* □ in our example. There is an underspecified dominance relation from *OBJ3* to *OBJ4* , in order to express that there is an undetermined number of object clauses inserted between them. Dominance relations must be understood in a large sense: *OBJ3* may merge with *OBJ4*.

The underspecified dominance relation is represented in figure 3 with a dashed vertical line. The line is labelled with two neutral features $cat = s$ and $funct = obj$, which mean that all nodes dominated by *OBJ3* and dominating *OBJ4* in a large sense, must be equipped with both features. The features labelling dominance relations interact with the feature structures of the concerned nodes by unification. This mechanism models the fact that all constituents included in the main interrogative clause and containing the trace are object clauses, which is a way of implementing island constraints. Lexical Functional Grammar (LFG) [5] uses a similar form of constraint on underspecified dominance: *functional uncertainty.*

In figure 3, the trace is represented by the subtree rooted at node *PP1*. This node is labelled with a positive feature $cat \rightarrow pp$ and a negative feature $funct \leftarrow loc$. It means that it provides a prepositional phrase which expects to receive a locative function. Both polarized features will be saturated by the verb *aller.*

3.2 Pied-piping

Pied-piping represents the ability of wh-words to drag complex phrases along with them when brought to the front of interrogative or relative clauses. Here is an example of pied-piping in which the extracted phrase is put between square brackets.

(3)

[Dans	*la*	*maison*	*du*	*père*	*de*	***qui]***
in	the	house	of	father	of	whom
Pierre	*pense-t-il*	*que*	*Marie*	*veut*	*aller*	
Pierre	does he believe	that	Marie	wants	to go	
□	*?*					
	?					

'in the house of whom father does Pierre believe that Marie wants to go ?'

In sentence 3, the wh-word is embedded more or less deeply in the extracted phrase, via a chain of noun complements introduced by the preposition *de*. Besides extraction, such a construction is a second source of unbounded dependency and IG also uses an underspecified dominance relation to model it.

Figure 4 gives an example of EPTD associated with the interrogative pronoun *qui* used with pied-piping. The only change with respect to figure 3 lies in the subtree rooted at node *PP2* representing the complex extracted phrase. In sentence 3, *PP2* represents *dans la maison du père de qui.* Anchor *qui* is embedded in a chain of noun complements introduced by the preposition *de*. Nodes *PP4* and *PP5* represent the extremities of the chain and the underspecified dominance relation between them means that the number of elements of the chain is undetermined. In particular, it can be null when *PP5* is identified with *PP4*. The features labelling the dominance relation express a constraint on the nodes constituting the chain. These nodes must be only nouns, noun phrases or prepositional phrases with either no syntactic function or complements introduced by the preposition *de*.

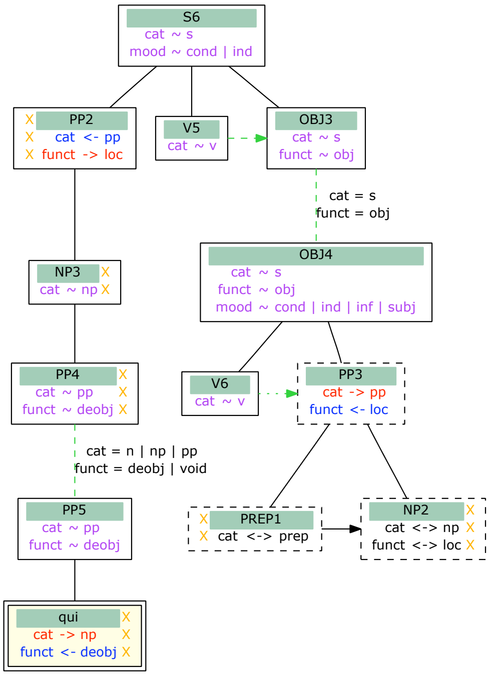

Fig. 4: *EPTD for the interrogative pronoun* qui

3.3 Subject inversion

Subject inversion is allowed in interrogative and relative clauses under some conditions. For the relative clauses, inversion is usually possible, but there is a case in which it is strictly forbidden: when there is a possible confusion between the inverted subject and an object of the verb. For the interrogative clauses, besides the presence of an object for the verb, some interrogative words prevent subject inversion while others force it. In the first class, there is *pourquoi* (*why*) and in the second case, there is *que* (*what*). Others like *où* (*where*) are unconcerned with the subject-verb order.

(4) *Que* *fait* *Marie* *?*
what is doing Marie ?
'What is Marie doing?'

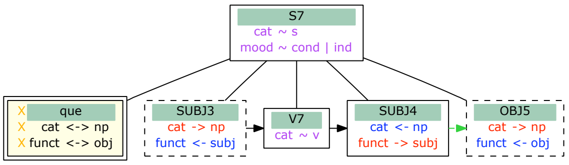

Fig. 5: *EPTD for the interrogative pronoun* que

Figure 5 shows how subject inversion in sentence 4 is modeled in an EPTD associated with the interrogative pronoun *que*. Node *SUBJ3* represents the trace of the subject at the initial position just before the verbal kernel represented by node *V7*. Node *SUBJ4* represents an expected noun phrase just after the verbal kernel, to which the EPTD gives the *subject* function. The interest of this way of representing subject inversion is that the same EPTD for a transitive verb is used in the canonical construction and the inverted construction of the subject-verb order. The doubling of the number of EPTDs for transitive verbs in the grammar is avoided this way.

3.4 Declarative and interrogative marking

Nodes representing clauses are equipped with a *typ* feature, which can take three values, *decl*, *inter* or *inj*, according to the type of the clause: declarative, interrogative or injunctive. The feature is polarized and since grammars are lexicalized, there is a subtle interaction between words to saturate them.

The interaction is especially complex for interrogative clauses. In direct interrogations, sentence 2 for instance, the question mark brings a negative feature *typ* ← *inter* and in indirect interrogations, the same feature is brought by the main verb which expects a question.

In direct interrogations, like sentence 2, if there is a subject clitic put just after the main verb, this clitic brings the positive feature *typ* → *inter*. Otherwise, the positive feature is brought by the wh-word.

The declarative marking of relative clauses is simpler: the relative pronoun brings the feature *typ* ↔ *decl* to the relative clause.

This way of marking interrogative and declarative clauses is not the only possible with IG. Its system of polarities is rich and flexible enough to provide other solutions.

4 Organization of the grammar

In previous sections, we focused on four main aspects of the grammar of the interrogative and relative clauses in French but the whole grammar is more complex and its implementation is a real challenge.

For this, we used the XMG tool [7]. XMG provides a high level language dedicated to grammar description. A grammar is designed as a set of classes, each class defining a set of PTDs. The initial classes are composed into complex classes by means of two operations: *conjunction* and *disjunction*. Classes are ranked by family. Figure 6 shows the architecture of our IG of interrogative and relative clauses in French. We call it IG_{Wh}. Ovals represent families and an arrow from an oval to another one means that some classes from the first family are used in the definition of classes of

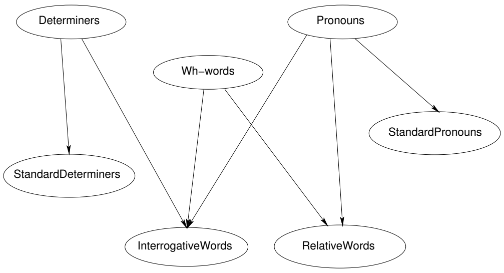

Fig. 6: *Organization of the grammar*

Some classes are distinguished as *terminal classes* and they are compiled by XMG into a set of EPTD constituting an IG usable for automated parsing. IG_{Wh} is composed of 40 classes. Among them, 16 terminal classes are distributed in 4 relative classes and 12 interrogative classes. These terminal classes are compiled into 77 EPTDs anchored with relative pronouns and 295 EPTDs anchored with interrogative pronouns and determiners.

5 Evaluation and comparison with other works

The evaluation of IG_{Wh} has two aspects:

- its precision measures the extent to which all parse trees generated from IG_{Wh} reflect valid constructions in French;
- its recall measures the extent to which all constructions with interrogative or relative clauses in French are captured by IG_{Wh}.

The use of treebanks on large real corpora with a classical F-measure is not well suited to such an evaluation: even if they are very large, their grammatical coverage of relative and interrogative clauses is too limited and they only include positive sentences, when ungrammatical sentences are necessary to spot overgeneration. Finally, the construction of such treebanks is very costly and for French, they are too limited both in number and size.

The least costly way of evaluating IG_{Wh} is to use it for parsing a test suite of grammatical and ungrammatical sentences illustrating most rules of the French grammar related to interrogative and relative clauses. Unfortunately, there exists no test suite satisfying this specification. The TSNLP [9] contains no relative clause and only direct interrogative sentences covering the grammar of these sentences very partially. The EUROTRA test suite [6] contains relative clauses but its coverage is limited and it does not contain any ungrammatical sentence.

We have built our own test suite, consisting in a hundred grammatical sentences and a hundred ungrammatical sentences. These sentences cover most syntactic phenomena related to relative and interrogative clauses. Of course, these sentences need a complete grammar to be parsed but, as we focus on relative and interrogative clauses, we take care to choose only simple rules with regard to other phenomena. For the parsing, we used LEOPAR[4], which is a parser devoted to IG. The hundred grammatical sentences were parsed successfully, all parse trees were verified manually and the hundred ungrammatical sentences were rejected by the parser.

Another way of evaluating our grammar is to compare it with other existing grammars of interrogative and relative clauses in French. Unfortunately, there are very few works about this question. Anne Abeillé [1] has developed a grammar of French in the formalism of Tree Adjoining Grammar (TAG). This grammar covers relative and interrogative clauses. Like our grammar, it is lexicalized, but the main difference is that all syntactic constructions related to wh-clauses are attached to the head verb of these clauses. From a computational point of view, this is an important drawback because it inflates the number of elementary trees associated with verbs, which makes the selection of relevant trees more difficult. Even if it fits on with the locality principle[5], such a feature is determined by the rigidity of the operation of syntactic composition, adjunction, which does not allow for flexibility about the way of attaching syntactic information to words. Nevertheless, the attachment of the syntactic constructions related to wh-clauses to verbs presents one advantage over the attachment to wh-words: the absence of a direct objet just after the head verb of the wh-clause can be made explicit in the elementary tree attached to this verb, so that subject inversion is completely controlled. Wh-extraction, interrogative and declarative marking are expressed with the TAG machinery: the mechanism of adjunction combined with an important system of control features. Another illustration of the limited expressivity of adjunction is the inability to represent the extraction of noun complements. Finally, a point remains unclear: the modelling of unbounded dependencies generated by pied-piping.

Other formalisms, such as HPSG [10] and LFG [5] propose solutions for relative and interrogative clauses and other languages than French but in this article, we have stressed the combination of four aspects, which is very specific to French, and an exhaustive comparison would be lengthy for this article.

References

[1] A. Abeillé. *Une grammaire électronique du français.* CNRS Editions, Paris, 2002.

[2] G. Bonfante, B. Guillaume, and M. Morey. Dependency constraints for lexical disambiguation. In *11th International Conference on Parsing Technology, IWPT'09*, Paris, France, 2009.

[3] G. Bonfante, B. Guillaume, and G. Perrier. Analyse syntaxique électrostatique. *Traitement Automatique des Langues (TAL)*, 44(3):93–120, 2003.

[4] G. Bonfante, B. Guillaume, and G. Perrier. Polarization and abstraction of grammatical formalisms as methods for lexical disambiguation. In *CoLing'2004*, pages 303–309, Genève, Switzerland, 2004.

[5] J. Bresnan. *Lexical-Functional Syntax.* Blackwell Publishers, Oxford, 2001.

[6] B. Daille, L. Danlos, and O. Laurens. L'analyse du français dans le projet EUROTRA. *Traitement Automatique des Informations*, 32(2):89–106, 1992.

[7] D. Duchier, J. Le Roux, and Y. Parmentier. XMG : Un compilateur de méta-grammaires extensible. In *TALN 2005, Dourdan, France*, 2005.

[8] B. Guillaume and G. Perrier. Interaction Grammars. *Research on Language and Computation*, 2009. To appear. A preliminary report is available at http://www.loria.fr/ ∼ perrier/langcomp2009.pdf.

[9] S. Lehmann, S. Oepen, S. Regnier-Prost, K. Netter, V. Lux, J. Klein, K. Falkedal, F. Fouvry, D. Estival, E. Dauphin, H. Compagnion, J. Baur, L. Balkan, and D. Arnold. TSNLP — Test Suites for Natural Language Processing. In *Proceedings of COLING 1996,Kopenhagen*, 1996.

[10] I. A. Sag, T. Wasow, and E. M. Bender. *Syntactic Theory: a Formal Introduction.* Center for the Study of Language and INF, 2003.

[4] http://www.loria.fr/equipes/calligramme/leopar

[5] A predicate and its arguments must be present in the same elementary tree.