English Relative Clause Extraction: A Syntactic and Semantic Approach[1]

Thomas C. Bourgeois
University of Arizona

0.0 Introduction

Within this paper we analyze the formation of and extraction from a specific type of noun phrase, namely that consisting of the definite article followed by a common noun modified by a relative clause, where the common noun can be the subject or the object of the modifying clause. Representative examples of this construction appear in Figure 1:

(1) (i). **Sal knows the man Sid likes.**
(ii). **Sal knows the man who bought the carrot.**

The framework we assume here makes use of a system of functional syntactical and (corresponding) semantical types assigned to each item in the string. These types act upon each other in functor-argument fashion according to a small set of combinatory rules for building syntactic and semantic structure, adopted here without proof but not without comment. To emphasize the direct correspondence of the syntax/semantics relationship, we describe combinatory rules in terms of how they apply on both levels. For maximum clarity, data appear in the form of triplets consisting of the phonological unit (the word), the syntactic category, and the semantic representation. We present an example below:

(2)

bought
(NP\S)/NP
$\lambda o \lambda s.\mathbf{B}(o),(s)$

0.1 Functional Application

The most elementary combinatory rule is that of Functional Application, and at the syntactic level involves simple right-handed (RFA) or left-handed (LFA) concatenation of a function and its argument. Let **X/Y** define a functor category which, when an argument of category **Y** appears adjacent to it on its right, may be combined via RFA to form the resulting category **X**. Symmetrically, the functor **Y\X** immediately preceded by an argument **Y** to its left undergoes LFA to form the resulting category **X**. Finally, let **X|Y** represent a functor which, when either immediately preceded or followed by an argument **Y**, can undergo FA to form the category **X**.

Let the semantic representation corresponding to the functor set **[X/Y, Y\X, X|Y]** be $\lambda a.\mathbf{B}(a)$, where the λ-expression to the left is said to bind the variable within the parentheses, in this case an argument to the predicate **B**. Before FA applies syntactically, the semantic relation between functor and argument looks like this: $\lambda a.\mathbf{B}(a)\mathbf{C}$; the analog of FA at this level is a process called lambda-conversion, where the λ-operator to the left, $\underline{\lambda a}$, disappears and all occurrences of the variable bound by it are replaced with the argument to the right, in this case $\underline{\mathbf{C}}$. Symbolically, this process translates into:

$$\lambda a.\mathbf{B}(a)\mathbf{C} \dashrightarrow \mathbf{B}(\mathbf{C}).$$

[1] This paper would not exist without the assistance and conversation of R. T. Oehrle, D. Saddy, H. Muadz, and the rest of the linguists at the University of Arizona.

0.2 First-Order Functional Composition

With the syntactic and semantic notation defined above, we continue on to another combinatory process, First-Order Functional Composition (FC). Given a syntactic sequence of the categories **X/Y Y/Z Z**, RFA could apply Successively in the following manner:

```
X/Y   Y/Z   Z
      -------      RFA
X/Y   Y
---------          RFA
X
```

Another way to obtain the final category **X** is to combine the leftmost two terms **X/Y Y/Z** to make a composite functor **X/Z** via FC. Following that operation the terms will be **X/Z Z**, which can undergo RFA to produce the final result:

```
X/Y   Y/Z   Z
----------         FC
X/Z         Z
-------------      RFA
X
```

FC is also a semantic process, which we call lambda-substitution, whereby the λ-operator of the leftmost functor is replaced by the λ-operator(s) of the functor with which it composes, and the composed functor fills the argument position of the leftmost functor. In other words,

λa.**B**(a)(λd.**C**(d)) --> λd.**B**(**C**(d)).

0.3 Currying

A functor of type **(Z\X)/Y** can combine with an argument **Y** on its right and afterward with an argument **Z** on its left to yield the category **X**. Applying Currying (CUR) to that functor reverses the order of application of the functor and its arguments; this has the effect of switching the parentheses, that is, producing the functor **Z\(X/Y)** as its output. It is worth noting that the required configuration of the functor is **Z\X/Y** and not, say, **X/Y/Z**. In other words, applying CUR to **(X/Y)/Z** to give **X/(Y/Z)** is an invalid application of the process.

In the language of the lambda-calculus, CUR has the effect of switching the order of application of the λ-operators, and switching the order of the arguments themselves. For example, the semantic functor-argument expression [λoλs.**P**(o).(s)] **(A) (B)** would be represented as
[λsλo.**P**(o).(s)] **(B) (A)**, as a result of applying CUR.

0.4 Type Lifting

The final process discussed is that of Type Lifting (TL), which reverses the argument-functor relationship. Given a type sequence like **X/Y Y**, which can undergo RFA resulting in the category **X**, the category **Y** of the argument may become the resultant category **(X/Y)\X** via the TL process. Thus

X/Y Y

may be rewritten as

X/Y (X/Y)\X

creating the environment for LFA to take place, giving the resulting category

$$x.$$

Given the formula $\lambda b.A(b)\ B$, the analog to TL amounts to representing B as $\lambda P.P(B)$, where **P** is a variable which ranges over predicates (functor categories) instead of simple arguments.

$$\lambda b.\mathbf{A}(b)\ \mathbf{B} \quad \text{-->}$$
$$\lambda \mathbf{P}.\mathbf{P}(\mathbf{B}) \quad (\lambda b.\mathbf{A}(b)) \quad \text{-->}$$
$$\mathbf{A}(\mathbf{B}).$$

The following table summarizes the syntactic/semantic processes described:

(3)

RFA:	x/y y -> x		
LFA:	y y\x -> x		
FC:	x/z z/y -> x/y	:	y\x -> y\z z\x
Curry:	(x\z)/y <-> x\(z/y)		
Lifting:	x -> (z/x)\z	:	x -> z/(x\z)

0.5 Details of the Lambda-Calculus and the Representation of Nouns

A detail critical to the analysis presented concerns the relative scope of the λ-operator undergoing λ-conversion. In the following formula (parentheses inserted perspicuously):

$$[\lambda x\ [\lambda y\{\ \mathbf{P}(x),(y)\}\ (\textit{Sal}\ )\ ]\ \textit{Sid}\ ]$$

functional application (λ-conversion) can be applied once for each λ-operator, resulting in a formula with no free variables:

$$\mathbf{P}((\textit{Sal}\ ),(\textit{Sid}\ )).$$

Contrast this with the following formula, where λx has narrow scope by virtue of its position:

$$[\lambda y.\ \mathbf{P}(y),\ (\lambda x\{\mathbf{Q}(x)\})\ (\textit{Sal}\ )]\ (\textit{Sid}\ ).$$

Applying FA to λy gives:

$$[\mathbf{P}(\textit{Sal}\ ),\ (\lambda x\{\mathbf{Q}(x)\})]\ (\textit{Sid}\ ).$$

This is as far as the derivation can progress, because there is no way to communicate the buried λx to a position where it will have scope over the square brackets. In order to do that, we would have to construct our formula as follows:

$$\lambda z\{\lambda y[\mathbf{P}(y),\ (\lambda x\{\mathbf{Q}(x)\}(z))]\ (\textit{Sal}\ )\}\ (\textit{Sid}\ )$$

Now the λx term can undergo conversion, satisfying its bound variable (x) with another variable (z) which happens to have sufficient scope to take ***Sal*** as an argument, allowing all of the bound variables to be filled with arguments by successive application of FA; this is shown in Figure 4.

(4)

$$\lambda z\{\lambda y[\mathbf{P}(y),\ (\lambda x\{\mathbf{Q}(x)\}(z))]\ (\textbf{\textit{Sal}})\}\ (\textbf{\textit{Sid}}) \text{ -->}$$
$$\lambda z\{\lambda y[\mathbf{P}(y),\ (\{\mathbf{Q}(z)\})]\ (\textbf{\textit{Sal}})\}\ (\textbf{\textit{Sid}}) \text{ -->}$$
$$\lambda y[\mathbf{P}(y),\ (\{\mathbf{Q}(\textbf{\textit{Sal}})\})]\ (\textbf{\textit{Sid}}) \text{ -->}$$
$$[\mathbf{P}(\textbf{\textit{Sid}}),\ (\{\mathbf{Q}(\textbf{\textit{Sal}})\})]$$

As a word about set notation, we assume that a common noun names a set of objects, and any given noun is associated with an indeterminate which we wish to anchor to that set. To illustrate:

(5)

carrot $\in$ ***CARROT*** (hereafter ***C***)
man $\in$ ***MAN*** (***M***)
Sal $\in$ ***Sal***; Sid $\in$ ***Sid***;
etc.

Notice that proper names are names of sets containing one element--the specific individual denoted by that name.

Having made the preceding remarks, we include a summary of the syntactic/semantic categories assigned to words in Figure 6 below:

(6) SYNTACTIC TYPES

man, carrot, table: **N**	the: **(NP/REL)/N**
Sid, Sal: **NP**	on:**PP/NP**
buy: **NP\SEN/NP**	bought: **NP\S/NP**
eat: **NP\SEN/NP**	ate: **NP\S/NP**
know:**NP\SEN/NP**	knew: **NP\S/NP**
that/who: **REL/(S | NP*[2])[3]**	
AUX: **(XP\S)/(S/XP)**	
NP\S: **VP**	**NP\SEN**: **UVP**

Note: The category **SEN** is a sentence without tense markings.

SEMANTIC TYPES

carrot $\in$ ***C***;	man $\in$ ***M***; Sid $\in$ ***Sid***; Sal $\in$ ***Sal***
buy/bought:	$\lambda x\ \lambda y.\mathbf{B}(y),(x)$
know/knew:	$\lambda x\ \lambda y.\mathbf{K}(y),(x)$
eat/ate:	$\lambda x\ \lambda y.\mathbf{E}(y),(x)$
the:	$\lambda \boldsymbol{L}\ \lambda rel\ \{\exists m(m \in \boldsymbol{L} \leftrightarrow m=n) \wedge [\ rel(n)]\}$
that/who :	$\lambda(s\|np^*).(s\|np^*)$
AUX:	$\lambda x\ \lambda sen.F[sen(x)],(x)$ (where F is: do; may;...)

---

[2] Where * has the usual meaning of denoting any number of NP's (including none). This was suggested by R. T. Oehrle.

[3] The reader should not become confused by the introduction of the category REL. Although not a formal category familiar to linguists, the notation is intended to imply that the relative marker serves a syntactic/type-changing function. In terms of Montague Semantics, 'who'/'that' is of category < <e,t>, <e,t> >, a functor which takes a sentence with a gap as an argument and returns a sentence with a gap, where the output is in a form which can be an argument of a determiner, as opposed to a garden-variety sentence with a gap, which cannot be the argument of a determiner.

1.0 An Argument Against the Relative Marker as NP-Functor

The relative marker 'who', 'that', etc. has been analyzed as the main functor in those noun phrases where it exists, its syntactic category being either **(NP\NP)/(S|NP*)** or **(N\N)/(S|NP*)**. However, there are compelling arguments against choosing either of these as the representation. We present two such arguments, one involving the use of proper names as the referent of the relative clause, and the second involving adjectival scope when a relative construction is present. Proposing, as we do here, that the determiner **(NP/N)** is the main functor even in relative constructions by extending its category to **(NP/REL)/N** presents another, preferable alternative.

1.1 (NP\NP)/(S|NP*)

If the relative marker is the main functor of category **(NP\NP)/(S|NP*)**, then one would expect that a relative clause can be attached to any noun phrase we care to choose. In fact this is not the case at all, if the choice of NP happens to be a proper name. In English, proper names which have the property that they denote a specific individual cannot be relativized. If they do not have that status, that is, if they are preceded by the definite article 'the', a relative clause becomes possible. Notice the strangeness of the first member of each sentence pair when both are spoken with the same intonational contour:

(7) (i). **?Maxie who likes Sid slept late into the morning.**
The Maxie who likes Sid slept late into the morning.

(ii). **?We went to the movie with Sal who had seen it.**
We went to the movie with the Sal who had seen it.

1.2 (N\N)/(S|NP*)

Based on the preceding evidence above we might conclude that the category of the relative marker should combine with a common noun instead of a noun phrase, the category **(N\N)/(S|NP*)**, thus neatly ruling out the questionable case above while permitting the other (acceptance of the second member of the pair amounts to changing the proper name status from denoting one individual to denoting a set of individuals (***Sal* --> *SAL***, in the notation adopted here). However, that choice also turns out to be myopic when one examines the properties of semantic scope in constructions like:

(8) **the former admiral that I saw**

Assuming the **(N\N)/(S|NP*)** category for 'that', and that adjectives have the syntactic category **N/N** irrespective of their particular semantics, two possible combinatory structures exist for the above example:

(9) (i). **[former admiral]$_N$ [that I saw]$_{N\backslash N}$**
(ii). **[former]$_{N/N}$ [admiral that I saw]$_N$.**

Since both (9i) and (9ii) are produced by FA (LFA and RFA respectively), neither can be argued preferable on syntactic grounds. However, based on a theory of compositional semantics, the two structures have vastly different interpretations. The scope of 'former' in (9i) is narrow, while the scope of the relative is wide-- the overall interpretation being 'I saw a someone who used to be an admiral'. On the other hand, in (9ii) the scope of 'former' is wide, while the scope of the relative is narrow--the interpretation is something like 'I saw an admiral, but I don't anymore'. This is predicted to be an ambiguous sentence, which is not the case; the latter reading (9ii) is not a reading available to English speakers. Clearly, the choice of the relative marker as the main functor of the noun phrase is not unproblematic.

1.3 (NP/REL)/N

The choice of the determiner as the overall functor in these constructions is thus empirically motivated, and, as we show below, accounts for the above difficulties. We begin with the semantics of the definite article corresponding to the syntactic category **NP/N**. 'the' assumes that there exists one and only one of the noun it precedes in the discourse. For example, imagine a situation of two tennis balls on a table. A phrase such as 'the tennis ball on the table belongs to me' is not appropriate here, because the listener cannot pick out a unique referent. Similarly, the same phrase is inappropriate if no tennis balls are on the table, or if there has been no reference to a tennis ball prior to its utterance. The situation where a phrase containing 'the tennis ball...' is appropriate exists if and only if speaker and hearer have mentioned a tennis ball, and have decided that that very tennis ball is to be referred to again. Thus 'the' picks out the existence of one and only one referent. This interpretation can be represented in triplet notation as:

(10)

the
NP/N
$\lambda L\ [\exists m(m \in L \leftrightarrow m=n)]$

where L is the set with which the common noun is to be identified. Extending the syntactic category of 'the' to **(NP/REL)/N**, where the noun **N** will be associated with the gap in the relative clause amounts to forming the logical conjunct of the interpretation of 'the' with the associated relative clause:

(11)

the
(NP/REL)/N
$\lambda L\ \lambda rel\ [\exists m(m \in L \leftrightarrow m=n) \wedge rel(n)]$

Having described an alternative to the relative marker as functor, it is now our task to show why this alternative is preferred. To that end, we repeat the above two NP types and derive the representation of their interpretations (for clarity, the derivations begin on the following page):

(12) (i). **?Maxie who likes Sid**
the Maxie who likes Sid
(iii). **the former admiral that I saw**

(i)

the Maxie who likes Sid
NP
$[\exists m(m \in \boldsymbol{MAXIE} \leftrightarrow m=n) \wedge L(\boldsymbol{Sid}),(n)]$

FA (semantics only)

the Maxie who likes Sid
NP
$[\exists m(m \in \boldsymbol{MAXIE} \leftrightarrow m=n) \wedge \{\lambda s.L(\boldsymbol{Sid}),(s)\}(n)]$

RFA

the Maxie	who likes Sid
NP/REL	**REL**
$\lambda rel.[\exists m(m \in \boldsymbol{MAXIE} \leftrightarrow m=n) \wedge rel(n)]$	$\lambda s.L(\boldsymbol{Sid}),(s)$

RFA

the	Maxie
(NP/REL)/N	**N**
$\lambda L\ \lambda rel.\ [\exists m(m \in L \leftrightarrow m=n) \wedge rel(n)]$	***MAXIE***

who likes Sid
REL
$\lambda s.L(\boldsymbol{Sid}),(s)$

RFA

who	likes Sid
REL/(S|NP*)	**NP\S**
$\lambda(s\|np^*).(s\|np)$	$\lambda s.L(\boldsymbol{Sid}),(s)$

RFA

likes	Sid
(NP\S)/NP	**NP**
$\lambda o \lambda s.L(o),(s)$	***Sid***

(iii)

the former admiral that Sal saw
NP
$[\exists m(m \in \mathbf{former}(\boldsymbol{A}) \leftrightarrow m=n) \wedge \mathbf{S}(n),(\boldsymbol{Sal})]$

FA (semantics only)

the former admiral that Sal saw
NP
$[\exists m(m \in \mathbf{former}(\boldsymbol{A}) \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{S}(o),(\boldsymbol{Sal})\}(n)]$

RFA

the former admiral
NP/REL
$\lambda rel.\ [\exists m(m \in \mathbf{former}(\boldsymbol{A}) \leftrightarrow m=n) \wedge rel(n)$

that Sal saw
REL
$\lambda o.\mathbf{S}(\mathbf{o}),(\boldsymbol{Sal})$

RFA

the
(NP/REL)/N
$\lambda \boldsymbol{L}\ \lambda rel.\ [\exists m(m \in \boldsymbol{L} \leftrightarrow m=n) \wedge rel(n)]$

former admiral
N
$\mathbf{former}(\boldsymbol{A})$

RFA

former
N/N
$\lambda x.\mathbf{former}(x)$

admiral
N
$\boldsymbol{A}$

RFA

that Sal saw
REL
$\lambda o.\mathbf{S}(o),(\boldsymbol{Sal})$

We hope the preceding derivations show that for each case there exists a unique (unambiguous) formal representation which corresponds to the interpretation available to the speaker. Further, the desired result that proper names cannot be relativized follows as a direct consequence of the way the we constructed the types. Thus, choosing the definite article as overall functor in these constructions is empirically and theoretically sound.

2. Extraction From a Relative Clause

Now let us turn to the syntactic and semantic construction of a sentence with a relative construction, and look at extraction from out of that clause. Based on the machinery we have developed thus far, we produce a formal representation of the following sentence (the derivation, which is completely straightforward, has been omitted):

(13)

Sal knew the man who bought the carrot.
S
$K([\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge B(n),(\mathbf{the}(\boldsymbol{C}))]),(\boldsymbol{Sal})$

Consider the result of extracting out of the relative clause by questioning its object:

(14) ***What did Sal know the man who bought?**

This is clearly word salad. Not only do native speakers find the sentence unacceptable, butthey also have difficulty assigning any concrete interpretation to it at all. Theories of grammar which involve movement or feature-systems are forced to invoke separate (stipulated) constraints in one form or another to block this type of movement/construction. However, as the next example shows, the use of the lambda-calculus to analyze such constructions rules them out directly, without requiring us to construct a theory in anticipation of them.

The mechanics of the analysis are not subtle; it is simply the scope of the λ-operators and the semantics of the determiner 'the' which do the work for us.

(15)

who	bought
REL/(S|NP*)	(NP\S)/NP
$\lambda(s\|np^*).(s\|np^*)$	$\lambda o \lambda s \mathbf{B}(o),(s)$

FC

who bought
REL/ NP
$\lambda o \lambda s.\mathbf{B}(o),(s)$

the	man
(NP/REL)/N	N
$\lambda \boldsymbol{L}\ \lambda rel.\ [\exists m(m \in \boldsymbol{L} \leftrightarrow m=n) \wedge rel(n)]$	$\boldsymbol{M}$

RFA

the man	who bought
NP/REL	REL/NP
$\lambda rel.\ [\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge rel(n)]$	$\lambda o \lambda s.\mathbf{B}(o),(s)$

FC

the man who bought
NP/NP
$[\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o \lambda s.\mathbf{B}(o),(s)\}(n)]$

CUR(semantics only)

the man who bought
NP/NP
$[\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda s \lambda o.\mathbf{B}(o),(s)\}(n)]$

FA (semantics only)

Notice that $\underline{\lambda o}$ does not have sufficient scope to undergo further FA.

the man who bought
NP/NP
$[\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{B}(o),(n)\}]$

know	the man who bought
UVP/NP	**NP/NP**
$\lambda o \lambda s \mathbf{K}(o),(s)$	$[\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{B}(o),(n)\}]$

FC

Sal	know the man who bought
NP	**UVP/NP**
Sal	$\lambda s\mathbf{K}(o),([\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{B}(o),(n)\}])$

LFA

did	Sal know the man who bought
(XP\S)/(S/XP)	**SEN/NP**
$\lambda x \lambda sen.\mathbf{D}[sen(x)],(x)$	$\mathbf{K}([\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{B}(o),(n)\}]),(\boldsymbol{Sal})$

RFA

what	did Sal know the man who bought
NP	**NP\S**
$\boldsymbol{W}$	$\lambda x\mathbf{D}[\mathbf{K}([\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \{\lambda o.\mathbf{B}(o),(n)\}]),(\boldsymbol{Sal})(x)],(x)$

LFA

*What did Sal know the man who bought?
S
$\mathbf{D}[\mathbf{K}(\{\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \lambda o.\mathbf{B}(o),(n)\}),(\boldsymbol{Sal})(\boldsymbol{W})],(\boldsymbol{W})$

Let us examine the result of the derivation just accomplished:

(16)

*What did Sal know the man who bought?
S
$D[K(\{\exists m(m \in \boldsymbol{M} \leftrightarrow m=n) \wedge \lambda o.B(o),(n)\}),(Sal)(W)],(W)$

From the category of the result, S, we know that the syntax was able to parse the entire string. If we had only syntactic operations at our disposal, we would be forced to appeal to some constraint separate from the combinatory rules we developed. Thankfully, this is not the case. The analogous processes developed for the semantics have not only ruled it out on the basis of the scope relations of the λ-operators, but also provided a possible account why the sentence has no definable meaning: a deeply embedded, unresolved variable exists in the input. Because of that embedding, the hearer can find no context to fill the variable, and confusion results.

3. Conclusion

Although we have analyzed only a small class of sentences within this domain, the result obtained is a compelling one. Stipulations forbidding extraction from relative clauses abound in theories of grammar, but deriving this result as a direct consequence of the formalism adopted here, with no premeditation on our part, is a credit to its power, applicability, and overall simplicity of form. It remains to be shown just how far this approach can lead in grammatical studies without further refinement, and, in terms of the analysis presented, how many so-called "subjacency effects" can be accounted for as results of the internal properties of the framework including both categorial grammar and the lambda-calculus.

Bibliography

Ajdukiewicz, K. (1967). "Syntactic connexion", in Polish Logic 1920-1939, Storrs McCall, ed., Oxford University Press, Oxford.

Curry, H. B. (1961). "Some logical aspects of grammatical structure", in R. Jakobson (ed.), Structure of Language and Its Mathematical Aspects. Proceedings of the Twelfth Symposium on Applied Mathmatics,pp 56-68, American Mathematical Society, Providence, RI.

Dowty, D. R., R. Wall, and S. Peters. (1981). Introduction to Montague Semantics, Studies in Linguistics and Philosophy, v. 11, Reidel, Dordrecht.

Gazdar, G., E. Klein, G. Pullum and I. Sag. (1985). Generalized Phrase Structure Grammar, Harvard University Press, Cambridge.

Lambek, J. (1958). "The mathematics of sentence structure", American Mathematical Monthly, v. 56, pp154-169.

Lyons, J. (1977). Semantics, v.I and II, Cambridge University Press, Cambridge.

Oehrle, R. T. (1974). "Some remarks on the painting of Rembrandt," in CLS 10, pp 504-516.

Oehrle, R. T., E. Bach, and D. Wheeler, eds. (1988). Categorial Grammars and Natural Language Structures, Studies in Linguistics and Philosophy, v.32, Reidel, Dordrecht.

Radford, A. (1981) Transformational Syntax, Cambridge University Press, Cambridge.

Steedman, M. (1985). "Dependency and Coordination in the grammar of Dutch and English", Language, v. 61, no. 3, pp 523-568.