Constructive models of extraction parameters[1]

Dick Oehrle
Linguistics & Cognitive Science
University of Arizona

Recursion as the basis of long-distance dependencies

An important and central insight of Tree Adjoining Grammar is its factorization of local dependencies—handled through local INITIAL TREES—and recursion—handled through AUXILIARY TREES and successive applications of the ADJUNCTION operation. Many different frameworks of grammatical description have converged on a conceptually similar distinction. In the transformational tradition, the idea of long-distance movement—movement across an 'essential variable'—has been abandoned in favor of sequences of short-distance hops (or checks). In feature-based phrase structure grammars such as GPSG and HPSG, the analog of recursive movement is the transitive closure of local consistency conditions on local trees containing the SLASH feature.

At first glance, then, this convergence in a variety of theoretical approaches suggests that recursion in some form is the essential engine in the characterization of natural language long-distance dependencies. And this assumption might lead us to the following thesis concerning the relation between recursion and extraction.

> *Thesis: if $\Gamma[\alpha]$ is a well-formed expression of category A containing a gap α of category B and $\Delta[\beta]$ is a well-formed expression of category B containing a gap β of category C, then the result of replacing the gap α in $\Gamma[\alpha]$ with $\Delta[\beta]$, which we write $\Gamma[\Delta[\beta]]$ is a well-formed expression of category A containing a gap β of category C.*

As an example of a case which might be adduced in support of this thesis, consider the unbounded nature of extraction from noun phrases, as discussed by Kroch [6]. The well-formedness of Which painting did you see? indicates that did you see is a well-formed expression containing a gap of type *np*, and the well-formedness of Which painting did you see a photograph of? and Which painting did you see a copy of? suggests (in a way consistent with the thesis) that a photograph of and a copy of are well-formed *np*'s containing *np* gaps. Accordingly, the thesis, if correct, requires that Which painting did you see a copy of a photograph of? also be well-formed, as indeed it is. Yet this simple and elegant thesis concerning recursion encounters well-known difficulties, which

[1]This paper is the product of joint work with Michael Moortgat, with whom a more comprehensive treatment of these questions is under preparation. This work has been supported by the National Science Foundation under Grant No. SBR-9510706, which we gratefully acknowledge.

have been construed as supporting additional theoretical devices such as filters and other forms of surface constraints. The goal of this paper is to show in the most direct possible way that in one well-known case, it is possible to formulate recursive principles in a way that obviates the need for additional theoretical mechanisms and, at the same time, offers a simple formal characterization of a proposed typological distinction of long-standing interest.

A typological parameter

As Perlmutter [15, 16] first observed, extraction from the *np*-position following a complementizer is possible in some languages, but not in others. Thus, we have:

(1a)	**French**	Marie se demandait qui Jean a dit que Martin a vu? (after [16]) 'Marie wondered who Jean said that Martin saw?'
(1b)		*Marie se demandait qui Jean a dit que a vu Martin? (after [16]) 'who did he say saw Martin?'
(2a)	**English**	Marie wondered who Jean said that Martin saw?
(2b)		*Mary wondered who Jean said that saw Martin?
(3a)		Mary wondered who Jean said Martin saw?
(3b)		Mary wondered who Jean said saw Martin?
(4a)	**Nederlands**	Wie zei Marie dat die appel opgegeten heeft? (after [8]) who said Marie that this apple eaten has 'who did Marie say ate this apple?'
(4b)		Wie zei Marie dat Martin gezien heeft? (after [8]) 'who did you say Martin saw'
	OR	'who did Marie say saw Martin'

There are two basic strategies to deal with these issues. The first is to propose general grammatical rules (selectively chosen by each language) which generate exactly the grammatical examples and fail to generate the ungrammatical examples; the second is to propose general grammatical principles which generate all the good examples and couple these principles wih constraints (selectively chosen by each language) which weed out particular cases. We call the first strategy 'constructive' and the second 'co-constructive'. There have been many co-constructive proposals to account for the above phenomena: we mention here only [15, 16, 1, 2]. In the sections to follow, we develop simple and appealingly symmetrical constructive accounts of these constrasting systems of extraction.

The framework

We work in the framework of multi-modal grammatical logic [10, 11, 4, 9, 14, 12], a framework we describe here only in enough depth to support the goals of this paper. From this perspective, the problem of grammatical composition, within and across such different dimensions of linguistic structure, is regarded as an inference problem: the component pieces of a complex linguistic structure are taken to be the premisses of a deductive problem, and its global structure to be a conclusion deducible from these premisses in a system of grammatical inference. Thus, grammaticality is identified with validity within this system. Moreover, the formal system characterizing validity offers a natural model, in the style of denotational semantics for programming languages [17], of the cognitive computation that must be assumed to provide the basis for real-time understanding of running speech.[2] Thus the logical methods described here are not introduced in a blind search for formal rigor; on the contrary, they are introduced because they provide an armentarium of subtle and suitable tools and methods that allow us to probe the properties of grammatical reasoning.

In such a system, if A is deducible from a structured set of premisses Γ, we write $\Gamma \Rightarrow A$. It is reasonable to suppose that the deducibility relation is reflexive and transitive: that is, for every formula A, we have $A \Rightarrow A$; and for every triple of formulas A, B, C, if $A \Rightarrow B$ and $B \Rightarrow C$, then $A \Rightarrow C$.

A *uni-modal* deductive system contains a single way (or *mode*) of putting resources premisses together. To reason about this mode, we introduce a product operator—a form of conjunction—together with its residuals (or adjoints)—forms of implication. For example, given a binary mode of composition, we have a product $\bullet$ and two directionally-sensitive implications written, as in the categorial tradition, / and \. Every product and its adjoints are connected by the basic adjointness laws. In the binary case, as here, these take the form:

$$A \Rightarrow C/B \quad \textit{iff} \quad A \bullet B \Rightarrow C \quad \textit{iff} \quad B \Rightarrow A\backslash C$$

As a simple illustration of the consequences of the adjointness laws, take A to be C/B; by reflexivity, we have $C/B \Rightarrow C/B$; using the first adjointness law (left to right), we have $(C/B) \bullet B \Rightarrow C$. This is called the *co-unit* of the adjunction and is also known variously as Modus Ponens (in the logical literature) or (functional) application (in the categorial literature).

There are a number of different presentations of this system of pure binary residuation logic: Gentzen style, natural deduction, Hilbert-style, proof nets. These can be easily shown to be equivalent with regard to provability and we identify them all with the non-associative Lambek calculus **NL** [7].

Keeping the logical rules expressed by the adjointness laws invariant, we may obtain other logical systems by adding *structural rules* [3, 5], such as the following:

RAssoc	$(A \bullet B) \bullet C \Rightarrow A \bullet (B \bullet C)$
LAssoc	$A \bullet (B \bullet C) \Rightarrow (A \bullet B) \bullet C$
Perm	$A \bullet B \Rightarrow B \bullet A$
Contr	$A \Rightarrow A \bullet A$
RWeak	$A \bullet B \Rightarrow A$
LWeak	$A \bullet B \Rightarrow B$

[2] Analogously, we may think of models of the unfolding processes of speech comprehension at the psychological and neurological levels as approximations, at different levels of scale, of the operational semantics of this process.

The presence or absence of these rules defines a family of unimodal logics of conjunction and implication, some of whose members (with characteristic arrows) are:

logic	*structural rules*	*arrows*
NL	none	$(A/B) \bullet B \Rightarrow A$, $B \Rightarrow (A/B)\backslash B$
L	*RAssoc, LAssoc*	$A/B \Rightarrow (A/C)/(B/C)$, $A\backslash(B/C) \Rightarrow (A\backslash B)/C$
LP	*RAssoc,LAssoc,Perm*	$A/B \Rightarrow B\backslash A$, $(A/B)/C \Rightarrow (A/B)/C)$

When a particular formula is provable in a particular logical system, we indicate this using Frege's symbol $\vdash$. Thus,

$$\mathbf{NL} \vdash s/(np\backslash s) \bullet (s/(np\backslash s)\backslash s) \Rightarrow s$$

$$\mathbf{L} \vdash vp/np \Rightarrow (vp/pp)/(np/pp)$$

If a formula is not provable in a particular system, we draw a slash through the turnstile, as in

$$\mathbf{NL} \nvdash vp/np \Rightarrow (vp/pp)/(np/pp)$$

From this general perspective, then, binary unimodal deductive systems are definable simply by specifying, once and for all, what structural rules the single mode of composition enjoys.

Although the applicability of these systems to the analysis of natural language properties has been the subject of intense scrutiny, it is clear that natural languages differ from unimodal deductive systems in an essential way. Namely, they exhibit a much more subtle control of inference than the all or nothing choice of structural rules allows. For example, individual languages often exhibit varying sensitivity to order. Japanese and Korean, for example, are strict about the position of the tensed verb in a clause but not strict about the position of the arguments preceding the verb. This suggests a richer deductive system, one based on multiple modes of combination.[3] Each mode has a fixed arity, an associated product operator of that arity and an implication for each argument position, satisfying the adjointness laws. Each mode is associated with a set of structural rules. However, something new arises as well: structural postulates involving more than one mode.

As an illustration which will be important in the sequel, consider a system with a single binary mode, associated with the binary product $\bullet$ and adjoints / and \, and a single unary mode, associated with a unary operator $\Diamond$ and a single adjoint $\Box^{\downarrow}$. The adjointness laws for the unary operator take the form:

$$\Diamond A \Rightarrow B \quad \textit{iff} \quad A \Rightarrow \Box^{\downarrow} B$$

[3]In fact, the presence of more than one mode of combination is implicit in linguistic practice: phonologists and morphologists have recognized different kinds of boundaries between elements; $\overline{X}$-bar theory recognizes different modes of combination ('spec-head' relation, for example) at different levels.

Just as we derived the co-unit above by starting with the sequent $C/B \Rightarrow C/B$, if take A above to be $\Box^{\downarrow}B$, then the right hand side holds by reflexivity and the left hand side gives us a unary counterpart to Modus Ponens:[4]

$$\Diamond\Box^{\downarrow}A \Rightarrow A$$

In other words, if the unary operator $\Diamond$ has an adjoint, then the composition of $\Diamond\Box^{\downarrow}$ has an interesting property: it can play a role in part of a deduction and then disappear. This property is the first of two crucial properties of multi-modal type logic we will need below. The second, a small set of structural rules involving the interaction of $\Diamond$ and $\bullet$, will be developed below, after we prepare the ground by developing some very small fragments which will support the illustration of the extraction parameters of interest here.

Fragments without extraction

We now develop the simplest possible fragments of French, English, and Dutch without extraction which can be directly extended to support the extraction constructions of interest. The many points of grammatical interest that these fragments touch on that are not directly relevant to the problem at hand will be systematically ignored. The logical framework is simply the pure residuation logic NL: $\bullet$, $/$, and $\backslash$ connected by the adjointness laws; no added structural rules. From this point of view, all that remains to be added is a set of atomic formulas (categories), common to all the fragments, and a set of lexical assumptions associating basic expressions with formulas.

[4]One may connect this straightforwardly with the binary case discussed earlier by regarding the product $A \bullet B$ as the result of applying the unary operator $A \bullet -$ to B. This unary operator may be regarded as a modality $\Diamond_A$, whose corresponding adjoint $\Box^{\downarrow}_A$ is the unary operator $A\backslash-$ which yields $A\backslash B$ when applied to B. Applying the unary adjointness law in this case, we have

$$\Diamond_A B \Rightarrow C \quad \textit{iff} \quad B \Rightarrow \Box^{\downarrow}_A C$$

But this is just another way of writing

$$A \bullet B \Rightarrow C \quad \textit{iff} \quad B \Rightarrow A\backslash C$$

Similarly, we can write $A \bullet B$ as the unary operator $\Diamond_B$ applied to A, and regard C/B as $\Box^{\downarrow}_B C$. Applying the unary adjointness law here gives

$$A \bullet B \Rightarrow C \quad \textit{iff} \quad A \Rightarrow C/B.$$

atom	*vernacular category*
s	sentence
is	inverted sentence
fs	verb-final clause
np	noun phrase (including proper names)
$partp$	participle phrase
c	*that*-clause, *que*-clause, *dat*-clause

The full set of formulae (categories) is obtained as usual by closing the set of atoms under the binary type constructors $\bullet$, /, and \.

The lexical declarations we need are given in the table below:[5]

language	*category*	*lexical inhabitants*
fr*nch	np	Marie, Jean, Martin
	$(np\backslash s)/partp$	a
	$partp/np$	vu
	$partp/c$	dit
	c/s	que
***ngl*sh**	np	Marie, Jean, Martin
	$(np\backslash s)/np$	saw
	c/s	that
	$(np\backslash s)/c$	said
	$((np\backslash s)/(np\backslash s))/np$	said
d*tch	np	Marie, Martin, die appel
	$np\backslash partp$	opgegeten, gezien
	$np\backslash(partp\backslash fs)$	heeft
	c/fs	dat
	$(is/c)/np$	zei

When word w inhabits category t, we write $w \Rightarrow t$.

For any logical system λ, a lexical type assignment ω is extended to *binarily bracketed sequences* of words in the standard way: thus, if I is an appropriate index set and $\prod_{i\in I} w_i$ is a binarily bracketed sequence of words and τ is a formula, if there are categories $\{\tau_i\}_{i\in I}$ such that $\omega \vdash w_i \Rightarrow \tau_i$ and

$$\lambda \vdash \textstyle\prod_{i\in I} \tau_i \Rightarrow \tau$$

To show both dependencies, we may indicate that such a situation holds by

$$\lambda, \omega \vdash \textstyle\prod_{i\in I} w_i \Rightarrow \tau$$

[5]The presence of asterisks is to emphasize the fragmentary character of these simple grammatical systems.

For example, we have

$$\text{NL}, \text{fr*nch} \vdash \mathsf{Jean} \bullet (\mathsf{a} \bullet (\mathsf{vu} \bullet (\mathsf{Martin}))) \Rightarrow s$$

because

$$\begin{array}{l} \text{fr*nch} \vdash \mathit{Jean} \Rightarrow np \\ \text{fr*nch} \vdash \mathsf{a} \Rightarrow (np\backslash s)/partp \\ \text{fr*nch} \vdash \mathsf{vu} \Rightarrow partp/np \\ \text{fr*nch} \vdash \mathsf{Martin} \Rightarrow np \\ \text{AND} \\ \text{NL} \vdash (np \bullet ((np\backslash s)/partp \bullet (partp/np \bullet np))) \Rightarrow s \end{array}$$

The first four lines come directly from our lexical assumptions; the final line can be straightforwardly demonstrated as displayed in the proof tree below, where inference steps are marked with t, a, or r, according to whether they depend on transitivity, adjointness, or reflexivity, respectively.[6]

$$\dfrac{\dfrac{\dfrac{}{partp/np \bullet np \Rightarrow partp}\,a,r \quad \dfrac{}{(np\backslash s)/partp \bullet partp \Rightarrow np\backslash s}\,a,r}{(np\backslash s)/partp \bullet (partp/np \bullet np) \Rightarrow np\backslash s}\,t \quad \dfrac{}{np \bullet np\backslash s \Rightarrow s}\,a,r}{(np \bullet ((np\backslash s)/partp \bullet (partp/np \bullet np))) \Rightarrow s}\,t$$

Similarly, as the reader is invited to show, we have:

$$\begin{array}{ll} \text{NL}, \text{fr*nch} & \vdash \mathsf{Marie} \bullet (\mathsf{a} \bullet (\mathsf{dit} \bullet (\mathsf{que} \bullet (\mathsf{Jean} \bullet (\mathsf{a} \bullet (\mathsf{vu} \bullet (\mathsf{Martin})))))) \Rightarrow s \\ \text{NL}, \text{*ngl*sh} & \vdash \mathsf{Jean} \bullet (\mathsf{said} \bullet (\mathsf{that} \bullet (\mathsf{Martin} \bullet (\mathsf{saw} \bullet \mathsf{Marie})))) \Rightarrow s \\ \text{NL}, \text{d*tch} & \vdash \mathsf{zei} \bullet (\mathsf{Marie} \bullet (\mathsf{dat} \bullet (\mathsf{Martin} \bullet ((\mathsf{die\ appel} \bullet \mathsf{gegeten}) \bullet \mathsf{heeft})))) \Rightarrow is \end{array}$$

These fragments are of course extremely simple. This is obvious at the lexical level, since each fragment contains fewer than 10 words and speakers of natural languages are estimated to know

[6]Actually, we let t stand for a generalization of transitivity which is easily shown to be valid in the presence of the adjointness laws. We illustrate with a simple special case. Suppose $A \Rightarrow B$ and $C \bullet B \Rightarrow D$. By adjointness,

$$C \bullet B \Rightarrow D \quad \textit{iff} \quad B \Rightarrow C\backslash D$$

By our second premise, the lefthand side holds; thus, the righthand side holds; by our first premise and transitivity, we have $A \Rightarrow C\backslash D$; taking this as the righthand side of the adjointness law, the lefthand side gives us $C \bullet A \Rightarrow D$. Thus, we have proved the derived rule of inference (with premisses represented on top of the line and conclusion below):

$$\frac{A \Rightarrow B \quad C \bullet B \Rightarrow D}{C \bullet A \Rightarrow D}$$

By an easy inductive argument, this simple result can be generalized to show that we can generalize transitivity to substitution inside a product of arbitrary depth.

on the order of tens of thousands of words. This can be remedied in part by enriching the lexicon. But enriching the lexicon is not in and of itself a sufficient remedy.

In the next section, we will examine the well-known inadequacies of **NL** as a logic of extraction and show how simple extensions of it can accommodate the properties of interest here of languages like French, English, and Dutch.

Extraction: preliminaries

An embedded question, such as qui a vu Martin in a French sentence such as Jean s'est demandé qui a vu Martin or who saw Martin in an English sentence like Jean wondered who saw Martin, consists of two basic parts: the question word *who* and the *body*—the clausal remnant saw Martin. Although the system **NL** is too weak to deal adequately with French or English embedded questions, its type system can handle this particular case and shows the way toward a system that handles a much broader range of cases.

We begin with the following fact, which follows directly from the lexical properties of the words in question by the adjointness laws:

$$\mathbf{NL}, \mathbf{*ngl*sh} \vdash \text{saw} \bullet \text{Martin} \Rightarrow np\backslash s$$

Now, writing cq for the type of an embedded question, adjointness allows us to solve for the unknown type x in the sequent

$$(x \bullet (np\backslash s)) \Rightarrow cq \quad \textit{iff} \quad x \Rightarrow cq/(np\backslash s)$$

Thus, adding cq to our stock of atoms and extending our lexical assignment by the declaration who $\Rightarrow cq/(np\backslash s)$, we can prove:

$$\text{Jean} \bullet (\text{wondered} \bullet (\text{who} \bullet (\text{saw} \bullet \text{Martin}))) \Rightarrow s$$

This analysis is lexically extendable to embedded questions with complementizer whether, by the addition of the lexical type declaration

$$\text{whether} \Rightarrow cq/s$$

But further generalizations within the system **NL** are only possible if completely unacceptable forms of lexical polymorphism are allowed. For example, to treat the embedded question who Martin saw from this perspective, we would need to be able to assign a type to Martin saw, which requires a new type $np\backslash(s/np)$ for saw, relative to which we can show Martin $\bullet$ saw $\Rightarrow s/np$. But we also need a new type for who, $cq/(s/np)$, in order to be able to derive who Martin saw as a cq. Switching the basic inference system from **NL** to **L** by adding the two Associativity rules allows one to combine all the cases in which the gap is rightmost into a single category (since it is possible to show that in the presence of Associativity that all clausal remnants with a single, final np gap belong to the type s/np), but distinct types are still needed for intial and final gaps and non-peripheral cases still remain.

Before proceeding further, it is worthwhile to take stock of the situation. We seek a system of inference with the following properties:

1. there is a type ξ such that we may take who, for example, to be of type $cq/(\xi\backslash s)$ and we may show using hypothetical reasoning, that the body is provably of type $\xi\backslash s$;

2. to show that the body is of type $\xi\backslash s$, we must be able to show

$$\xi \bullet [\text{body}] \Rightarrow s$$

   This step requires communication between the hypothetical premise ξ and the position of the gap inside the body of the embedded question;

3. communication between the hypothetical premise ξ and the position of the gap must be statable by logical principles; and

4. the additional logical principles allowing communication between the hypothetical premise ξ and the position of the gap must not lead to overgeneration (as occurs if we extend our logical system from **NL** to **LP** by adding both the Associativity Rules and Permutation; while this would allow communication between the hypothetical premise ξ and any possible position in the body, it would also completely destroy the possibility of distinguishing expressions by the order of their components (just as the associativity rules destroy the possibility of distinguish expressions by the grouping of sub-expressions)).

All these desiderata can be simultaneously satisfied in a simple multi-modal system of grammatical inference.

Extraction: a multi-modal approach

Extend **NL** by the addition of a unary mode associated with the unary type constructor $\Diamond_{wh}$ and its adjoint $\Box^{\downarrow}_{wh}$, to form the system we shall refer to as $\mathbf{NL}_{\Diamond_{wh}}$. Recall that by the adjointness laws, we have

$$\Diamond_{wh}\Box^{\downarrow}_{wh}A \Rightarrow A$$

Now, if we assume that the *single* type assignment in our fragment for who is

$$\text{who} \Rightarrow cq/(\Diamond_{wh}\Box^{\downarrow}_{wh}np\backslash s),$$

then we can treat who saw Martin as an embedded question, since we have

$$\mathbf{NL}_{\Diamond_{wh}} \vdash (cq/(\Diamond_{wh}\Box^{\downarrow}_{wh}np\backslash s) \bullet np\backslash s) \Rightarrow cq.$$

It is worth seeing how the proof of this theorem unfolds, in order to appreciate the deductive role played by the modalities.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Diamond_{wh}\Box^{\downarrow}_{wh}np \Rightarrow np}\,\textit{unary a!,r} \quad \dfrac{}{np \bullet (np\backslash s) \Rightarrow s}\,a,r}{(\Diamond_{wh}\Box^{\downarrow}_{wh}np \bullet (np\backslash s)) \Rightarrow s}\,t}{np\backslash s \Rightarrow \Diamond_{wh}\Box^{\downarrow}_{wh}np\backslash s}\,a \qquad \dfrac{}{(cq/((\Diamond_{wh}\Box^{\downarrow}_{wh}np)\backslash s) \bullet (\Diamond_{wh}\Box^{\downarrow}_{wh}np)\backslash s) \Rightarrow cq}\,a,r}{(cq/((\Diamond_{wh}\Box^{\downarrow}_{wh}np)\backslash s) \bullet np\backslash s) \Rightarrow cq}\,t$$

Thus, for the special case in which the body of the embedded question is of type $np\backslash s$, we now have two types for who which satisfy all our desiderata (some vacuously), namely the **NL**-type $cq/(np\backslash s)$ and the $\mathbf{NL}_{\Diamond_{wh}}$-type $cq/(\Diamond_{wh}\Box^{\downarrow}_{wh}np\backslash s)$. We have already seen that the first of these is difficult to extend uniformly to a larger range of relevant cases, for at least two reasons:

- atomic categories like np are not part of the logical vocabulary, so our logical system cannot formulate general laws in terms of particular atoms;
- on the other side of the coin, formulating filler-gap communication in terms of particular atoms would miss the point, since similar communication rules hold with respect to other atomic categories (such as ap and pp).

In fact, in standard generative syntax, these problems were recognized very early, and movement rules were formulated not with regard to particular categories, but with regard to a particular feature (or set of features), such as $[+wh]$. But in contrast to the inert feature $[+wh]$, which has no intrinsic logical behavior, the type constructor $\Diamond_{wh}$ is a logical operator, with an adjoint $\Box^{\downarrow}_{wh}$. But over and above the behavior of the operator $\Diamond_{wh}$ with its adjoint $\Box^{\downarrow}_{wh}$ (which plays a role in the proof displayed above), as a product operator, $\Diamond_{wh}$ can also appear in interaction rules, connecting it with other operators.

We have already seen how the type $cq/(\Diamond\Box^{\downarrow}np\backslash s)$ accounts for French, English, and Dutch sentences such as:

French Jean se demandait qui a vu Martin.
Jean refl asked:impf who has seen Martin
'Jean was wondering who saw Martin.'

English Jean was wondering who saw Martin.

Dutch Jan vroeg zich af wie slaapte
Jan asked refl who slept
'Jan wondered who slept.'

The next simplest step of communication between filler and gap involves sentences such as:

French Jean se demandait qui Martin connait.
Jean refl asked:impf who Martin knows
'Jean was wondering who Martin knows.'

English Jean was wondering who Martin saw.

Dutch Jan vroeg zich af wie Martin plaagte
Jan asked refl who Martin teased
'Jan wondered who Martin teased.'

In French and English, these sentences will be derivable if we add the following interaction postulate:

$$\vec{K}\,2r \quad \Diamond_{wh}A \bullet (B \bullet C) \Rightarrow B \bullet (C \bullet \Diamond_{wh}A).$$

In Dutch, the required interaction postulate is:

$$\vec{K}\,2l \quad \Diamond_{wh}A \bullet (B \bullet C) \Rightarrow B \bullet (\Diamond_{wh}A \bullet C).$$

These postulates are pleasantly symmetric. To see that they do what we say they do, look at the proofs below:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{(np\backslash s)/np \bullet np \Rightarrow np\backslash s}\,a,r \quad \dfrac{}{np \bullet np\backslash s \Rightarrow s}\,a,r}{(np \bullet ((np\backslash s)/np \bullet np)) \Rightarrow s}\,t}{(np \bullet ((np\backslash s)/np \bullet \Diamond\Box^{\downarrow}np)) \Rightarrow s}\,a}{(\Diamond\Box^{\downarrow}np \bullet (np \bullet (np\backslash s)/np)) \Rightarrow s}\,\vec{K}\,2r!}{(np \bullet (np\backslash s)/np) \Rightarrow (\Diamond\Box^{\downarrow}np)\backslash s}\,a \quad \dfrac{}{(cq/((\Diamond\Box^{\downarrow}np)\backslash s) \bullet ((\Diamond\Box^{\downarrow}np)\backslash s)) \Rightarrow cq}\,a,r}{cq/((\Diamond\Box^{\downarrow}np)\backslash s) \bullet (np \bullet (np\backslash s)/np) \Rightarrow cq}\,t}{\text{who} \bullet (\text{Martin} \bullet \text{saw}) \Rightarrow cq}\,lex$$

$$
\dfrac{
  \dfrac{
    \dfrac{}{(np \bullet np\backslash s) \Rightarrow s}\,{\scriptstyle a,r}
    \qquad
    \dfrac{
      \dfrac{}{\Diamond\Box^{\downarrow} np \Rightarrow np}\,{\scriptstyle a}
      \quad
      \dfrac{}{(np \bullet np\backslash(np\backslash s)) \Rightarrow np\backslash s}\,{\scriptstyle a,r}
    }{(\Diamond\Box^{\downarrow} np \bullet np\backslash(np\backslash s)) \Rightarrow np\backslash s}\,{\scriptstyle t}
  }{
    \dfrac{
      \dfrac{(np \bullet (\Diamond\Box^{\downarrow} np \bullet np\backslash(np\backslash s))) \Rightarrow s}{(\Diamond\Box^{\downarrow} np \bullet (np \bullet np\backslash(np\backslash s))) \Rightarrow s}\,\vec{K}\,2l!
    }{np \bullet np\backslash(np\backslash s) \Rightarrow (\Diamond\Box^{\downarrow} np)\backslash s}\,{\scriptstyle a}
  }\,{\scriptstyle t}
  \qquad
  \dfrac{}{cq/((\Diamond\Box^{\downarrow} np)\backslash s) \bullet ((\Diamond\Box^{\downarrow} np)\backslash s) \Rightarrow}
}{
  \dfrac{cq/((\Diamond\Box^{\downarrow} np)\backslash s) \bullet (np \bullet np\backslash(np\backslash s)) \Rightarrow cq}{\textsf{wie} \bullet (\textsf{Martin} \bullet \textsf{plaagte}) \Rightarrow cq}\,lex
}
$$

The postulate $\vec{K}\ 2l$ recursively allows a modally decorated type to adjoin to the left of any right branch. For example, starting with

$$\Diamond\Box^{\downarrow} A \bullet (B \bullet (C1 \bullet C2))$$

the modally-decorated subformula can move in one step to the left of the product $(C1 \bullet C2)$ and subsequently in a second step to the left of $C2$, as illustrated below:

$$\Diamond\Box^{\downarrow} A \bullet (B \bullet (C1 \bullet C2)) \Rightarrow B \bullet (\Diamond\Box^{\downarrow} A \bullet (C1 \bullet C2)) \Rightarrow B \bullet (C1 \bullet (\Diamond\Box^{\downarrow} A \bullet C2))$$

This correctly allows for

> Jan vroeg zich af wat (Marie (Piet zou geven)
> Jan ask refl particle what Marie Piet would give
> 'Jan wondered what Marie would give Pete'

While $\vec{K}\ 2l$ allows the modally-decorated type to look recursively down the left branch of a right branch, it is also possible in Dutch to find the gap down the left branch of a left branch:[7]

> Jan vroeg zich af ((op wie)(Marie (gestellt was)))
> Jan asked refl particle prep whom Marie like
> 'Jan wondered who Marie liked'

In this example, the extracted phrase must communicate with the position to the left of gestellt. This is accomplished by adding to the Dutch postulate package the interaction postulate $\vec{K}\ 1l$, formulated below:

$$\vec{K}\ 1l \quad \Diamond A \bullet (B \bullet C) \Rightarrow (\Diamond A \bullet B) \bullet C$$

[7] The example involves pied-piping with the preposition op; this fact is orthogonal to our interests here, so is not pursued here. For treatments of pied-piping, see Morrill [13].

Unlike $\overrightarrow{K}\ 2l$, the postulate $K2r$ is not recursive, since its output can never be matched to its input. Still, in English, the output of $K2r$ must be able to communicate with more deeply embedded positions, as in

Jean wondered (who (Maxima (tried (to (telephone)))))
Jean wondered (who (Maxima (persuaded (to (telephone Kim)))))

These examples are obtainable with the mirror images of the postulates for Dutch:

$$\begin{array}{ll} \overleftarrow{K}\ 1r & ((A \bullet B) \bullet \Diamond C) \Rightarrow ((A \bullet \Diamond C) \bullet B) \\ \overleftarrow{K}\ 2r & ((A \bullet B) \bullet \Diamond C) \Rightarrow (A \bullet (B \bullet \Diamond C)) \end{array}$$

We assume that these postulates hold for French as well as English. On this view then, the differences between French and English, on the one hand, and Dutch, on the other, reside in the choice between two sets of interaction postulates, displayed in Figures 1 and 2.

$$\begin{array}{ll} \overrightarrow{K}\ 2r & \Diamond_{wh} A \bullet (B \bullet C) \Rightarrow B \bullet (C \bullet \Diamond_{wh} A) \\ \overleftarrow{K}\ 1r & ((A \bullet B) \bullet \Diamond C) \Rightarrow ((A \bullet \Diamond C) \bullet B) \\ \overleftarrow{K}\ 2r & ((A \bullet B) \bullet \Diamond C) \Rightarrow (A \bullet (B \bullet \Diamond C)) \end{array}$$

Figure 1: postulates for French and English

$$\begin{array}{ll} \overrightarrow{K}\ 2l & \Diamond_{wh} A \bullet (B \bullet C) \Rightarrow B \bullet (\Diamond_{wh} A \bullet C) \\ \overrightarrow{K}\ 1l & \Diamond A \bullet (B \bullet C) \Rightarrow (\Diamond A \bullet B) \bullet C \end{array}$$

Figure 2: postulates for Dutch

The Dutch postulates allow an extracted phrase to occur directly following a complementizer. For example, consider the sentence Wie zei Marie dat die appel opgegeten heeft? Figure 3 displays the bracketing we assume and the succession of structures involved in a proof.[8]

On the other hand, the postulates proposed here for English and French do not allow extraction sites to follow a complementizer. More precisely, although it is possible for a modally-decorated expression to communicate with the position following a complementizer, this requires the expression to be on the right branch of a binary structure whose left branch is the complementizer, and this position makes it impossible for the expression to combine with the predicate.

[8] Full details of the proof depend on an analysis of extraposition, which we need not pursue here.

$$\frac{\dfrac{\text{((zei Marie) (dat (}\Diamond\Box^{\downarrow}np\text{ ((die appel) (opgegeten heeft)))))}}{\text{((zei Marie)(}\Diamond\Box^{\downarrow}np\text{ (dat ((die appel) (opgegeten heeft)))))}}\ \overrightarrow{K}\,2l}{\Diamond\Box^{\downarrow}np\text{((zei Marie)(dat ((die appel) (opgegeten heeft))))}}\ \overrightarrow{K}\,2l$$

$$\frac{\dfrac{\dfrac{\text{FAIL}}{\text{(Marie (said ((that }\Diamond\Box^{\downarrow}np\text{)(saw Martin))))}}\ \overleftarrow{K}\,1r}{\text{(Marie ((said (that (saw Martin))) }\Diamond\Box^{\downarrow}np\text{))}}\ \overleftarrow{K}\,2r}{\Diamond\Box^{\downarrow}np\text{(Marie (said (that (saw Martin))))}}\ \overrightarrow{K}\,2r$$

Discussion

The principles of distributivity on which the above account of extraction systems depends on are non-deterministic and dynamic. These properties distinguish this approach from alternatives in the literature and offer new perspectives on natural language extraction systems. The fuller report on this research in preparation will contain a comparison with current theoretical alternatives mentioned in the introduction.

References

[1] J.W. Bresnan. 1972. *Theory of Complementation in English Syntax.* Ph.D. dissertation. MIT.

[2] N. Chomsky & H. Lasnik. 1977. Filters and control. *Linguistic Inquiry* 8.425-504.

[3] G. Gentzen. 1934-5. Untersuchungen über das logische Schliessen, *Mathematische Zeitschrift* **39**, pp. 176-210, 405-431. (English translation in M. E. Szabo, ed., *The Collected Papers of Gerhard Gentzen*, North-Holland, Amsterdam).

[4] P. Hendriks. 1995. *Comparatives and Categorial Grammar.* Ph.D. dissertation. Rijksuniversiteit Groningen.

[5] S. C. Kleene. 1952. *Introduction to Metamathematics.* D. Van Nostrand, New York.

[6] A.S. Kroch. 1989. Asymmetries in Long-Distance Extraction. In M.R. Baltin & A.S. Kroch, eds., *Alternative Conceptions of Phrase Structure*, 66-98. Chicago and London: University of Chicago Press.

[7] J. Lambek. 1961. On the calculus of syntactic types. In R. O. Jakobson, ed., *Structure of Language in its Mathematical Aspects. Proceedings of the 12th Symposium in Applied Mathematics.* American Mathematical Society, Providence.

[8] J. Maling & A. Zaenen. 19xx. The non-universality of a surface filter. In J. Maling & A. Zaenen, eds., *Syntax and Semantics xx: Icelandic Syntax.* xxx-xxx. Orlando: Academic Press.

[9] M. Moortgat. 1996. Categorial type logics. In J. van Benthem and A. ter Meulen, eds., *Handbook of Logic and Language.* Elsevier, Amsterdam.

[10] M. Moortgat and R. T. Oehrle. 1993. *Categorial Grammar: Logical Parameters and Linguistic Variation.* Lecture notes. European Summer School in Logic, Language, and Information. Faculdade de Letras, Universidade de Lisboa, Portugal.

[11] M. Moortgat and R. T. Oehrle. 1993. Adjacency, dependency, order. P. Dekker & M. Stokhof, eds. *Proceedings of the Ninth Amsterdam Colloquium.* 447-467. Institute for Logic, Language, and Computation, Universiteit van Amsterdam.

[12] G. Morrill. 1994. *Type Logical Grammar: Categorial Logic of Signs.* Kluwer, Dordrecht.

[13] G. Morrill. 1995. Discontinuity in Categorial Grammar. *Linguistics & Philosophy* 18.175-219.

[14] R.T. Oehrle. 1997. Substructural logic and linguistic inference. Under review.

[15] D.M. Perlmutter. 1968. *Deep and Surface Constraints in Syntex.* Ph.D. dissertation. MIT.

[16] D.M. Perlmutter. 1971. *Deep and Surface Constraints in Syntax.* New York: Holt, Rinehart & Winston.

[17] J. Stoy. 1981. *Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory.* Cambridge, Mass. & London: MIT Press.