COMBINATORY CATEGORIAL GRAMMAR

Mark Steedman and Jason Baldridge

1 INTRODUCTION

Categorial Grammar (CG, Ajdukiewicz 1935; Bar-Hillel 1953) is one of the oldest lexicalized grammar formalisms, in which all grammatical constituents are distinguished by a syntactic type identifying them as either a function from arguments of one type to results of another, or as an argument. Such types, or *categories*, are transparently related to to the semantic type of the linguistic expression itself, differing mainly in the inclusion of information about language-specific linear order.

The earliest forms of CG were immediately recognized as being context-free and weakly equivalent to context-free phrase-structure grammars (CFPSG, Bar-Hillel, Gaifman and Shamir 1964). Soon after their elaboration by Bar-Hillel, Lambek (1958) cast CG as a logical calculus, which was also widely (and correctly) assumed to be context-free, although the actual proof—due to Pentus (1993)—was much harder to discover.[1]

The early evidence of weak equivalence to CFPSG led to a partial eclipse of CG in the 1960's. However, interest in CG on the part of syntacticians and computational linguists began to revive in the late 1970's and early 1980's. One reason for this revival came from contemporary developments in formalizing a type-driven semantics for natural language in the work of Richard Montague (1974) and his followers (see Partee 1976), which made the syntactic/semantic type-transparency of CG attractive. Another reason was the realization that transformational generative grammar was overly expressive (Peters and Ritchie 1973), leading to a search for more minimal extensions of context-free core grammars of various kinds (e.g. Gazdar 1981), including CG (e.g. Karlgren 1974, Landsbergen 1982).

Some early extensions to CG were "combinatory" in nature, extending the core CG with functional operations on adjacent categories, such as "wrap" (Bach 1979; Dowty 1979), functional composition (Ades and Steedman 1982), type-raising (Steedman 1985), and substitution (Szabolcsi 1989). These devel-

---

[1] See also Pentus 2003. The source of this difficulty is the essential use of an axiom schema in the definition of the Lambek calculus.

opments in turn led to a revival of interest in the non-combinatory type-logical alternative stemming from Lambek's work in the late 1950's, in which some but not all of these combinatory extensions emerged as theorems (see Oehrle, this volume).

The distinction between combinatory and type-logical approaches has remained fairly sharp since these early developments. On the one hand, Combinatory Categorial Grammar (CCG) of the kind presented in this chapter has retained an active concern with keeping expressive power and automata-theoretic complexity to a minimum, and has been actively involved with issues of linguistic explanation and practical computational linguistics, including wide-coverage parsing using statistical models. On the other hand, the Lambek tradition of type-logical grammars has been more concerned with theoretical issues and relations to logic and theorem-proving.

This chapter presents a formulation of CCG that goes some way toward reconciling this difference. While we retain the combinatory apparatus and low expressive power, we also incorporate the slash-typing characteristic of multi-modal type-logical grammar as the sole means of constraining derivation in CCG. This move allows the rules of the system to be stratified and selectively used in lexically specified contexts, thereby removing the need for the category-based restrictions on combinatory rules used for this purpose in previous formulations of CCG.

We begin by motivating CCG in terms of the current state of linguistic theory and then outline the modalized version of the formalism. CCG is then applied to the bounded constructions (binding, reflexivization, heavy NP shift, dative shift, raising, object and subject control, and passive). Next, we give analyses for the unbounded constructions (including extraction, scrambling, and coordination) in a number of languages, including English, Dutch, Japanese, Turkish, and Irish Gaelic. Finally, we briefly consider intonation structure and parentheticalization in English, and end with some remarks on implications for the theory of performance and computational applications.

2 THE CRISIS IN SYNTACTIC THEORY

The continuing need for volumes like the present one raises an obvious question: why are there so many theories of grammar around these days?[2] It is usual in science to react to the existence of multiple theories by devising a crucial experiment that will eliminate all but one of them. However, this tactic does not seem to be applicable to these proliferating syntactic theories. For one thing, in some respects they are all rather similar. Sometimes the similarities are disguised by the level of detail at which the grammar is presented—for example, Tree-Adjoining Grammar (TAG, Joshi 1988) and CCG can be regarded as precompiling into lexical categories some of the feature-unification that goes on during derivations in Lexical-Functional Grammar (LFG, Bresnan 1982), Head-driven Phrase Structure Grammar (HPSG, Pollard and Sag 1994) and other attribute-value grammars. Nevertheless, (thanks to Reinhart and Reuland 1991, 1993 and Pollard and Sag 1992, 1994, who clarified the descriptive account considerably), all of the theories under discussion including CCG and at least some varieties of Government-Binding or Principles and Parameters grammar (GB) have essentially the same binding theory, with a lexically defined domain of locality corresponding to the tensed clause, and a command or scope relation defined at some level representing predicate argument structure, such as logical form. The mechanisms involved, even when couched in terms of transformations like "NP movement," seem to be of rather low expressive power—essentially context-free (CF) and "base generable," to use Brame's (1978) term. Many phenomena involving dependencies bounded by the tensed verbal domain, such as raising, control, passivization, reflexivization, and the like, have this character. While some deep problems remain—in particular, the question of what the primitive components of linguistic categories themselves are—the theories are all in formal terms pretty much alike in their analysis of these constructions.

It is only when we consider the unbounded dependencies that cross the bounds of the tensed clause in constructions such as the relative clause, various kinds of "reduced" or "gapped" coordinate structures, and other "stylistic" constructions, including intonation structure and parentheticalization that the

---

[2] Besides those discussed in this volume, others in active use include Tree-Adjoining Grammar (TAG, Joshi 1988 and Government-Binding theory itself (GB, a.k.a. Principles and Parameters, the Minimalist Program, etc., Chomsky 1981, 1995).

theories differ in important ways. However, in most cases, the apparatus that is added to the CF core is sufficiently powerful and expressive that it is impossible to falsify or to distinguish any of the alternatives on grounds of expressive power. “*Wh*-movement” or equivalent coindexing of traces in GB, “functional uncertainty” or the ability to define dependencies in terms of paths defined as regular expressions in LFG, set-valued “SLASH features” in HPSG, certain classes of structural rules in Type Logical Grammar (TLG, Oehrle, this volume), are all examples of powerful mechanisms of this kind. It is a measure of their expressive power that they have to be attended by seemingly arbitrary constraints on their operation which are strikingly similar to one or another of the constraints that limited the classical transformational rules that are nowadays called MOVE and DELETE, such as the Coordinate Structure Constraint (Ross 1967) and the Fixed Subject Condition or “**that*-trace” filter, first identified by Bresnan (1972).

Constraints on rules are not necessarily in themselves a sign of anything wrong with a theory of grammar. They can arise from all kinds of extragrammatical sources, such as the requirements of semantics, the parser, or the language learner. (Island constraints like the Complex Noun Phrase Constraint of Ross 1967 provide an example of a group of constraints that should probably be explained in terms of probabilistically or semantically guided parsing rather than in terms of grammar as such.)

However, when a constraint is observed to hold cross-linguistically, as in the case of certain restrictions discussed below which relate coordinate constructions to primary word-order, that fact calls for *some* kind of explanation. One way to provide that explanation is to show that the constraints stem from limitations in automata-theoretic power of the grammar itself. A theory that is incapable in the first place of expressing grammars for languages that violate the condition provides a very convincing explanation for why they hold. Such a theory of grammar may also bring beneficial complexity and learnability results (although such theoretical results do not necessarily tell us much about the actual difficulty of practical processing and language learning for realistic grammars).

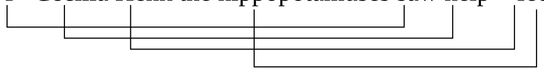
The project of explaining constraints on observed grammars as arising in part from grammar formalisms of low expressive power was the impulse behind Generalized Phrase Structure Grammar (GPSG, Gazdar 1981; Gazdar et al. 1985), which tried to capture as much as possible within a strictly context-free formalism. While it was clear from the start that phenomena existed that were unlikely to be capturable in this way, the effects of seeing just how many linguistic generalizations *could* be captured in context-free terms, supporting a fully compositional semantics, was extremely salutary. Most of all, it focused attention on multiple long range dependencies, since these required generalization of the mediating SLASH feature to be either a stack- or set- valued feature. In particular the fact that multiple dependencies in English show a tendency to nest rather than cross, as evidenced by the following minimal pair, suggested that SLASH features should be stacks.

(1) a. a violin which$_i$ [this sonata]$_j$ is hard to play$_j$ upon$_i$
b. *a sonata which$_i$ [this violin]$_j$ is hard to play$_i$ upon$_j$

The two dependencies in (1a) must nest, rather than intercalate, as they would have to for (1b) to have a meaning to do with playing sonatas on violins (the asterisk here means “not allowed with the intended reading”).

However, the tendency to nest multiple dependencies is by no means universal. In certain Dutch constructions, multiple dependencies obligatorily intercalate (Huybregts 1976, 1984; Shieber 1985), as in the following example:[3]

(2) ... omdat ik Cecilia Henk de nijlpaarden zag helpen voeren.
... because I Cecilia Henk the hippopotamuses saw help feed



‘... because I saw Cecilia help Henk feed the hippopotamuses.’

GPSG itself does not seem to have been particularly amenable to any restricted kind of generalization (although such a generalization is implicit in Pollard 1984 and Gazdar 1988), and constraining automata-theoretic power ceased to be a major focus of concern during its evolution into HPSG. However, a number of other formalisms, including TAG and CCG, continued to explore the possibility of capturing human grammars using low-power formalisms. In particular, Ades and Steedman (1982:522) suggested that the *same* stack might be implicated both in the push-down automaton (PDA) character-

---

[3] The indicated dependencies are those between semantically related arguments and predicates, rather than surface dependencies between verbs and NP arguments that would be attributed on a VP analysis of the construction. However, in either case the Dutch dependencies cross.

istic of context-free grammar and in mediating multiple unbounded dependencies. Vijay-Shanker and Weir (1990, 1993, 1994) subsequently showed that all three formalisms were weakly equivalent to Linear Indexed Grammar, and delineated a new level in the Chomsky Hierarchy characterized by a generalization of the PDA, called an Extended Push Down Automaton (EPDA), which utilized a single stack of stack-valued features. Subsequent explorations with the TAG and CCG frameworks suggest that this level may be the lowest at which all syntactic phenomena of natural grammar can be captured.[4]

Such a theory offers the possibility of reducing the operations MOVE and DELETE to what is sometimes called MERGE—that is, the simple combination of adjacent constituents.

To do this we must begin by standing traditional generative syntax on its head.

3 COMBINATORY CATEGORIAL GRAMMAR

Combinatory Categorial Grammar (CCG), like other varieties of categorial grammar discussed by Oehrle, this volume, is a form of lexicalized grammar in which the application of syntactic rules is entirely conditioned on the syntactic type, or *category*, of their inputs. No rule is structure- or derivation-dependent.

Categories identify constituents as either *primitive categories* or *functions*. Primitive categories, such as N, NP, PP, S, and so on, may be regarded as further distinguished by features, such as number, case, inflection, and the like. Functions (such as verbs) bear categories identifying the type of their result (such as VP) and that of their argument(s)/complements(s) (both may themselves be either functions or primitive categories). Function categories also define the order(s) in which the arguments must combine, and whether they must occur to the right or the left of the functor. Each syntactic category is associated with a logical form whose semantic type is entirely determined by the syntactic category, under a principle of “Categorial Type Transparency” (Steedman 2000b, (hereafter, *SP*).

Pure CG (Ajdukiewicz 1935, Bar-Hillel 1953) limits syntactic combination to rules of functional *application* of functions to arguments to the right or left.

---

[4] This conjecture has been challenged by Rambow (1994) and subsequently defended by Joshi, Rambow and Becker (2000).

This restriction limits expressivity to the level of context-free grammar, and CCG generalizes the context-free core by introducing further rules for combining categories. Because of their strictly type-driven character and their semantic correspondence to the simplest of the combinators identified by Curry and Feys (1958), these rules are called *combinatory* rules and are the distinctive ingredient of CCG, giving it its name. They are strictly limited to certain directionally specialized instantiations of a very few basic operations, of which the most important are *type-raising* and functional *composition*.[5]

Though early work in CCG focused primarily on phenomena in English and Dutch, grammar fragments capturing significant cross-linguistic generalizations have been constructed more recently in the framework (e.g., Turkish, Hoffman 1995; Japanese, Komagata 1999; Tzotzil, Trechsel 2000; Tagalag and Toba Batak, Baldridge 2002; Haida, Enrico and Baldridge ????). In this chapter, we present basic aspects of analyses of English, Dutch, Japanese and Turkish, with a particular focus on a generalization for free word-order that leaves expressive power at the same low level in the spectrum of "mildly context-sensitive" grammars (Joshi 1988) as standard CCG. Finally, the problem of parsing in the face of so-called spurious ambiguity is not only easily solvable with standard parsing methodologies, yielding processors which are of polynomial worst-case complexity and practicable average case complexity, as well as compatible with state-of-the-art probabilistic optimization (Hockenmaier and Steedman 2002b; Hockenmaier 2003a; Clark and Curran 2004), but also directly compatible under the most restrictive assumptions possible with what is known about human sentence processing, as discussed by Tanenhaus (this volume).

3.1 Categorial Grammar

In CCG, as in other varieties of Categorial Grammar reviewed by Wood (1993) and exemplified in the bibliography below, syntactic information of the kind that can be captured for English in familiar context-free production rules like (3) is transferred to lexical entries like (4):

---

[5] A third class of combinatory rules related to *Substitution*, Curry and Feys' **S** combinator, are ignored here.

(3) $S \longrightarrow NP \quad VP$
$VP \longrightarrow TV \quad NP$
$TV \longrightarrow \{\mathit{proved},\ \mathit{finds},\ \ldots\}$

(4) proved := $(S\backslash NP)/NP$

This syntactic "category" identifies the transitive verb as a function, and specifies the type and directionality of its arguments and the type of its result. We here use the "result leftmost" notation in which a rightward-combining functor over a domain β into a range α are written α/β, while the corresponding leftward-combining functor is written $\alpha\backslash\beta$, where α and β may themselves be function categories.[6] As in any other theory of grammar, we must assume that the ensemble of such syntactic category types that can co-exist in the lexicon of any human language is subject to universal constraints related to learnability, of a kind investigated for CCG by McConville (2006, 2007) using default inheritance in a hierarchical feature system.

We follow Jacobson (1990, 1992a), Hepple (1990) and Baldridge (2002); Baldridge and Kruijff (2003) (and depart from *SP*) in assuming that rules and function categories are "modalized" using feature-values, as indicated by a subscript on slashes. Specifically, we assume that function categories may be restricted as to the rules that allow them to combine with other categories, via slashes typed with four feature values: $\star$, $\times$, $\diamond$, and $\cdot$. The effect of each of these slash-types will be explicated as we introduce each of the combinatory rules and define their interaction with the lexical slash-types. The basic intent is as follows: the $\star$ lexical type is the most restricted and allows only the most general applicative rules; $\diamond$ permits order-preserving associativity in derivations; $\times$ allows limited permutation; and $\cdot$ is the most permissive lexical type, allowing all rules to apply. The relation of these types to each other can be compactly represented via the hierarchy given in Figure 1.[7]

The effect of the slash-types is to permit lexical control over CCG's combinatory rules by defining the ability of functional categories to serve as input to only a subset of the available rules. Without typed slashes, language-specific restrictions or even bans on some combinatory rules are necessary in order to

---

[6] There is an alternative "result on top" notation due to Lambek (1958), according to which the latter category is written $\beta\backslash\alpha$.

[7] The use of a hierarchy such as this as a formal device is optional, and instead could be replaced by multiple declarations of the combinatory rules.

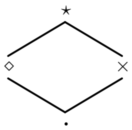


Figure 1: CCG type hierarchy for slash features (adapted from Baldridge and Kruijff 2003).

block certain ungrammatical word orders. With them, the combinatory rules are truly universal: the grammar of every language utilizes exactly the same set of rules, without modification, thereby leaving all cross-linguistic variation in the lexicon. As such, CCG is a *fully* lexicalized grammar formalism. See Baldridge (2002), Baldridge and Kruijff (2003), and Hoyt and Baldridge (2008) for further discussion of the implications of the slash-typing formulation of CCG.[8]

The most freely-combining types of slash $/_{\cdot}$ and $\backslash_{\cdot}$ allow a category to combine by any combinatory rule. The slashes in (4) are of this type. It will be convenient to abbreviate this type as a plain forward or backward slash, continuing to write such categories as before.

In order to allow functors such as (4) to combine with their arguments, we need combinatory rules, of which the two simplest are the following functional application rules:

(5) *The functional application rules*
  a. $X/_{\star}Y \quad Y \quad \Rightarrow \quad X$ $(>)$
  b. $Y \quad X\backslash_{\star}Y \quad \Rightarrow \quad X$ $(<)$

Because $\star$ is the supertype of all other slash-types, the $/_{\star}$ and $\backslash_{\star}$ slashes on these rules mean that *all* categories can combine by these rules.[9]

---

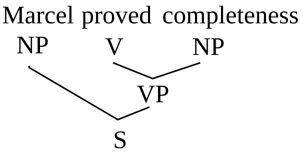
[8] The fact that restrictions are not required under this formulation of CCG answers a common and long-standing criticism of the theory from researchers in the TLG community. However, there is an obvious duality between restricting rules as to the categories that they may apply to, and restricting the categories themselves by distinguishing different slash-types—see Baldridge and Kruijff (2003) for an embedding of the modal formulation of CCG within a version of CCG which permits rule restrictions. Furthermore, while it is possible to define a TLG system that acts on the slash-types and categories described here (see Baldridge 2002), we do not here assume that typed slashes are true implicational operators as they are in TLG.

[9] This accords with the fact that, in TLG, under the residuation laws, all modalities have access to the base logic (in which the elimination rules correspond to CCG's application rules). Note, however, that it would be entirely possible to devise modal settings in multi-modal CCG in which

These rules have the form of very general binary phrase structure rule schemata. In fact, "pure" categorial grammar limited to these these two rules alone is essentially context-free grammar written in the accepting, rather than the producing, direction, with a consequent transfer of the major burden of specifying particular grammars from the PS rules to the lexicon. While it is now convenient to write derivations as in (6a), they are equivalent to conventional phrase structure derivations (6b):

(6) a.

$$\begin{array}{ccc}
\text{Marcel} & \text{proved} & \text{completeness}\\
\overline{NP} & \overline{(S\backslash NP)/NP} & \overline{NP}\\
 & \multicolumn{2}{c}{\overline{\qquad\quad S\backslash NP \qquad\quad}{}^{>}}\\
\multicolumn{3}{c}{\overline{\qquad\qquad\qquad S \qquad\qquad\qquad}{}^{<}}
\end{array}$$

b. 

It is important to note that such tree-structures are simply a representation of the process of derivation. They are not structures that need to be built by a processor, nor do they provide the input to any rules of grammar.

Despite this close correspondence, the categories labeling the nodes of the derivation in (6a) are much more informative than the atomic symbols in the tree (6b). Subcategorization is directly encoded in functor categories rather than implicitly in syntactic productions or through the use of preterminal symbols such as $V_{intrans}$, V_{trans} and $V_{ditrans}$. Furthermore, there is a systematic correspondence between notions such as *intransitive* and *transitive* — after the transitive category $(S\backslash NP)/NP$ consumes its object argument, the resulting category $S\backslash NP$ is exactly that of an intransitive verb. This is a result of the way lexical categories are defined in combination with the universal rules of functional application.[10]

Categories can be regarded as encoding the semantic type of their translation. This translation can be made explicit in the following expanded notation, which associates a logical form with the entire syntactic category, via the colon operator, which is assumed to have lower precedence than the categorial slash operators. (Agreement features are also included in the syntactic category, represented as subscripts, much as in Bach 1983. The feature 3*s* is "underspecified" for gender and can combine with the more specified *3sm* by a standard

---

some categories can be used with composition rules but not with application, as in Jacobson's analysis of raising (1992b).

[10] See Oehrle, this volume, for a deductive explanation within the Lambek framework of the relationship between categories and phrase structure labels.

unification mechanism that we will pass over here—see Shieber 1986.)[11]

(7) proved := $(S\backslash NP_{3s})/NP : \lambda x\lambda y.prove'xy$

We must also expand the rules of functional application in the same way:

(8) *Functional application*

a. $X/_{\star}Y : f \quad Y : a \quad \Rightarrow \quad X : fa \qquad (>)$

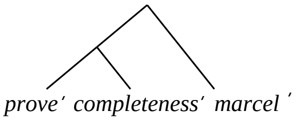
b. $Y : a \quad X\backslash_{\star}Y : f \quad \Rightarrow \quad X : fa \qquad (<)$

All such combinatory rules are subject to a similar transparency condition to the Principle of Categorial Type-Transparency, called the Principle of Combinatory Type-Transparency (*SP*), which says that the semantic type of the reduction is the same as its syntactic type, here functional application. They yield derivations like the following:

(9)
$$\begin{array}{cccl}
\text{Marcel} & \text{proved} & \text{completeness} & \\
\hline
NP_{3sm} : marcel' & (S\backslash NP_{3s})/NP : \lambda x\lambda y.prove'xy & NP : completeness' & \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash NP_{3s} : \lambda y.prove'completeness'y} & > \\
\cline{1-3}
\multicolumn{3}{c}{S : prove'completeness'marcel'} & <
\end{array}$$

The derivation yields the category S with a compositional interpretation, equivalent under a convention of left associativity to (10a):

(10) a. $(prove'completeness')marcel'$ b.



Thus, the traditional subject-predicate structure reflecting c-command relations exhibited in (10b) is expressed at the level of propositional logical form or LF-structure.

---

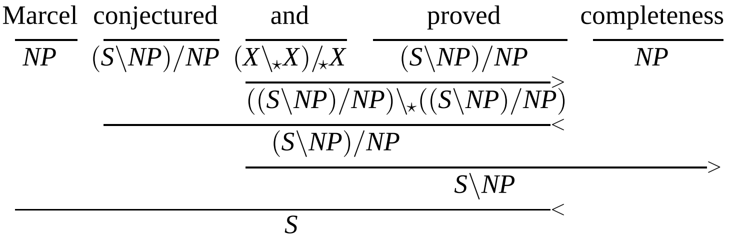
[11] It is possible to bind arguments in semantic representations using mechanisms other than those of the λ-calculus. For example, Steedman (1990), Zeevat (1988) and Hoffman (1995) employ unification for this purpose. The use of the λ-calculus as the representation framework is also optional since interpretations can instead be encoded with other representation languages such as Indexed Languages (Zeevat 1988), Hybrid Logic Dependency Semantics (Kruijff 2001) or Minimal Recursion Semantics (Copestake, Lascarides and Flickinger 2001). See Baldridge and Kruijff (2002) for an approach which integrates CCG with Hybrid Logic Dependency Semantics, and Villavicencio (2002) for one which uses Minimal Recursion Semantics within the context of Unification-Based Generalized Categorial Grammar.

3.2 Coordination

Coordination is captured in the present version of CCG via the following category schema for conjunctions like *and*, allowing constituents of like type to conjoin to yield a single constituent of the same type:[12]

(11) *The Conjunction Category*
and := $(X\backslash_{\star}X)/_{\star}X$

The $\star$ feature on the slashes of this category restrict it to combine *only* by the application rules (5). It gives rise to derivations like the following:

(12)


3.3 Composition

In order to allow coordination of contiguous strings that are not standardly assumed to constitute constituents, CCG allows certain further operations on functions related to Curry's combinators (Curry and Feys 1958). For example, functions may *compose*, as well as apply, under the following rules:[13]

(13) *The harmonic functional composition rules*

a. $X/_{\diamond}Y : f \quad Y/_{\diamond}Z : g \quad \Rightarrow \quad X/_{\diamond}Z : \lambda z.f(gz)$ $(>\mathbf{B})$

b. $Y\backslash_{\diamond}Z : g \quad X\backslash_{\diamond}Y : f \quad \Rightarrow \quad X\backslash_{\diamond}Z : \lambda z.f(gz)$ $(<\mathbf{B})$

The operation of these rules in derivations is indicated by an underline indexed $>\mathbf{B}$ or $<\mathbf{B}$ respectively (because Curry called his composition combinator $\mathbf{B}$). The $\diamond$ slash-type means that only categories bearing that type or the most general $\cdot$ type (here abbreviated as plain slash) may compose. Categories bearing the incompatible $\times$ type or the least general $\star$ type (such as the

---

[12] The semantics of this category, or rather category schema, is somewhat complex, and is omitted here.

[13] Combinatory rules like functional composition resemble a highly restricted (because they are type-driven rather than structure-dependent) class of "generalized" or "double-based" transformations of the kind proposed in Chomsky 1957.

conjunction category (11)) cannot combine by these rules.

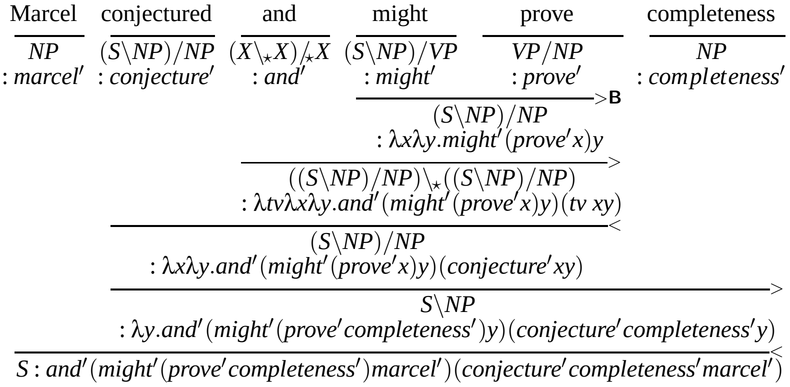
Without the use of the hierarchy given in Figure 1 relating the various types, the forward composition rule would be stated with the following four instantiations (the semantics for which is as in (13)):

(14) a. $X/_{\diamond}Y \quad Y/_{\diamond}Z \quad \Rightarrow \quad X/_{\diamond}Z$
b. $X/_{\diamond}Y \quad Y/_{\cdot}Z \quad \Rightarrow \quad X/_{\cdot}Z$
c. $X/_{\cdot}Y \quad Y/_{\diamond}Z \quad \Rightarrow \quad X/_{\diamond}Z$
d. $X/_{\cdot}Y \quad Y/_{\cdot}Z \quad \Rightarrow \quad X/_{\cdot}Z$

We explain why only these four mixtures are utilized for $>\mathbf{B}$ in section 4.

The effect of (13a) can be seen in the derivation of sentences like (15), which crucially involves the composition of two verbs to yield a composite of the same category as a transitive verb. It is important to observe that composition also yields an appropriate interpretation for the composite verb *might prove*, as $\lambda x \lambda y.might'(prove' x)y$, an object which if applied to an object *completeness* and a subject *Marcel* yields the proposition $might(prove' completeness')marcel'$. The coordination will therefore yield an appropriate semantic interpretation.[14]

(15)



CCG generalizes composition to $\mathbf{B}^n$ for small n—e.g.

(16) $X/_{\diamond}Y : f \quad (Y/_{\diamond}W)/_{\diamond}Z : g \quad \Rightarrow \quad (X/_{\diamond}W)/_{\diamond}Z : \lambda z \lambda w.f((gz)w) \qquad (>\mathbf{B}^2)$

Among other consequences, this generalization permits modal verbs to compose into ditransitive verbs, as in the following:

---

[14] The analysis begs some syntactic and semantic questions about the coordination. See *SSI* for a more complete account.

(17)
$$\begin{array}{cc}
\text{might} & \text{give} \\
\hline
(S\backslash NP)/VP & (VP/NP)/NP \\
\hline
\multicolumn{2}{c}{((S\backslash NP)/NP)/NP \quad {>}\mathbf{B}^2}
\end{array}$$

CCG includes a further related family of binary combinatory rules first proposed by Szabolcsi 1989, 1987, based on the combinator **S**, which Steedman 1987 called rules of *substitution*. These rules are not discussed here, except to note that they are subject to a similar generalization, suggesting the following generalization about allowable binary rules in CCG:[15]

(18) Binary rules in CCG are those whose semantics corresponds to the application to the principal functor $X|Y$ of a combinatory term of bounded size made up of the unary combinators **B** and **S**, plus application of the result to the subordinate functor $W|Z$.

3.4 Type-Raising

Combinatory grammars also include type-raising rules, which turn arguments into functions over functions-over-such-arguments. These rules allow arguments to compose with the verbs that seek them, and thereby take part in coordinations as in (20).

(19) *Forward type-raising* (>**T**)
$X : a \quad \Rightarrow \quad \mathrm{T}/_i(\mathrm{T}\backslash_i X) : \lambda f.fa$

The subscript i on the slashes means that they both have the same type as what ever function $\mathrm{T}\backslash_i X$ the raised category is applied to. T is a metavariable over categories. If instantiated as S, it allows the following derivation:

(20)
$$\begin{array}{cccccc}
\text{Marcel} & \text{proved} & \text{and} & \text{I} & \text{disproved} & \text{completeness} \\
\hline
NP & (S\backslash NP)/NP & (X\backslash_\star X)/_\star X & NP & (S\backslash NP)/NP & NP \\
\cline{1-1}\cline{4-4}
S/(S\backslash NP)\;{>}\mathbf{T} & & & S/(S\backslash NP)\;{>}\mathbf{T} & & \\
\cline{1-2}\cline{4-5}
\multicolumn{2}{c}{S/NP\;{>}\mathbf{B}} & & \multicolumn{2}{c}{S/NP\;{>}\mathbf{B}} & \\
\cline{3-5}
& & \multicolumn{3}{c}{(S/NP)\backslash_\star(S/NP)\;{>}} & \\
\cline{2-5}
& \multicolumn{4}{c}{S/NP\;{<}} & \\
\cline{3-6}
& & \multicolumn{4}{c}{S\;{>}}
\end{array}$$

---

[15] For example, the basic composition rules (13) and (26) are unary **B** plus application, rule (16) is **BBB** plus application, and so on. We are grateful to Fred Hoyt for discussions on this question.

The variable X in type-raising is restricted to primitive argument categories, NP, PP etc., and to primitive functors like verbs. It therefore resembles the traditional notion of *case*—in this case, the nominative. Unlike the other combinatory rules, it can be regarded as a lexical or morphological-level process, although for an almost caseless language like English it is often convenient to include it in the derivation, as above, via a unary rule, and in fact this is how it is implemented in parsers like Hockenmaier and Steedman 2002b and Clark and Curran 2004. We shall see later that English includes further type-raising categories corresponding to all the other traditional cases.[16]

4 THE COMBINATORY PROJECTION PRINCIPLE

We have given examples of several rules that encode the syntactic reflex of a few basic semantic functions (combinators). However, a larger set of possible rules could be derived from the combinators. CCG restricts the set to be only those which, in addition to the aforementioned Principle of Combinatory Type-Transparency, obey the following further principles:

(21) *The Principle of Adjacency:*
Combinatory rules may only apply to finitely many phonologically realized and string-adjacent entities.

(22) *The Principle of Consistency:*
All syntactic combinatory rules must be consistent with the directionality of the principal function.

(23) *The Principle of Inheritance:*
If the category that results from the application of a combinatory rule is a function category, then the slash type of a given argument in that category will be the same as the one(s) of the corresponding argument(s) in the input function(s).

The first of these principles is merely the definition of combinators themselves. The other principles say that combinatory rules may not override, but must rather "project," the directionality specified in the lexicon. More concretely,

---

[16] To the extent that both type-raising and case associate entities with roles in actions, they are both akin to the psychologists' notion of the "affordance" of a percept, as that word is used by Gibson (1966) and his followers—see Steedman 2002.

the Principle of Consistency excludes the following kind of rule:

(24) $X\backslash_{\star}Y \quad Y \quad \Rightarrow \quad X$ (disallowed)

The Principle of Inheritance excludes rules like the following hypothetical instances of composition:

(25) a. $X/_{\diamond}Y \quad Y/_{\diamond}Z \quad \Rightarrow \quad X\backslash_{\diamond}Z$ (disallowed)

b. $X/_{\diamond}Y \quad Y/_{\diamond}Z \quad \Rightarrow \quad X/_{\times}Z$ (disallowed)

On the other hand, these principles do allow rules such as the following, along with generalization along the lines of (16):

(26) *The crossing functional composition rules*

a. $X/_{\times}Y : f \quad Y\backslash_{\times}Z : g \quad \Rightarrow \quad X\backslash_{\times}Z : \lambda z.f(gz)$ $(>\mathbf{B}_{\times})$

b. $Y/_{\times}Z : g \quad X\backslash_{\times}Y : f \quad \Rightarrow \quad X/_{\times}Z : \lambda z.f(gz)$ $(<\mathbf{B}_{\times})$

Such rules are not theorems of type calculi such as that of Lambek (1958) and its descendants, and in fact cause collapse of such calculi into permutation completeness if added as axioms (Moortgat 1988), a fact that has motivated the development of multi-modal varieties of categorial grammar within the type-logical tradition by Hepple (1990) and Morrill 1994, cf. Oehrle, this volume. While such rules do not cause a collapse in CCG *even without the modalities*, the present use of slash-types to provide finer control over the rules is directly inspired by multi-modal categorial grammar (see Baldridge 2002).

5 THE BOUNDED CONSTRUCTIONS

The treatment in CCG of the bounded constructions traditionally known as Reflexivization, Dative-shift, Raising, Object- and Subject- Control, and Passivization is equivalent in essence to the treatment of these phenomena in G/HPSG and LFG (and is unlike the treatment in GB, TAG, and the type-logical varieties of CG discussed by Oehrle in the present volume) in that it expresses the underlying dependencies and structures at the level of logical form, rather than at the level of derivation. The logical forms in question as they are presented here are extremely simplified and leave many semantic subtleties to be specified in meaning postulates that we do not specify.

5.1 Binding Theory

We define a structural notion of LF-command over logical forms like the one built in derivation (9), along lines set out more fully in Steedman 1996, (hereafter, *SSI*).

(27) *LF-command*: a node α in a logical form Λ LF-commands a node β in Λ if the node immediately dominating α dominates β and α does not dominate β.

(The relation "dominates" is defined as the recursive transitive closure of the parent relation. The relation "immediately dominates" is defined as holding between the first branching node that dominates a node and that node.)

If the LF interpretations of bound pronouns and reflexive/reciprocal anaphors bound in a logical form Λ are defined as (non-branching) "pro-terms" of the form $pro'x$, $ana'x$, in which x is identical to some other node in Λ, then a binding theory much like that of Chomsky (1981) can be defined.[17] For example, Condition C of the binding theory, which rules out *He*$_i$ *likes John*$_i$, *He*$_i$ *thinks Mary likes John*$_i$, etc., can be defined as follows:

(28) *Condition C*: No node except the argument in a pro-term may be LF-commanded by itself.

We shall see directly that Condition A of the binding theory follows immediately from Condition C and the assumption that reflexivization is lexicalized. Condition B, which says that pronouns must not be bound in their local tensed domain) is claimed in *SSI* to arise because pronominal anaphora is *not* lexicalized, but mediated by contextual update.

5.2 Reflexivization

Condition A says that reflexives etc. must be bound in their local tensed domain, excluding **Himself*$_i$ *likes Harry*$_i$, **Harry*$_i$ *thinks Sally likes himself*$_i$, and the like. This is naturally captured in CCG if we follow Reinhart and Reuland (1991, 1993) in assuming that reflexivization is lexicalized. (We also follow those authors and Pollard and Sag 1992 in assuming that English reflexives have logophoric homonyms that are pronominal and not subject to Condition

---

[17] For two nodes to be identical, one must be a pointer to the other. Mere equality of content is not identity.

A). That is to say that the pronoun *itself*, like all noun phrases, is type-raised. Unlike most arguments, it is a clitic, like French *se*, which means that it is specialized to apply only to *lexical* verbal categories. The natural way to capture this specialization is to define it as a lexicon-internal morphological operator.

Its category is as follows:[18]

(29) -itself := $(S\backslash NP_{3sn})\backslash_{LEX}((S\backslash NP_{3sn})/NP) : \lambda p\lambda y.p(ana'y)y$

It gives rise to derivations like the following

(30)
$$\begin{array}{lll}
\text{The fixed}-\text{point theorem} & \text{proved} & -\text{itself} \\
\hline
S/(S\backslash NP_{3sn}) : & (S\backslash NP_{agr})/NP : & (S\backslash NP_{3sn})\backslash_{LEX}((S\backslash NP_{3sn})/NP) : \\
fptheorem' & \lambda x\lambda y.prove'xy & \lambda p\lambda y.p(ana'y)y \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash NP_{3sn} : \qquad <} \\
 & \multicolumn{2}{c}{\lambda y.prove'(ana'y)y} \\
\hline
\multicolumn{3}{c}{S : prove'(ana'fptheorem')fptheorem' \qquad >}
\end{array}$$

The logical form yielded by above the example conforms to Condition C (28). However, it should be observed that a category parallel to (29) that would license **Itself proved the fixed-point theorem* would be in violation of Condition C (28), and cannot therefore exist in any language. Moreover the binding captured in (10) is by definition limited to the domain of a lexical verb. Condition A is therefore captured without further stipulation.

5.3 Heavy Noun Phrase Shift

The availability of backward crossed composition (26b) allows us to account for the fact that most adjuncts and second arguments can invert order with the first argument of the verb, via derivations like the following:

---

[18] This category can be thought of as suggestive of a more involved strategy using the unary modalities of TLG. For example, we could assume, similar to Morrill (1988) and Hepple (1990), that all categories from the lexicon are encapsulated in unary modalities. Thus, a transitive verb from the lexicon would appear as $\Box^{\downarrow}_{lex}\Diamond_{lex}((S\backslash NP_{agr})/NP)$, and the category for *itself* would then be defined as $\Box^{\downarrow}_{lex}\Diamond_{lex}((S\backslash NP_{3sn})\backslash(\Box^{\downarrow}_{lex}\Diamond_{lex}((S\backslash NP_{3sn})/NP)))$ instead of (29). For a derivation without the reflexive, the unary modalities on the transitive category can be dropped (via TLG's residuation laws for unary modalities, which in CCG would be enacted using the unary rule $\Box^{\downarrow}_{i}\Diamond_{i}X \Rightarrow X$) to allow the category to be used in the usual manner. Using unary modalities in this manner would ensure that the category of *thinks that Mary likes* would be $(S\backslash NP_{agr})/NP$ (lacking lexical unary modalities) and thus not be an acceptable argument for the category given above for *itself*, thereby blocking ungrammatical sentences such as **The dog$_i$ thinks that Mary likes itself$_i$*.

(31)

$$\begin{array}{llll}
\text{I} & \text{introduced} & \text{to Marcel} & \text{my very heavy friends} \\
\hline
S/(S\backslash NP): & ((S\backslash NP)/PP_{TO})/NP: & S\backslash(S/PP_{TO}): & S\backslash(S/NP): \\
\lambda p.p\ me' & \lambda x\lambda y\lambda z.introduce'yxz & \lambda q.q\ marcel' & \lambda r.r\ friends' \\
\hline
\multicolumn{2}{c}{(S/PP_{TO})/NP: \quad >\mathbf{B}^2} & & \\
\multicolumn{2}{c}{\lambda x\lambda y.introduce'yx\ me'} & & \\
\hline
\multicolumn{3}{c}{S/NP : \lambda x.introduce'marcel'x\ me' \quad <\mathbf{B}_{\times}} & \\
\hline
\multicolumn{4}{c}{S : introduce'marcel'friends'me' \quad <}
\end{array}$$

Such derivations preserve the binding condition C at the level of logical form as required by the following:

(32) I introduced to each other some very heavy friends.

The crucial involvement of the type-raised PP category also predicts that preposition stranding will be incompatible with the HNPS construction, as in (a), despite the possibility of stranding *in situ* propositions as in (b), which involves the unraised PP/NP (traces are included to indicate intended readings):

(33) a. *Who did you introduce to *t* your very heavy friends?
b. Who did you introduce your very heavy friends to *t*?

A derivation similar to (31), suggested as an exercise, allows adjuncts of type $S\backslash S$, $VP\backslash VP$, etc. to take part in the construction, as in the following example:

(34) I shall meet tomorrow some very heavy friends from Hoboken.

5.4 Dative Shift

The ditransitive category for a verb like *give* is as follows:[19]

(35) gave := $((S\backslash NP)/NP)/_{\diamond}NP : \lambda x\lambda y\lambda z.give'yxz$

The $\diamond$ type of the first slash is incompatible with combination via the backward crossed composition rule (26b). Ditransitives therefore cannot undergo HNPS, ruling out sentences like the following, despite the fact that the type-raised accusative would otherwise permit a derivation analogous to (31):

(36) *I gave a book my very heavy friend from Hoboken.

---

[19] For reasons discussed by Oehrle (1975), we should not assume that the predicate *give′* is identical to that of the verb in *I gave the flowers to Marcel.* In fact, the binding facts force the assumption that the underlying predicate in *Marcel showed me/*myself to *me/myself* reverses the command relations between object and showee.

The category is otherwise unremarkable, apart from the fact that the command relation of the two rightward arguments is *reversed* between the surface derivation (which is suggested as an exercise) and the logical form that results from category (35), in which x LF-commands y. This property (which, as we shall see in section 6.2, is universal in SVOX, VSOX, XOSV, and XOVS constructions) captures the following binding assymetry between indirect and direct object for ditransitive verbs, via Condition C (28):

(37) a. Marcel showed me myself$_{S:show'(ana'me')me'marcel'}$

b. *Marcel showed myself me$_{S:show'me'(ana'me')marcel'}$

The type-logical varieties of CG discussed by Oehrle in this volume typically eschew such "intrinsic" use of logical form (and indeed the entire account of binding offered here). Instead, such grammars typically reverse the order of indirect and direct object in the syntactic category of ditransitives, so that accusative as first argument commands dative as second, and include the WRAP operations first introduced by Bach (1979) and Dowty (1979), together with corresponding slash-modalities (Jacobson 1992a), to recapture the now-inconsistent English word-order.

Such an alternative is not without appeal. However, it greatly complicates the grammar in other respects, especially as concerns the account of coordination presented in section 6.2 below.

5.5 Raising

Raising verbs like *seem* have categories like the following:

(38) seems := $(S\backslash NP)/(S_{TO}\backslash NP) : \lambda p\lambda y.seem'(py)$

The primitive $seem'$ is a modal or intensional operator which the interpretation composes with the complement predicate, thus:

(39)
$$\begin{array}{cccc}
\text{Marcel} & \text{seems} & \text{to} & \text{drink} \\
\hline
S/(S\backslash NP) & (S\backslash NP)/(S_{TO}\backslash NP) & (S_{TO}\backslash NP)/(S_{INF}\backslash NP) & S_{INF}\backslash NP \\
:\lambda p.p\ marcel' & :\lambda p\lambda y.seem'(py) & :\lambda p.p & :drink' \\
\cline{3-4}
 & & \multicolumn{2}{c}{S_{TO}\backslash NP : drink' \quad >} \\
\cline{2-4}
 & \multicolumn{3}{c}{S\backslash NP : \lambda y.seem'(drink'y) \quad >} \\
\hline
\multicolumn{4}{c}{S : seem'(drink'marcel') \quad >}
\end{array}$$

This analysis can therefore be viewed as a lexicalized version of Jacobson's (1990; 1992b) analysis of raising, according to which a unary composition combinator or "Geach Rule" applies to to a suitably slash-modality -restricted category which we might write as $seems := S/_{\mathbf{B}}S_{TO} : seem'$. However, all unary rules in the present version of CCG are lexicalized.

Auxiliaries should be analyzed as modality-contributing raising verbs of this kind, as Clark (1997) points out.

5.6 Object-Control

Persuade is one of a class of verbs where surface objects control an infinitival complement's subject, and which are completely free in their interaction with other operations such as passivization and Heavy Noun Phrase Shift:

(40) a. I persuaded Marcel to take a bath.
b. I persuaded Marcel to bathe himself.
c. Marcel was persuaded to take a bath.
d. I persuaded to take a bath my very heavy friend from Hoboken.

The CCG lexical entry for such verbs is as in the following example:

(41) persuaded := $((S\backslash NP)/(S_{TO}\backslash NP))/NP : \lambda x\lambda p\lambda y.persuade'(p(ana'x))xy$

The subject of the infinitive at the level of logical form is a proterm $ana'x$ bound to the object. The controlled infinitival subject may in turn bind a reflexive, as in (40b), to make logical forms like the following for (40b), which is consistent with Condition C (28):

(42) $S : persuade'(bathe'(ana'(ana'marcel'))(ana'marcel'))marcel'me'$

The category permits HNPS, on the assumption that the infinitival complement can type-raise:[20]

---

[20] The possibility of both extraction out of, and HNPS over, infinitival complements means that they must have both unraised and raised categories.

(43)
$$
\begin{array}{cccc}
\text{I} & \text{persuaded} & \text{to take a bath} & \text{my very heavy friends}\\
\hline
S/(S\backslash NP): & ((S\backslash NP)/VP_{TO})/NP: & S\backslash(S/VP_{TO}): & S\backslash(S/NP):\\
\lambda p.p\ me' & \lambda x\lambda y\lambda z.persuade'yxz & \lambda q.q\ (take'bath') & \lambda r.r\ friends'\\
\cline{1-2}
\multicolumn{2}{r}{>\mathbf{B}^{2}} & & \\
\multicolumn{2}{c}{(S/VP_{TO})/NP:} & & \\
\multicolumn{2}{c}{\lambda x\lambda y.persuade'yx\ me'} & & \\
\cline{1-3}
\multicolumn{3}{r}{<\mathbf{B}_{\times}} & \\
\multicolumn{3}{c}{S/NP:\lambda x.persuade'(take'bath')x\ me'} & \\
\cline{2-4}
\multicolumn{4}{r}{<}\\
\multicolumn{4}{c}{S:persuade'(take'bath')friends'me'}
\end{array}
$$

A small class of verbs like *see* take bare infinitival complements and seem to be similarly free (although the passive form mysteriously needs a *to*-infinitival for which there is no correponding active).

(44) a. I saw Marcel take a bath
b. Marcel was seen to take a bath.
c. I saw take a bath my very heavy friend from Hoboken.

The category is parallel to (41):

(45) saw := $((S\backslash NP)/(S_{inf}\backslash NP))/NP : \lambda x\lambda p\lambda y.see'(p(ana'x))xy$

Other superficially similar control verbs are more idiosyncratic, and the data are sometimes a little unclear. *Expect* seems to passivize, but not to be compatible with Heavy NP Shift:

(46) a. I expect Marcel to take a bath
b. Marcel was expected to take a bath.
c. *I expected to take a bath my very heavy friend from Hoboken.

The latter observation can be captured by imposing the diamond type on the first slash, preventing the backward crossed rule from composing *expect* and *to take a bath* analogously to derivation (43):

(47) expected := $((S\backslash NP)/(S_{TO}\backslash NP))/_{\diamond}NP : \lambda x\lambda p\lambda y.expect'(p(ana'x))xy$

Other object control verbs are more restricted, apparently rejecting not only HNPS but also passive:

(48) a. I wanted Marcel to take a bath
b. *Marcel was wanted to take a bath.
c. *I wanted to take a bath my very heavy friend from Hoboken.

We follow Jackendoff (1972) and Sag and Pollard (1991) in assuming that the incompatibility of passivization arises from the thematic role of the object in interaction with the interpretation of the passive itself. While this could be realized syntactically via morphologically null case-marking, we will assume for present purposes that the anomaly of (48b) is semantic.[21]

(49) wanted := $((S\backslash NP)/(S_{TO}\backslash NP))/_{\diamond}NP : \lambda x \lambda p \lambda y.want'(p(ana'x))xy$

Bach, Dowty, and Jacobson treat object control verbs and other ditransitives as having a "wrapping" category, taking the object and infinitival arguments in the opposite order and combining them with a special combinatory rule. As in the case of raising, the present category (49) can be seen as lexicalizing a version of this analysis involving a unary version of the "commuting" combinator **C**, where

(50) $\mathbf{C} fxy \equiv fyx$

However, the inclusion of WRAP as a projective syntactic operator considerably complicates the account of coordination developed in section 6.2 below, so the present theory continues to lexicalize all unary rules, including WRAP.

5.7 Subject-Control

A number of intransitive verbs support subject control:

(51) I wanted/expected/promised to take a bath

The categories are like the following:

(52) wanted := $(S\backslash NP)/(S_{TO}\backslash NP) : \lambda p \lambda y.want'(p(ana'y))y$

A much smaller class of subject control verbs take objects, notably *promise*. These verbs are incompatible with HNPS and passive:

(53) a. I promised Marcel to take a bath
b. *Marcel was promised to take a bath.
c. *I promised to take a bath my very heavy friend from Hoboken.

The category is as follows:

(54) promised := $((S\backslash NP)/(S_{TO}\backslash NP))/_{\diamond}NP : \lambda x \lambda p \lambda y.promise'(py)xy$

---

[21] In fact, a search on the internet turns up positive examples of such sentences, such as the following utterance by a native speaker of (Australian) English: *I was writing it because I was wanted to write, but I didn't know what I wanted to write*.

Again, the anomaly of passive is assumed to be semantic in origin, presumable stemming from the absence of an LF-commanding antecedent for the complement subject. Oddly, the past participle of *promise* (but apparently not *want*) *is* allowed with subject control if the complement is also passive, as in

(55) Marcel was promised to be left alone.

This observation seems to confirm Jackendoff's view that the anomaly of certain controlled passives is semantic rather than syntactic. [22]

5.8 Other Lexically Headed Bounded Constructions

Certain other constructions that have received attention within Construction Grammar (CxG) approaches (Goldberg 1995, 2006; Croft 2001) can be similarly lexicalized in CCG.

For example, we will assume that passives are derived from actives via lexical function application of the following category associated with the morpheme *-en*, applying to the first rightward argument of the base category, where $/\ldots$ schematizes over categories with zero or more further rightward arguments, such as ditransitives and control verbs, and $\lambda\ldots$ similarly schematizes over their interpretations:

(56) -en := $((S\backslash NP)/\ldots)\backslash_{LEX}(((S\backslash NP)/\ldots)/NP) : \lambda p\lambda\ldots\lambda x.p\ldots x\ one'$

This category yields the following lexical entries for the passives of the verbs discussed above:[23]

(57) a. proven := $S_{EN}\backslash NP : \lambda x.prove'x\ one'$
b. given := $(S_{EN}\backslash NP)/NP : \lambda x\lambda y.give'yx\ one'$
c. persuaded := $(S_{EN}\backslash NP)/(S_{TO}\backslash NP) : \lambda p\lambda y.persuade'(p(ana'y))y\ one'$
d. promised := $(S_{EN}\backslash NP)/(S_{TO}\backslash NP) : \lambda p\lambda y.promise'(p(ana'one'))y\ one'$

The latter category licenses both *Marcel was promised to be left alone*, and *#Marcel was promised to leave*. On the simplifying assumption that *be*, like *to*, is an identity function, so that *to be left alone* is semantically an agentless passive $S_{TO}\backslash NP : \lambda x.leave\text{-}alone'(ana'x')one'$, the respective logical forms are as follows:

---

[22] Again, similar to what is noted in footnote 21, a search on internet comes up with plenty of positive examples, such as *Morgan was promised to receive $52 million for the project in the next fiscal year*.

[23] *By*-passives are assumed to be derived by a similar rule treating the *by*-PP as a manner adverbial semantically linked by an event variable that we suppress here.

(58) a. $promise'(leave\text{-}alone'(ana'marcel')one')marcel'one'$
b. $promise'(leave'(ana'one'))marcel'one'$

Both are impeccable in terms of the binding theory, so the anomaly of the latter must stem from aspects of semantic representation that are not addressed here. It is similarly assumed that the anomaly of the following passives has a semantic source (see Huddleston and Pullum 2002:1432), although in these cases feature-based lexicalized exclusion is technically possible (cf. Croft 2001:49):

(59) a. #The best beaches on the East Coast are boasted by Skegness.
b. #Seven dollars was cost by this pack of cigarettes.
c. #The Famous Five were befallen by a strange adventure.
d. #A hundred people are held by this auditorium.
e. #Politics was being talked by the guests
f. #A ton is weighed by this suitcase.

Similarly, the following examples suggest that the "*way* construction" analyzed by Jackendoff (1990) and Goldberg (1995, 2006) is specifically headed by the reflexive *his way*:

(60) a. Marcel slept his way to the top.
b. # Marcel slept a/the/Anna's way to the top.
c. # Marcel slept his path/career to the top.

We can therefore regard such reflexives as a morpholexical operator analogous to the reflexive (29). For example:

(61) -his way := $((S\backslash NP_{3sn})\backslash_{LEX}(S\backslash NP_{3sn}))/PP_{LOC}$
$: \lambda p\lambda q\lambda y.cause'(iterate'(qy))(result'(py))$

Alternatively, the possessive pronouns themselves can be considered as bearing an alternative, "multiplely-rooted", category, mapping the word "way" (or some very restricted class of constituents of type N headed by nouns like "way") onto the above category via vacuous abstraction, thus:[24]

(62) -his := $(((S\backslash NP_{3sn})\backslash_{LEX}(S\backslash NP_{3sn}))/PP_{LOC})/''way''$
$: \lambda i\lambda p\lambda q\lambda y.cause'(iterate'(qy))(result'(py))$

The fact that these categories make phrases like "his way to the top" into ad-

---

[24] See Hockenmaier 2003a for an extended analysis of head-word-feature passing in CCG parsing, where it is needed for statistical modeling.

juncts may explain the fact that extractions like the following seem bad:[25]

(63) a. *Where did you sleep your way to?
b. *To the door Marcel sneezed his way.

A large number of other constructions identified by Goldberg can be lexicalized with similar apparatus, from idioms like “kick the bucket” and a number of causatives like “hammer the metal flat” to fully productive constructions such as the ethic dative exemplified in “cry me a river” (cf. Abeille and Schabes 1989 and Kay 2002). Some unbounded cases are considered in section 6.4 below.

6 THE UNBOUNDED CONSTRUCTIONS

The effect of including rules corresponding to the combinators **B** and **T** is to induce a rebracketing and reordering calculus over the strings and derivations that the lexicon and application alone determine, in which every type-driven derivational step is guaranteed to project function argument relations correctly. The fact that the syntactic combinators can be lexically restricted by typed-slashes means that languages like English and Dutch, in which reordering is quite limited, can be captured without generating undesired scrambled word orders. A number of linguistic predictions follow.

6.1 Unbounded Extraction and the Across-the-Board Condition

Since complement-taking verbs like *think*, $VP/_{\diamond}S$, can in turn compose via rule (13a) with fragments like *Marcel proved*, S/NP, derived as in (20), we correctly predict the fact that right-node raising is unbounded, as in (65a), and also provide the basis for an analysis of the similarly unbounded character of leftward extraction, as in (65b), without movement or empty categories, via the following category for the relative pronoun:

(64) that := $(N\backslash N)/(S/NP) : \lambda p\lambda n\lambda x.(nx)\wedge(px)$

---

[25] We are indebted to Cem Bozsahin for drawing our attention to examples like (63).

(65) a. [I conjectured]$_{S/NP}$ and [you think Marcel proved]$_{S/NP}$ completeness.
b. The result [that]$_{(N\backslash_{\star}N)/_{\star}(S/NP)}$ [I conjectured]$_{S/NP}$ and [you think Marcel proved]$_{S/NP}$.
c. $*$The man [who]$_{(N\backslash_{\star}N)/_{\star}(S/NP)}$ [[you think that]$_{S/_{\diamond}S}$ [proved completeness]$_{S\backslash NP}$]$_{*S/NP}$.

It is the category (64) of the relative pronoun that establishes the long-range dependency between noun and verb (via the non-essential use of the (non-essential) variable x in the logical form). This relation too is established in the lexicon: syntactic derivation merely projects it onto the clausal logical form. In the terms of the Minimalist Program (MP) of Chomsky, in which such relationships are established by the operation MOVE, it should be clear that CCG reduces this operation to the other major MP operation MERGE, since composition and type-raising, as well as application, correspond to the latter more basic operation. It is the $\diamond$ feature on the complement-taking verb *think* that allows the crucial composition in (65b) and prevents crossed composition in the $*$*that-t* constraint-violating (65c).[26].

The $\star$ type of the conjunction category (11) means that it can combine like types *only* by the application rules (5). Hence, as in GPSG (Gazdar 1981), this type-dependent account of extraction and coordination, as opposed to the standard account using structure-dependent rules, makes the across-the-board condition (ATB, Williams (1978)) on extractions from coordinate structures a prediction or theorem, rather than a stipulation, as consideration of the types involved in the following examples will reveal:[27]

---

[26] See *SSI* and Baldridge 2002 for details, including discussion of the possibility of subject extraction from bare complements, and other extraction asymmetries

[27] Lakoff (1986) suggested on the basis of examples first noticed by Ross 1967 and Goldsmith 1985 like *What did you go to the store and buy*, *How much beer can you drink and not get sick?*, *This is the stuff that those guys in the Caucasus drink every day and live to be a hundred*, that the coordinate structure constraint and the ATB exception are an illusion. This argument has recently been revived by Kehler (2002) and Asudeh and Crouch (2002). However, it has always also been argued (by Ross and Goldsmith, among others including Lakoff himself in an earlier incarnation) that these extractions involve another, non-coordinate, subordinating lexical category for "and", and as such do not constitute counterexamples to the coordinate structure and ATB constraints after all.

(66) a. The result [that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ [[Harry conjectured]$_{S/NP}$ and [Marcel proved]$_{S/NP}$]$_{S/NP}$]$_{N\backslash_{\diamond}N}$
b. The result [that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ [[Harry conjectured]$_{S/NP}$ and [Marcel proved was correct]$_{S/NP}$]$_{S/NP}$]$_{N\backslash_{\diamond}N}$
c. The result that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ *[[Harry conjectured]$_{S/NP}$ and [Marcel proved it]$_{S}$]]
d. The result that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ *[[Harry conjectured it]$_{S}$ and [Marcel proved]$_{S/NP}$]

It also predicts Williams' "same case" condition on the ATB

(67) a. The result that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ *[[Marcel proved]$_{S/NP}$ and [amazed me]$_{S\backslash NP}$]
b. The result that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ *[[Marcel proved was correct]$_{S/NP}$ and [amazed me]$_{S\backslash NP}$]
c. The result that$_{(N\backslash_{\diamond}N)/_{\diamond}(S/NP)}$ *[[amazed me]$_{S\backslash NP}$ and [Marcel proved]$_{S/NP}$]

However, in the case of (67c), there is an alternative derivation (68a) that treats *Marcel proved* as an entire reduced relative clause modifier of type $N\backslash N$, which can coordinate with *that amazed me*$_{N\backslash N}$, equivalent to (68b), so that (67c) is allowed:[28]

(68) a. The result [[that amazed me]$_{N\backslash N}$ and [Marcel proved]$_{N\backslash N}$]
b. The result [[that amazed me]$_{N\backslash N}$ and [that Marcel proved]$_{N\backslash N}$]

This alternative is not available for (67a,b), since the verbphrase $S\backslash NP$ cannot act as a reduced relative. Thus, we also capture this asymmetric exception to the same-case condition on the across-the-board exception to Ross's Coordinate Structure Constraint.

6.2 Argument Cluster Coordination

This apparatus has also been applied to a wide variety of coordination phenomena, including English "argument-cluster coordination", "backward gapping"

---

[28] We pass over the question of exactly how reduced relatives are assigned the category $N\backslash N$, noting that in the CCG version of the Penn treebank (CCGbank, Hockenmaier and Steedman (2007)), this is done with a unary rule that turns SS/NP into $N\backslash N$.

and verb-raising constructions in Germanic languages, and English gapping. The first of these is illustrated by the following analysis, from Dowty (1988), in which the ditransitive verb category$(VP/NP)/_{\diamond}NP$ is abbreviated as DTV, and the transitive verb category VP/NP is abbreviated as TV:[29]

(69)
$$\begin{array}{cccccc}
\text{give} & \text{Walt} & \text{the salt} & \text{and} & \text{Malcolm} & \text{the talcum}\\
\overline{} & \overline{}{}^{<\mathbf{T}} & \overline{}{}^{<\mathbf{T}} & \overline{\phantom{(X\backslash_{\star}X)}} & \overline{}{}^{<\mathbf{T}} & \overline{}{}^{<\mathbf{T}}\\
DTV & TV\backslash_{\diamond}DTV & VP\backslash TV & (X\backslash_{\star}X)/_{\star}X & TV\backslash_{\diamond}DTV & VP\backslash TV\\
 & \multicolumn{2}{c}{\overline{\phantom{VP\backslash_{\diamond}DTVVVVVVV}}{}^{<\mathbf{B}}} & & \multicolumn{2}{c}{\overline{\phantom{VP\backslash_{\diamond}DTVVVVVVV}}{}^{<\mathbf{B}}}\\
 & \multicolumn{2}{c}{VP\backslash_{\diamond}DTV} & & \multicolumn{2}{c}{VP\backslash_{\diamond}DTV}\\
 & & & \multicolumn{3}{c}{\overline{\phantom{(VP\backslash_{\diamond}DTV)\backslash_{\star}(VP\backslash_{\diamond}DTV)}}{}^{>}}\\
 & & & \multicolumn{3}{c}{(VP\backslash_{\diamond}DTV)\backslash_{\star}(VP\backslash_{\diamond}DTV)}\\
 & \multicolumn{5}{c}{\overline{\phantom{(VP\backslash_{\diamond}DTV)\backslash_{\star}(VP\backslash_{\diamond}DTV)VVVVVVVVVVVV}}{}^{<}}\\
 & \multicolumn{5}{c}{VP\backslash_{\diamond}DTV}\\
\multicolumn{6}{c}{\overline{\phantom{(VP\backslash_{\diamond}DTV)\backslash_{\star}(VP\backslash_{\diamond}DTV)VVVVVVVVVVVVVVVVV}}{}^{<}}\\
\multicolumn{6}{c}{VP}
\end{array}$$

Since we have assumed the previously discussed rules of forward type-raising ($>$**T**) and forward composition ($>$**B**), this construction is correctly predicted to exist in English by arguments of symmetry, which imply that their backward varieties, $<$**T** and $<$**B** must also be assumed.

Given independently motivated limitations on type-raising, examples like the following are still disallowed:[30]

(70) *Three mathematicians [[in ten]$_{PP}$ [derive a lemma,]$_{S\backslash NP}$] and [in a hundred prove completeness.]

6.3 Germanic Crossing Dependencies

The availability of *crossed* composition (26) to the grammar of in Dutch and certain Swiss dialects of German allows crossed dependencies, as in the following example (from Shieber):

---

[29] In more recent work, Dowty has disowned CCG in favour of TLG, because of "intrinsic" use of logical form to account for binding phenomena that it entails, as discussed above. See SSI for further discussion.

[30] This appears to offer an advantage over non-type-raising accounts using the product operator $\bullet$ of Lambek (Pickering and Barry 1993; Dowty 1997).

(71)

$$\begin{array}{ccccccl}
\text{das} & \text{mer} & \text{em Hans} & \text{es huus} & \text{hälfed} & \text{aastriiche} & \\
\text{that} & \text{we}-\text{NOM} & \text{Hans}-\text{DAT} & \text{the house}-\text{ACC} & \text{helped} & \text{paint} & \\
\cline{2-6}
 & NP^{\uparrow}_{nom} & NP^{\uparrow}_{dat} & NP^{\uparrow}_{acc} & ((S_{+SUB}\backslash NP_{nom})\backslash NP_{dat})/VP & VP\backslash NP_{acc} & \\
\cline{5-6}
 & & & & \multicolumn{2}{c}{((S_{+SUB}\backslash NP_{nom})\backslash NP_{dat})\backslash NP_{acc}} & >\mathbf{B}_{\times} \\
\cline{4-6}
 & & & \multicolumn{3}{c}{(S_{+SUB}\backslash NP_{nom})\backslash NP_{dat}} & > \\
\cline{3-6}
 & & \multicolumn{4}{c}{S_{+SUB}\backslash NP_{nom}} & > \\
\cline{2-6}
 & \multicolumn{5}{c}{S_{+SUB}} & >
\end{array}$$

“that we helped Hans paint the house”

The · slash-type of the verbs *hälfed* and *aastriichte* permits the forward crossed composition rule (26b) to apply. The tensed verb is distinguished as the head of a subordinate clause via the feature SUB. The type-raised NP categories are abbreviated as $NP^{\uparrow}_{case}$, since the fact that they are raised is not essential to understanding the point about crossing dependencies. It is correctly predicted that the following word orders are also allowed in at least some dialects (Shieber 1985:338-9):

(72) a. a. das mer em Hans hälfed es huus aastriiche.
b. b. das em Hans mer es huus hälfed aastriiche.

The construction is completely productive, so the dependencies are not only intersective, but unbounded. For example, we have the following (also from Shieber):

(73)

$$\begin{array}{ccccccccl}
\text{das} & \text{mer} & \text{d}'\text{chind} & \text{em Hans} & \text{es huus} & \text{lönd} & \text{hälfe} & \text{aastriiche} & \\
\text{that} & \text{we}-\text{NOM} & \text{the children}-\text{ACC} & \text{Hans}-\text{DAT} & \text{the house}-\text{ACC} & \text{let} & \text{help} & \text{paint} & \\
\cline{2-8}
 & NP^{\uparrow}_{nom} & NP^{\uparrow}_{acc} & NP^{\uparrow}_{dat} & NP^{\uparrow}_{acc} & ((S_{+SUB}\backslash NP_{nom})\backslash NP_{acc})/VP & (VP\backslash NP_{dat})/VP & VP\backslash NP_{acc} & \\
\cline{6-7}
 & & & & & \multicolumn{2}{c}{(((S_{+SUB}\backslash NP_{nom})\backslash NP_{acc})\backslash NP_{dat})/VP} & & >\mathbf{B}^{2}_{\times} \\
\cline{6-8}
 & & & & & \multicolumn{3}{c}{(((S_{+SUB}\backslash NP_{nom})\backslash NP_{acc})\backslash NP_{dat})\backslash NP_{acc}} & >\mathbf{B}_{\times} \\
\cline{5-8}
 & & & & \multicolumn{4}{c}{((S_{+SUB}\backslash NP_{nom})\backslash NP_{acc})\backslash NP_{dat}} & > \\
\cline{4-8}
 & & & \multicolumn{5}{c}{(S_{+SUB}\backslash NP_{nom})\backslash NP_{acc}} & > \\
\cline{3-8}
 & & \multicolumn{6}{c}{S_{+SUB}\backslash NP_{nom}} & > \\
\cline{2-8}
 & \multicolumn{7}{c}{S_{+SUB}} & >
\end{array}$$

“that we let the children help Hans paint the house”

Again the unbounded dependencies are projected from the lexical frame of the verb, without syntactic movement.

Such crossing dependencies cannot be captured by CFG and have given rise to proposals for “verb-raising” transformational operations. The fact that CCG can express them implies that it is trans-context-free in terms of generative capacity. CCG is is in fact provably weakly equivalent to TAG, Head Grammar (Pollard 1984), and Linear Indexed Grammar (Aho 1968), a group constituting the least expressive natural generalization of CFG that has so far been identified in the spectrum of mildly context-sensitive grammars identified by Joshi (1988). This equivalence gives rise to a polynomial time worst-case complexity result (Vijay-Shanker and Weir 1990, 1993, 1994).

Recent work has begun to consider the relationship between these formalisms in terms of their *strong* generative capacity: Hockenmaier and Young (2008) and Koller and Kuhlmann (2009) show there are indeed differences in the structural analysis which can be assigned by CCG and TAG.

6.4 Other Lexically-Headed Unbounded Constructions

The following examples suggest that "*tough*-movement" is unbounded and lexically headed by the eponymous class of adjectives:

(74) a. John is easy to please.
b. Marcel is hard to believe we could please.

This observation can be captured in the following category for the adjectives, subcategorizing, like the relative pronoun, for T/NP

(75) tough := $(S_{AP}\backslash NP)/((S_{TOINF}\backslash NP)/NP) : \lambda p \lambda x.\mathit{difficult}'(px\ \mathit{one}')$

Similarly, the following examples suggest that the "more-more" construction discussed by Goldberg (1995, 2006) and Jackendoff (1990) is headed by the definite article:

(76) a. The more books you buy, the merrier person you think you become.
b. #A/several/some more books you buy, the merrier person you think you become.

This observation can be captured in the following category "multiply rooted" category for the definite, subcategorizing, like the relative pronoun, for T/NP:[31].

(77) the := $((((S/(S/NP))/NP_{COMP})/(S/NP))/NP_{COMP})/''\mathit{the}''$
$: \lambda i \lambda x \lambda p \lambda q \lambda y.\mathit{cause}'(qy)(px)$

---

[31] We pass over the elliptical form of this construction, as in "The more, the merrier," which is presumably mediated by a related lexically derived category.

As with the bounded lexically headed constructions, many more unbounded constructions offer themselves as lexicalizable in this way. For example, the following (from Kay 2002) seems a suitable case for treatment with “doing” as head:

(78) a. What’s this fly doing in my soup?
b. What do you think this fly is doing in my soup?
c. What’s this fly think it’s doing in my soup?
d. This fly’s doing no good in my soup.

7 SCRAMBLING

Many languages, such as Turkish and Japanese, permit more freedom in word order than languages like English and Dutch. The most basic expression of this is local scrambling, in which the arguments of a verb appear in permuted orders within its clausal domain. This can be seen in the Turkish transitive sentence (79a) and its scrambled counterpart (79b), adapted from Hoffman (1995):

(79) a. Ayse kitabi okuyor
Ayse-NOM book-ACC read-PROG

b. Kitabi Ayse okuyor
book-ACC Ayse-NOM read-PROG

‘Ayse reads the book.’

Long distance scrambling, on the other hand, describes the appearance of an argument of a lower clause intermixed with the arguments of a higher clause. For example, the argument *kitabi* ‘book’ of the lower verb *okudugunu* ‘read’ scrambles out of its “base” position in (80a) into the matrix clause (80b) (from Hoffman 1995):

(80) a. Fatma [Esra’nın kitabi okudugunu] biliyor.
Fatma [Esra-GEN book-ACC read-GER-ACC] know-PROG
‘Fatma knows that Esra read the book.’

b. *Kitabi*$_i$ Fatma [Esra’nın t$_i$ okudugunu] biliyor.
book-ACC$_i$ Fatma [Esra-GEN t$_i$ read-GER-ACC] know-PROG
‘As for the book, Fatma knows that Esra read it.’

The essential tension which arises in providing an analysis of *local* scrambling is that between utilizing base generation or devising a sufficiently liberal syntactic system. In CCG, base generation amounts to lexical ambiguity for verbs that allow scrambling. For example, if we assume the Turkish lexicon contains the two categories in (81) for *okuyor* ‘read’, both of the word orders in (79) are captured, as shown in derivations (82) and (83).

(81) a. okuyor := $(S\backslash NP_{nom})\backslash NP_{acc}$

b. okuyor := $(S\backslash NP_{acc})\backslash NP_{nom}$

(82)
$$\begin{array}{cccl}
\text{Ayse} & \text{kitabi} & \text{okuyor} & \\
\overline{NP_{nom}} & \overline{NP_{acc}} & \overline{(S\backslash NP_{nom})\backslash NP_{acc}} & \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash NP_{nom}} & < \\
\cline{1-3}
\multicolumn{3}{c}{S} & <
\end{array}$$

(83)
$$\begin{array}{cccl}
\text{Kitabi} & \text{Ayse} & \text{okuyor} & \\
\overline{NP_{acc}} & \overline{NP_{nom}} & \overline{(S\backslash NP_{acc})\backslash NP_{nom}} & \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash NP_{acc}} & < \\
\cline{1-3}
\multicolumn{3}{c}{S} & <
\end{array}$$

It may appear that using multiple categories as such fails to recognize the connection between the two orders; however, they can actually be generated from the specification of a single category, given a suitable theory of the lexicon. For example, one could assume that the category (81a) is the kernel category and use a lexical rule to generate (81b) from it. A more involved strategy is that advocated by Foster (1990), where unordered categories in the lexicon potentially project multiple ordered categories for use by the grammar. The difference between Foster’s strategy and one which uses lexical rules is that his approach does not require any language specific rules in order to create ordered categories from an unordered kernel category. This retains a tight connection between the different orders in a principled manner.

An alternative to multiple categories is to relax the definitions of categories and combinatory rules to allow a single category to project multiple word orders directly in syntactic combination. This is the strategy advocated by Hoffman (1995) to deal with scrambling in Turkish. She allows categories to contain multi-set arguments, as in (84), and redefines the combinatory rules to be sensitive to multi-sets, as shown for backward application in (85).[32] With this application rule, the category (84) can consume its arguments in either order.

---

[32] The α is a variable for a set of categories.

(84) okuyor $:= S\{\backslash NP_{nom}, \backslash NP_{acc}\}$

(85) $Y \quad X(\alpha \uplus \{\backslash Y\}) \quad \Rightarrow \quad X\alpha \qquad\qquad (<)$

While this modified CCG apparatus suffices for local scrambling, it does not directly handle long distance scrambling. Accordingly, Hoffman redefines the type-raising and composition rules to work with such categories, but does so in a manner which increases the generative capacity of the system. In order to retain mild context-sensitivity while using multi-sets in categories, Baldridge (2002) provides more conservative definitions of the rules that permit the CCG system to use a flexible category in the same way as if it had access to an entire set of ordered categories that collectively capture the scrambled word orders. Then, the permutative powers already inherent in the crossed composition rules can be utilized for long distance scrambling. For example, to derive (80b), the subject *Fatma* of the matrix verb must type-raise and forward *cross* compose into the verbal cluster in order for the derivation to proceed, as shown in (86).[33]

(86)
$$\begin{array}{ccccc}
\text{Kitabi} & \text{Fatma} & \text{Esra'nın} & \text{okudugunu} & \text{biliyor}\\
\hline
NP_{acc} & NP_{nom} & NP_{gen} & S_{acc}\{\backslash NP_{gen}, \backslash NP_{acc}\} & S\{\backslash NP_{nom}, \backslash S_{acc}\}\\
\cline{2-2}\cline{3-4}
 & S/(S\backslash NP_{nom}) \;{>}\mathbf{T} & \multicolumn{2}{c}{S_{acc}\backslash NP_{acc} \;<} & \\
\cline{3-5}
 & & \multicolumn{3}{c}{(S\backslash NP_{nom})\backslash NP_{acc} \;{<}\mathbf{B}}\\
\cline{2-4}
 & \multicolumn{3}{c}{S\backslash NP_{acc} \;{>}\mathbf{B}_{\times}} & \\
\cline{1-3}
\multicolumn{3}{c}{S \;<} & &
\end{array}$$

Under this account, local scrambling is viewed as a clause-bounded phenomena, while long distance scrambling takes a form similar to other “extraction”-type phenomena, such as relativization.

The word order of Turkish is of course not entirely free. While verbal arguments can scramble around, the elements of some noun phrases are more fixed, as can be seen in (87):

---

[33] In this derivation, we suppress the {} brackets around singleton sets to improve legibility.

(87) a. [Siyah kedi] geldi
[black cat] come-PAST
'The black cat came in.'

b. *Kedi siyah geldi
cat black come-PAST

c. *Siyah geldi kedi
black come-PAST cat

Under the assumption that the category of *siyah* 'black' has a rightward slash, (87b) is obviously blocked. Modal control is crucial in the case of (87c); the slash must be the non-permuting slash-type in order to bar backward crossed composition from applying:

(88)
$$\begin{array}{ccc} \text{siyah} & \text{geldi} & \text{kedi} \\ \hline NP_x/_{\diamond}NP_x & S\backslash NP_{nom} & NP_{nom} \\ \cline{1-2} \multicolumn{2}{r}{*} & \end{array}$$

Turkish thus demonstrates the need for liberal access to the crossed composition rules at the clausal level while retaining tighter control over them at the phrasal level.[34] This type of control is needed for harmonic composition rules as well: for example, Baldridge's (2002) analysis of syntactic extraction asymmetries in Tagalog maintains tight control over forward harmonic composition while allowing local scrambling, and Trechsel (2000) utilizes restrictions with similar effects in Tzotzil.

With its universal rule set and lexical control over it via modally typed slashes, CCG supports these competing tensions straightforwardly and without recourse to powerful syntactic rules, structure-dependent transformations, or other devices.

8 GAPPING AND THE ORDER OF CONSTITUENTS

The phenomenon of "argument cluster coordination" illustrated in (69) is an example of a much broader cross-linguistic generalization due to Ross (1970), concerning a relation between basic word-order parameters such as verb finality and constraints on deletion or gapping under coordination. While Ross originally framed his generalization in terms of a then-orthodox deep-

---

[34] It would indeed be possible to give adjectives a permutative slash, and this is indeed a necessary degree of freedom: possessive noun phrases in Turkish can be discontinuous (Hoffman 1995), allowing orders akin to (87c) in addition to (87a).

structural word order, and on that basis it has been challenged, when reformulated in terms of surface constituent order, it appears to hold.

Ross (1970) noticed that direction of gapping (leftward or rightward) depends on basic constituent order:

(89) a. SOV: *SOV and SO, SO and SOV
b. VSO: VSO and SO, *SO and VSO
c. SVO: SVO and SO, *SO and SVO

Apparent exceptions such as Zapotec (traditionally VSO) and Dutch (traditionally SOV), which allow both leftward and rightward gapping, also have conspicuously mixed word-order, rendering the identification of “basic” word-order moot in those languages.

8.1 Gapping and SOV Word Order

On the assumption that type-raising is order-preserving, and defined over the Japanese SOV lexical type for transitive verbs like (90), the subject and object NP can not only combine with the verb by forward application, but also by forward composition, as in (91).

(90) tazuneta := $(S\backslash NP_{nom})\backslash NP_{acc} : \lambda x\lambda y.visit'xy$

(91) a. Ken-ga Naomi-o tazuneta.
Ken-NOM Naomi-ACC visit-PAST.CONCL
‘Ken visited Naomi.’

b.

$$\dfrac{\dfrac{\dfrac{\text{Ken-ga}}{S/(S\backslash NP_{nom})}{\scriptstyle >\mathbf{T}} \quad \dfrac{\text{Naomi-o}}{(S\backslash NP_{nom})/((S\backslash NP_{nom})\backslash NP_{acc})}{\scriptstyle >\mathbf{T}}}{S/((S\backslash NP_{nom})\backslash NP_{acc})}{\scriptstyle >\mathbf{B}} \quad \dfrac{\text{tazuneta}}{(S\backslash NP_{nom})\backslash NP_{acc}}}{S}{\scriptstyle >}$$

The resulting nonstandard constituent *Ken-ga Naomi-o* can therefore conjoin, in a mirror image of the English derivation (69):

(92) [Ken-ga Naomi-o], [Erika-ga Sara-o] tazuneta.
$S/((S\backslash NP_{nom})\backslash NP_{acc})$ $S/((S\backslash NP_{nom})\backslash NP_{acc})$ $(S\backslash NP_{nom})\backslash NP_{acc}$
Ken-NOM Naomi-ACC Erika-NOM Sara-ACC visit-PAST.CONCL
‘Ken visited Naomi, and Erika, Sara.’

Ditransitives similarly allow larger argument clusters (to save space the intransitive category is abbreviated *VP*, the transitive *TV*, and the ditransitive

DTV):

(93) a. Kyooju-ga komonjo-o gakusee-ni kasita.
Professor-NOM manuscript-ACC student-DAT lent-PAST.CONCL
'The professor lent the manuscript to the student.'

b. Kyooju-ga komonjo-o gakusee-ni kasita.
>T >T >T
S/VP VP/TV TV/DTV DTV
>B
S/TV
>B
$S/DTV)$
>
S

In this case there is another derivation for the argument cluster:

(94) Kyooju-ga komonjo-o gakusee-ni kasita.
>T >T >T
S/VP VP/TV TV/DTV DTV
>B
VP/DTV
>B
S/DTV
>
S

Again, both derivations are guaranteed to yield identical logical forms, and all non-standard argument cluster constituents formed in both derivations can coordinate.

Ross's generalization that SOV verbs gap on the left conjunct is therefore captured: the Principles of Adjacency, Consistency, and Inheritance, together with the order-preserving constraint on type-raising that is by definition in an order-dependent language, permit any raised categories or rules of composition that would produce a *leftward*-looking function. "Forward Gapping" is therefore disallowed in any language with a pure verb-final lexicon:

(95) *Ken-ga Naomi-o tazunete, Erika-ga Sara-o
Ken-NOM Naomi-ACC visit-PAST.ADV Erika-NOM Sara-ACC
'Ken visited Naomi, and Erika, Sara.'

Dutch, which is often regarded as an SOV language, *does* allow coordinations on the above pattern in subordinate-clause conjunctions, in apparent exception to Ross' generalization:

(96) . . . dat Maaike aardappels eet en Piet bonen
. . . that Maaike potatoes eats and Piet beans
'. . . that Maaike eats potatoes and Piet beans.'

However, this exception is clearly related to the fact that Dutch has VSO/SVO word order as well as SOV. In CCG terms, this corresponds to the fact that its lexicon is not *purely* SOV, and that main verbs must be assumed to bear VSO categories. Ross' generalization should therefore be rephrased, as in *SP*, in terms of surface order, i.e available lexical type(s), not a single underlying order. Indeed, CCG rejects the very notion of "underlying word order."

This observation is relevant to the fact that Japanese also allows OSV word order, as in (97):

(97) Naomi-o Ken-ga tazuneta.
Naomi-ACC Ken-NOM visit-PAST.CONCL
'Ken visited Naomi.'

OS order can also give rise to constituent cluster coordination parallel to (92), as in (98):[35]

(98) [Naomi-o Ken-ga,] [Sara-o Erika-ga] tazuneta
$S/((S\backslash NP_{acc})\backslash NP_{nom})$ $S/((S\backslash NP_{acc})\backslash NP_{nom})$ $(S\backslash NP_{acc})\backslash NP_{nom}$
Naomi-ACC Ken-NOM, Sara-ACC Erika-NOM visit-PAST.CONCL
'Ken visited Naomi and Erika, Sara.'

As discussed in section 7, we can regard these variant constituent orders in Japanese as lexically specified, either via multiple verb categories or via explicitly unordered leftward verb categories such as (84), in keeping with the observation that local scrambling (as distinct from true extraction) is clause-bounded.

Unlike Hoffman's (1995) extension of CCG for scrambling, the formulation discussed in section 7 does not immediately allow the following pattern of coordination to be captured:

(99) ?[Naomi-o Ken-ga,] [Erika-ga Sara-o] tazuneta
$S/((S\backslash NP_{acc})\backslash NP_{nom})$ $S/((S\backslash NP_{nom})\backslash NP_{acc})$ $(S\backslash NP_{nom})\backslash NP_{acc}$
Naomi-ACC Ken-NOM, Erika-NOM Sara-ACC visit-PAST.CONCL
'Ken visited Naomi and Erika, Sara.'

---

[35] This fact precludes any attempt to account for (97) in terms of forward crossed composition, as in the following derivation:

(i)
$$\begin{array}{ccc}
\text{Naomi-o} & \text{Ken-ga} & \text{tazuneta} \\
\hline
S/(S\backslash NP_{acc})\ {>}\mathbf{T} & S/(S\backslash NP_{nom})\ {>}\mathbf{T} & (S\backslash NP_{nom})\backslash NP_{acc} \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash NP_{acc} \quad {*}{>}\mathbf{B}_{\times}} \\
\hline
\multicolumn{3}{c}{S \quad >}
\end{array}$$

The status of such coordinations in Japanese is not entirely clear. They tend to be judged less acceptable than the earlier alternatives, and seem to be pragmatically marked. (Interestingly, the similarly problematic SO&OSV pattern seems to be even less acceptable than the OS&SOV pattern in (99).) But the consensus seems to be that they are grammatical.

We will return briefly to this problem in section 8.3.

8.2 Gapping and VSO Word Order

It is obvious on arguments from symmetry that the VSO subject and object must be assumed to raise over VSO verbs like the Irish Gaelic transitive verb *chonaic* 'saw', (100), and to compose with each other and with adjuncts in an order-preserving way to yield a single function, as in (101).

(100) chonaic := $(S/NP)/_{\diamond}NP : \lambda x \lambda y.see'yx$

(101) a. Chonaic Eoghan Siobhán.
saw Eoghan Siobhán
'Eoghan saw Siobhán.'

b.

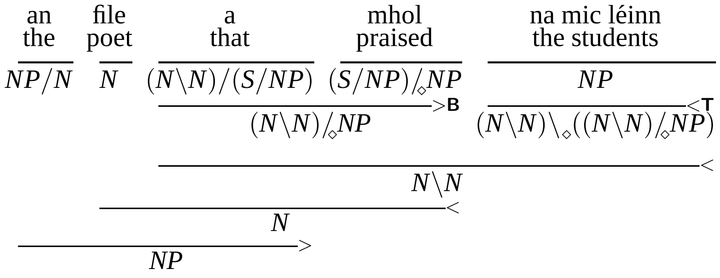
$$\begin{array}{ccc}
\text{Chonaic} & \text{Eoghan} & \text{Siobhán} \\
\overline{(S/NP))/_{\diamond}NP} & \overline{(S/NP)\backslash((S/NP)/NP)}^{\,<\mathbf{T}} & \overline{S\backslash(S/NP)}^{\,<\mathbf{T}} \\
\cline{2-3}
 & \multicolumn{2}{c}{S\backslash((S/NP)/NP) \quad {<}\mathbf{B}} \\
\hline
\multicolumn{3}{c}{S \quad {<}}
\end{array}$$

The diamond slash-type on the rigid category (100) prevents Irish from exhibiting scrambling in the mirror-image of Japanese (97), and prevents reordering analogous to Heavy NP shift (31) (which is otherwise allowed in Irish as in English), just as that modality does for English ditransitives (cf. (37b)). Nevertheless the category allows the formation of relative clauses in the usual way, which may as in Dutch be ambiguous (McCloskey 1978):[36]

---

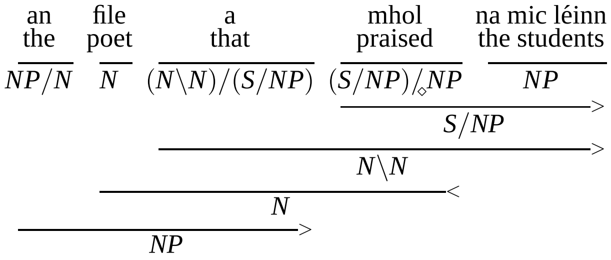
[36] See the discussion in Baldridge (2002) regarding extraction of the indirect and direct object arguments of English ditransitives for an alternative which uses a more complex set of slash modalities to allow extraction without scrambling.

(102)



‘the poet that praised the students’

(103)



‘the poet that the students praised’

Since such clusters necessarily bear leftward functor categories, when they coordinate they give the appearance of rightward gapping, in line with Ross’ generalization.

(104) Chonaic [Eoghan Siobhán] agus [Eoghnaí Ciarán].
$(S/NP))/_\diamond NP$ $S\backslash((S/NP)/NP)$ $(\mathrm{X}\backslash_\star \mathrm{X})/_\star \mathrm{X}$ $S\backslash((S/NP)/NP)$
saw Eoghan Siobhán and Eoghnaí Ciarán
‘Eoghan saw Siobhán, and Eoghnaí, Ciarán.’

Again the three principles correctly exclude the “backward gapping” construction that Ross (1970) held to be generally disallowed in strictly verb-initial languages:

(105) *[Eoghan Siobhán] agus chonaic [Eoghnaí Ciarán].
$S\backslash((S/NP)/NP)$ $(\mathrm{X}\backslash_\star \mathrm{X})/_\star \mathrm{X}$ $(S/NP))/_\diamond NP$ $S\backslash((S/NP)/NP)$
Eoghan Siobhán and saw Eoghnaí Ciarán

As in the case of SOV languages, allegedly VSO languages exist that allow leftward gapping as well as rightward. Zapotec (Rosenbaum 1977) is a standard example. However, like Dutch, Zapotec has mixed word order. It is therefore again consistent with a version of Ross’ generalization formulated as in *SP* in terms of surface orders and available lexical verb categories rather than "deep" or "underlying" word-order.

8.3 Gapping and SVO Word Order

The fact that gapping in English and other verb-medial languages is rightward (as in the VSO pattern) needs further apparatus since the NPs in the ungapped conjunct are separated by the SVO verb.

(106) Marcel proved completeness, and Gilbert, soundness.

Comparison with example (69) and the fact that the English lexicon contains a limited class of VSO verbs already suggests that an explanation is not far away.

SP, following Steedman 1990, proposes a class of "decomposition" rules which map the left conjunct onto a virtual VSO verb and a virtual SO argument cluster with the same category as the adjacent gapped right coordinate. The latter can coordinate with the virtual cluster and apply to the verb (whose interpretation has to be obtained contextually, possibly by processes akin to VP anaphora). This augmentation of CCG makes a number of correct predictions about the possibility of "Stripping" constructions in English. Karamanis (2000) and Bozsahin (2000) have used this apparatus to capture the kind of mixed-order gapping illustrated for Japanese in (99) in Greek and Turkish, respectively. White and Baldridge (2003) compile the effect of decomposition into a coordinating category to permit gaps to be parsed and realized in a computational implementation of CCG.[37] However, the decomposition analysis itself remains controversial and is passed over here.

9 INTONATION STRUCTURE AND PARENTHETICALS

We also have seen that, in order to capture coordination with rules adhering to the constituent condition, CCG generalizes surface constituency to give substrings like *Marcel proved* and even *a policeman a flower* the full status of constituents.

But if they are constitutents of coordinate constructions, they are predicted to be possible constituents of ordinary non-coordinate sentences as well. The

---

[37] The implementation is an open-source Java-based system and can be downloaded from `http://openccg.sourceforge.net`: see Baldridge et al. (2007) for details.

characteristics of intonation structure and the related phenomenon of parentheticalization show that this prediction is correct. (Because of restrictions of space, this part of the account is sketched in less detail than that on coordination. For details the reader is referred to Prevost (1995) and Steedman (1991, 2000a).)

9.1 English Intonation and Information Structure

Consider the following minimal pair of dialogs, in which intonational tunes are indicated both informally via parentheses and small capitals as before, and in the standard notation of Pierrehumbert (1980) and Pierrehumbert and Beckman (1988), in which prosodic phrases are specified solely in terms of two kinds of elements, the pitch accent(s) and the boundary:

(107) Q: I know who proved soundness. But who proved COMPLETENESS?
A: (MARCEL) (proved COMPLETENESS).
H* L L+H* LH%

(108) Q: I know which result Marcel PREDICTED. But which result did Marcel PROVE?
A: (Marcel PROVED) (COMPLETENESS).
L+H*LH% H* LL%

In (107A), there is a prosodic phrase on MARCEL including the sharply rising pitch accent that Pierrehumbert calls H*, immediately followed by an L boundary, perceived as a rapid fall to low pitch. There is another prosodic phrase having the somewhat later-rising and (more importantly) lower-rising pitch accent called L+H* on *COMPLETENESS*, preceded by null tone (and therefore interpolated low pitch) on the word *proved* and immediately followed by an utterance-final rising boundary, written LH%.

In (108A) above, the order of the two tunes is reversed: this time, the tune with pitch accent L+H* and boundary LH% occurs on the word *PROVED* in one prosodic phrase, *Marcel PROVED*, and the other tune with pitch accent H* and boundary LL% is carried by a second prosodic phrase *COMPLETENESS*.

The intuition that these tunes strongly convey systematic distinctions in discourse meaning is inescapable. For example, exchanging the answer tunes between the two contexts in (107) and (108) yields complete incoherence. Prevost and Steedman (1994) claim that the tunes L+H* LH% and H* L (or H*

LL%) are respectively associated with the “theme” and “rheme” of the sentence, where these terms are used in the sense of Mathesius (1929), Firbas (1964, 1966), and Bolinger (1989), and correspond roughly to a generalization of the more familiar terms “topic” and “comment”, which however are generally restricted by definition to traditional constituents.

Informally the theme can be thought of as corresponding to the content of a contextually available *wh*-question, which may be explicit, as in (107) and (108), or implicit in other discourse content. The position on the pitch accent, if any, in the theme, distinguishes words corresponding to “focused” elements of the content which distinguish this theme from other contextually available alternatives. The rheme can then be thought of as providing the answer to the implicit *wh*-question, with the pitch accent again marking focused words which distinguish this answer semantically from other potential answers. The system comprising the oppositions of theme/rheme and focus/background is known as information structure. Steedman (2000a) provides a more formal definition in terms of the “alternative semantics” of Rooth (1985, 1992), and the related “structured meanings” of Cresswell (1973, 1985), von Stechow (1991), and others.[38]

Since alternatives like the following are equally valid surface derivations in CCG, it will be obvious that CCG provides a framework for bringing intonation structure and its interpretation – information structure – into the same syntactic system as everything else:

(109)
$$\begin{array}{ccc}
\text{Marcel} & \text{proved} & \text{completeness}\\
\overline{NP : marcel'} & \overline{(S\backslash NP)/NP : prove'} & \overline{(S\backslash NP)\backslash((S\backslash NP)/NP)}\\
 & & : \lambda p.p\ completeness'\\
\overline{S/(S\backslash NP) : \lambda f.f\ marcel'}^{>\mathbf{T}} & \multicolumn{2}{c}{\overline{S\backslash NP : \lambda y.prove' completeness' y}^{<}}\\
\multicolumn{3}{c}{\overline{S : prove' completeness' marcel'}^{<}}
\end{array}$$

---

[38] The much-abused term “focus” is used in CCG strictly in the “narrow” or phonological sense of the term, to refer to the effects of contrast or emphasis on a word that ensues from the presence of a pitch-accent.

(110)
$$\begin{array}{ccc}
\text{Marcel} & \text{proved} & \text{completeness} \\
\hline
NP : marcel' & (S\backslash NP)/NP : prove' & S\backslash(S/NP) \\
\overline{\qquad\qquad\qquad}{>}\mathbf{T} & & : \lambda p.p\ completeness' \\
S/(S\backslash NP) : \lambda f.f\ marcel' & & \\
\multicolumn{2}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad}{>}\mathbf{B}} & \\
\multicolumn{2}{c}{S/NP : \lambda x.prove'x\ marcel'} & \\
\multicolumn{3}{c}{\overline{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}{<}} \\
\multicolumn{3}{c}{S : prove'completeness'marcel'}
\end{array}$$

Crucially, these alternative derivations are guaranteed to yield the same predicate argument relations, as exemplified by the logical form that results from (109) and (110), $prove'completeness'marcel'$. It follows that c-command-dependent phenomena such as binding and control can be captured at the level of logical form (Steedman 1991). However, the derivations build this logical form via different routes that construct lambda terms corresponding semantically to the theme and rheme. In particular the derivation (109) corresponds to the information structure associated with the intonation contour in (107), while derivation (110) corresponds to that in (108).

This observation can be captured by making pitch accents mark both arguments and results of CCG lexical categories with theme/rheme markers θ/ρ, as in the following category for a verb bearing an L+H* accent:

(111) proved := $(S_\theta\backslash NP_\theta)/NP_\theta : \lambda x\lambda y.*prove'xy$

The predicate is marked as focused or contrasted by the * marker in the logical form. θ/ρ marking is projected onto the arguments and result of constituents by combinatory derivation. The boundary tones like LH% have the effect of completing information structural constituents, and transfering theme/rheme marking to θ'/ρ' marking to constituent interpretations at logical form. We will pass over further details of exactly how this works, referring the reader to Prevost (1995) and to Steedman (2000a). The latter paper generalizes this approach to the full range of tunes identified by Pierrehumbert, including those with multiple pitch accents and multiple or disjoint themes and rhemes.

9.2 Parentheticals

While we will not discuss parentheticals in any detail here, it seems likely that they too should be defined in terms of information structural units. In most cases, the parenthetical intrusion itself appears at the boundary between theme and rheme, hence it is subject to the same constraints as intonational phrase boundaries:

(112) a. Marcel proved, so he claimed, a crucial theorem.
b. *Three mathematicians, expostulated Harry, in ten derive a lemma.

10 IMPLICATIONS FOR PERFORMANCE: THE STRICT COMPETENCE HYPOTHESIS

The minimum apparatus besides competence grammar that is required for processing consists of the characteristic automaton for the relevant class of grammars (including its possibly limited working memories), a minimal algorithm for applying the rules, and some memory for building interpretable structure. Any extra apparatus such as rule-orderings or "strategies," covering grammars, and the like, is otiose. To the extent that such extra stipulations are cross-linguistically universal, they complicate the problem of explaining language evolution. To the extent that they are language-specific, they do the same disservice to the problem of explaining child language acquisition.

The most restrictive hypothesis of all is that the processor involves no resources at all beyond the minimum specified above. Such processors are incapable of building intermediate structures other than those corresponding to the constituents defined by the competence grammar, and for this reason the hypothesis that the human processor has this character is called the "strict competetence" hypothesis (SCH).

One very simple processor adhering to this principle is based on the left-to-right version of the Cocke-Kasami-Young (CKY) parser (see Harrison 1978), a bottom-up parser which fills the cells of an $n \times n$ table or half-matrix t representing all spans between positions (i, j) in a string of n words.

The associativity of functional composition in interaction with type-raising potentially creates exponentially many multiple derivations for any given constituent for a given span with a given sense or interpretation (the so-called "spurious ambiguity" problem). It follows that such a parser will have exponential computational costs *unless* we either include a check that a newly-derived category spanning (i, j) *including its normalized logical form* is not already on the list in $t(i, j)$ before appending it (a suggestion first made by Karttunen (1989)), or preempt all necessarily redundant combination entirely, using the filtering method of Eisner (1996).

Such parsers have been shown by Komagata (1999) to be of roughly cubic observed time complexity in the length of the sentence for reasonable-sized hand-built grammars. Hockenmaier, Bierner and Baldridge (2004) demonstrate their practicality as a basis for large-scale grammars induced from corpora. White (2006) and Espinosa, White and Mehay (2008) extend this approach to perform efficient wide-coverage sentence realization with such grammars.

Cubic time costs are still prohibitive for really large volume parsing and unrealistic as a model of the human parser, which appears to be linear time or better. For large volume parsing of text corpora, statistical optimization techniques integrating probabilistic head-dependencies with competence-based grammar of the kind proposed by Collins (1999) and Charniak, Goldwater and Johnson (1998) are particularly well-adapted to CCG parsing. Clark (2002), Clark, Hockenmaier and Steedman (2002), Clark and Curran (2007), Hockenmaier and Steedman (2002a), Hockenmaier and Steedman (2002b), Hockenmaier (2003a,b), and Gildea and Hockenmaier (2003) show that statistically optimized CCG parsers give rates of dependency recovery that are as good overall as state-of-the-art treebank parsers, and do better on recovering long-range dependencies.

For modeling human parsing, there is every indication that something even more restrictive is needed. Bever's (1970) observation that naive subjects typically fail to find any grammatical analysis at all for "garden path" sentences like (113a) shows that the human processor is "incomplete":

(113) a. The doctor sent for the patient arrived.
b. The flowers sent for the patient arrived.

The fact that (as Bever also noticed) the same subjects typically judge the isomorphic sentence (113b) grammatical suggests that the human sentence processor prunes the search space on the basis either of the relative likelihood of noun phrases like *the doctor* or *the flower* being dependent in relations like subject or object on verbs like *send for*, or the relative likelihood of the various logical forms corresponding to entire prefixes such as *the flowers/doctor sent for* in a particular context. In the case of (113a) this will cause the only analysis compatible with the rest of the sentence to be rejected, causing the garden path. Crain and Steedman (1985) and Altmann and Steedman (1988) showed that manipulating the context for related sentences in such a way as to pragmatically support the modifier reading eliminates the classic garden path effect. This fact suggests that the latter alternative is at work, rather than (or perhaps as well as) the former purely statistical mechanisms.

These authors proposed a modification of the basic parser according to which each word was processed in a left-to-right traversal of the sentence and rival analyses developed in parallel could more or less immediately be pruned under a "weak" or "filtering" interaction with an incrementally assembled semantic interpretation, restricted to sending an interrupt to any syntactic analysis whose yield was unlikely or implausible.[39] [40]

However, in terms of traditional grammar, both probabilistic and weak semantically interactive interpretations of the plausibility effect on garden paths present a problem for SCH. If the parser is to take account of the incompatibility of *flowers* and the subject slot of the tensed verb reading of *sent for*, this information must become available before *the patient* is integrated. (Otherwise the processor would be able to "see" the incompatible verb *arrived*, and avoid the garden path in (113a).)

This means that the parser must implicitly or explicitly have access to the interpretation or partial structure corresponding to the prefix *The flowers sent for* . . . . But this substring is not a legal constituent according to standard grammars. So SCH appears to be breached: the parser has built or thought about building a relation that the grammar does not recognize via constituency.

This may not seem to be a very serious problem in English, where the subject and verb are immediately adjacent and could be related by other means, albeit in violation of SCH. However in verb final languages characterized by

---

[39] This form of incrementality is weaker than those proposed by Hausser (1986) and Phillips (1996, 2003), since it is limited by "islands" such as right-adjuncts, into which composition cannot take place. Hausser's and Phillips notion of incrementality is by contrast strictly word-by-word. (Phillips' method of incremental structure-building is in addition nonmonotonic.)

[40] There is a misleading tendency in the literature to refer to the above theory as the "referential" theory of disambiguation, and to claim that evidence of other incremental semantic effects on parsing contradicts this theory (Sedivy and Spivey-Knowlton 1993; Spivey-Knowlton and Sedivy 1995; Tanenhaus and Trueswell 1995). However, the incremental semantic interaction that Crain and Steedman (1985) and Altmann and Steedman (1988) propose under these principles clearly involves all aspects of meaning that contribute to semantic plausibility — referential, sense-semantic, and knowledge-based. It should also be noted that probability as reflected in statistical models used in computational linguistics represents a mixture of semantic and knowledge-based relations bearing on plausibility, of very much the kind that these authors call for. Incrementality of this nature is already standard in computational applications: for example, Kruijff et al. (2007) discuss a robotic dialogue system that uses an incremental CKY parser with contextual disambiguation for comprehending situated dialogue.

constructions like the Dutch example (2), in which arbitrarily many arguments can be separated from their verbs by long-distance dependencies, similar effects are much more problematic, in effect requiring the parser to have sophisticated predictive mechanisms and to build explicit or implicit partial structures corresponding to non-constituent fragments.

Dutch, German and Japanese native speakers greet with hilarity the suggestion that their languages prohibit any analysis until the verb group (in the Dutch bare infinitival construction, the *entire* verb group) has been processed. Moreover, there are a number of experimental results which are claimed to show effects of early syntactic commitment. In particular, Gorrell (1995b,a); Inoue and Fodor (1995); Mazuko and Itoh (1995); Sturt and Crocker (1996); Kamide and Mitchell (1999) show that Japanese speakers are committed to one analysis of an ambiguity arising from the possibility of null anaphora in complex argument sequences, as revealed by garden path effects when a verb incompatible with the preferred analysis is encountered. Konieczny et al. (1997) show a similar early committment for German. All authors relate these effects to availability of *case* information in these languages, a phenomenon whose resemblance to type-raising has already been noted.

In this connection, it is interesting that, both in the case of (113), and for the SOV language cases, the relevant prefix strings are available as non-standard constituents, complete with logical forms, under alternative CCG derivations of the kind illustrated for the SOV case in (93) and (94). CCG therefore provides everything that is needed for the parser to compare the analyses either in probabilistic or semantic/pragmatic terms under the weak-interactive theory. CCG thus allows such processors to adhere rigorously to the Strict Competence Hypothesis while maintaining incrementality, even for verb-final languages.

11 COMPUTATIONAL APPLICATIONS

The fact that CCG and its relatives are of (low) polynomial worst-cases complexity means that divide-and-conquer parsing algorithms familiar from the con. text-free case readily generalize. Statistical optimization therefore also makes minor differences in algorithmic complexity much less important than algorithmic simplicity and transparency. Head dependencies compile into the model a powerful mixture of syntactic, semantic, and world-dependent regularities that can be amazingly effective in reducing search. Using the annotated CCG derivations and associated word-word dependencies available in CCGbank (Hockenmaier, 2006; Hockenmaier and Steedman, 2007), recent work has built wide-coverage, robust parsers with state of the art performance (Hockenmaier and Steedman, 2002b; Hockenmaier, 2003b; Clark and Curran, 2004, 2007) . Birch, Osborne and Koehn (2007) and Hassan, Sima'an and Way (2009) use CCG categories and parsers as models for statistical machine translation.

The OpenCCG system[41] supports (multi-modal) CCG grammar development and performs both sentence parsing and realization; it has also been used for a wide-range of dialog systems—see the discussion in Baldridge et al. (2007) regarding OpenCCG grammar development and applications and White (2006) on efficient realization with OpenCCG. This work has been connected to CCGbank to bootstrap a grammar for use with OpenCCG that supports wide-coverage sentence realization (Espinosa, White and Mehay, 2008).

Villavicencio (2002) and Zettlemoyer and Collins (2005) have exploited the semantic transparency of CCG to model grammar induction from pairs of strings and logical forms, while Piantadosi et al. (2008) use CCG to model acquisition of quantifier semantics. Indeed, the main current obstacle to further progress is the lack of labeled data for inducing bigger lexicons and models. Supertagging models that use grammar-informed initialization and priors based on CCG's categories and rules may help reduce the amount of human annotated data required to create large lexicons for new languages and domains (Baldridge, 2008).

12 CONCLUSION

Because of its very literal-minded adherence to the constituent condition on rules, and the consequent introduction of composition and type-raising, which project directionality specifications and other information from the lexicon subject to principles of slash inheritance and consistency, Combinatory Categorial Grammar abandons traditional notions of surface constituency in favor of "flexible" surface structure, in which most contiguous substrings of a grammatical sentence are potential constituents, complete with a compositional semantic interpretation, for the purposes of the application of grammatical rules.

The benefits of this move are the following.

---

[41] http:openccg.sourceforge.net

1. Coordination, Parentheticalization, and Intonation Structure can all be handled with the same apparatus that is required for “*wh*-movement” constructions such as relativization, using purely type-driven syntactic rules that strictly adhere to the Constituent Condition on Rules.

2. The rules of syntax are universal and invariant; lexical control over their applicability allows the more powerful rules to be used in the contexts where they are needed while keeping them from causing overgeneration elsewhere.

3. Everything that depends on relations of “c-command” (e.g. binding and control, quantifier scope) must be dealt with at the level of logical form (cf. Bach 1980; Lasnik and Saito 1984), with a consequent transfer of responsibility for the grammar of bounded constructions to the lexicon.

4. The modules of Phonological Form, S-Structure, and Intonational Structure are unified into a single surface derivational module.

5. Efficient processing including weakly semantically interactive incremental parsing remains possible and is compatible with rigorous observation of the Strict Competence Hypothesis, even for head-final languages.

6. Standard techniques for obtaining wide coverage computational parsers and statistical parsing models can be applied.

In respect of the last point, in eliminating all intervening modules between phonetic or phonological form, CCG is in broad accord with the principles of the Minimalist Program, advocated by Chomsky (1993, 1995) in recent years, and in particular the version proposed by Epstein et al. (1998) (cf. Kitahara 1995), in which it is proposed to equate Chomsky’s operations MERGE and MOVE as a single operation. To the extent that both relativization (and other so-called movements) and in-situ argument reduction are effected in CCG by the same type-driven operation of functional application, it can be seen as formalizing this idea, and extending it to cover DELETE. However it should be noted that in other respects the frameworks are quite different. In particular, the meaning of the term “derivation” as used by Epstein et al. is quite different from the sense of that term used here and in *SP*.

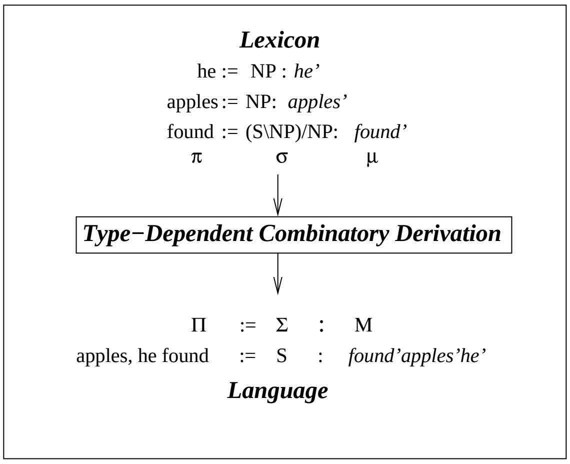


Figure 2: Generative Architecture of CCG

When viewed in the accepting or recognizing direction, the combinatory rules map strings of lexical items onto combinatory derivations. Because lexical items and combinatory rules are semantically compositional under the Type-Transparency Principles of CCG, such derivations are guaranteed to deliver logical forms surface-compositionally, without the mediation of any independent derivational machinery. *SP* and Steedman 2007 show that this generalization extends to the "covert" variety of movement that has been invoked to explain the possibility of quantifier scope alternation. Certain desirable consequences also follow for efficient processing (Clark, Hockenmaier and Steedman 2002; Hockenmaier and Steedman 2002b; Hockenmaier 2003a; Clark and Curran 2004).

When viewed as a generative grammar, the architecture of the theory that ensues can be summarized as in figure 2, replacing the standard T- or Y- diagram.

According to this architecture, the lexicon pairs words ϕ with categories consisting of a syntactic type σ and a logical form λ. Universal grammar defines the possible directional type(s) σ for any semantic type λ_τ in a given language. The combinatory rules, rules from a set which is also universally specified, subject to the Principles of Adjacency, Consistency, and Inheritance set out in section (4), then projects the lexicon onto the language, which consists of phonological strings Φ paired with a syntactic start symbol ∑ of the grammar, such as S, paired with a logical form Λ. The syntactic projection including the processes responsible for relativization, coordination, parentheticalization and intonation structure, is accomplished by pure combinatory reduction—that is, by simple merger of adjacent constituents by type-driven combinatory rules, without structure-dependent syntactic operations corresponding to MOVE or DELETE.

REFERENCES

Abeille, Anne, and Yves Schabes. 1989. Parsing idioms with a lexicalized tree adjoining grammar. In *Proceedings of the 4th International Meeting of the European Association for Computational Linguistics*, 1–9. ACL.

Ades, Anthony, and Mark Steedman. 1982. On the order of words. *Linguistics and Philosophy*, 4, 517–558.

Aho, Alfred. 1968. Indexed grammars—an extension of context-free grammars. *Communications of the Association for Computing Machinery*, 15, 647–671.

Ajdukiewicz, Kazimierz. 1935. Die syntaktische konnexität. In Storrs McCall, ed., *Polish Logic 1920-1939*, 207–231. Oxford: Oxford University Press. Translated from *Studia Philosophica*, 1, 1-27.

Altmann, Gerry, and Mark Steedman. 1988. Interaction with context during human sentence processing. *Cognition*, 30, 191–238.

Asudeh, Ash, and Richard Crouch. 2002. Coordination and parallelism in glue semantics: Integrating discourse cohesion and the element constraint. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG02 Conference*. Stanford CA: CSLI Publications.

Bach, Emmon. 1979. Control in Montague grammar. *Linguistic Inquiry*, 10, 513–531.

Bach, Emmon. 1980. In defense of passive. *Linguistics and Philosophy*, 3, 297–341.

Bach, Emmon. 1983. Generalized categorial grammars and the English auxiliary. In Frank Heny and Barry Richards, eds., *Linguistic Categories: Auxiliaries and Related Puzzles, II*, 101–120. Dordrecht: Reidel.

Baldridge, Jason. 2002. *Lexically Specified Derivational Control in Combinatory Categorial Grammar*. Ph.D. thesis, University of Edinburgh.

Baldridge, Jason. 2008. Weakly supervised supertagging with grammar-informed initialization. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, 57–64. Manchester, UK: Coling 2008 Organizing Committee. URL http://www.aclweb.org/anthology/C08-1008.

Baldridge, Jason, Sudipta Chatterjee, Alexis Palmer, and Ben Wing. 2007. DotCCG and VisCCG: Wiki and programming paradigms for improved grammar engineering with OpenCCG. In *Proceedings of the Workshop on Grammar Engineering Across Frameworks*. Stanford, CA.

Baldridge, Jason, and Geert-Jan Kruijff. 2002. Coupling CCG and hybrid logic dependency semantics. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, 319–326. Philadelphia, Pennsylvania.

Baldridge, Jason, and Geert-Jan Kruijff. 2003. Multi-modal combinatory categorial grammar. In *Proceedings of 11th Annual Meeting of the European Association for Computational Linguistics*, 211–218. Budapest.

Bar-Hillel, Yehoshua. 1953. A quasi-arithmetical notation for syntactic description. *Language*, 29, 47–58.

Bar-Hillel, Yehoshua, Chaim Gaifman, and Eliyahu Shamir. 1964. On categorial and phrase structure grammars. In Yehoshua Bar-Hillel, ed., *Language and Information*, 99–115. Reading, MA: Addison-Wesley.

Bever, Thomas. 1970. The cognitive basis for linguistic structures. In John Hayes, ed., *Cognition and the Development of Language*, 279–362. New York: Wiley.

Birch, Alexandra, Miles Osborne, and Philipp Koehn. 2007. Ccg supertags in factored translation models. In *Proceedings of the 2nd Workshop on Statistical Machine Translation*, 9–16. ACL.

Bolinger, Dwight. 1989. *Intonation and Its Uses*. Stanford, CA: Stanford University Press.

Bozsahin, Cem. 2000. Directionality and the lexicon: Evidence from gapping. Ms. Middle East Technical University, Ankara.

Brame, Michael. 1978. *Base Generated Syntax*. Seattle, WA: Noit Amrofer.

Bresnan, Joan. 1972. *Theory of Complementation in English Syntax*. Ph.D. thesis, MIT.

Bresnan, Joan, ed. 1982. *The Mental Representation of Grammatical Relations*. Cambridge, MA: MIT Press.

Charniak, Eugene, Sharon Goldwater, and Mark Johnson. 1998. Edge-based best-first chart parsing. In *Proceedings of the 6th Workshop on Very Large Corpora, Montreal, August*, 127–133.

Chomsky, Noam. 1957. *Syntactic Structures*. The Hague: Mouton.

Chomsky, Noam. 1981. *Lectures on Government and Binding*. Dordrecht: Foris.

Chomsky, Noam. 1993. A minimalist program for linguistic theory. In Kenneth Hale and Samuel Jay Keyser, eds., *The View from Building 20*, 1–52. Cambridge, MA: MIT Press.

Chomsky, Noam. 1995. *The Minimalist Program*. Cambridge, MA: MIT Press.

Clark, Stephen. 1997. *Binding and Control in Categorial Grammar*. Master's thesis, University of Manchester.

Clark, Stephen. 2002. A supertagger for combinatory categorial grammar. In *Proceedings of the TAG+ Workshop*, 19–24. Venice.

Clark, Stephen, and James R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In *Proceedings of the 42nd Meeting of the ACL*, 104–111. Barcelona, Spain.

Clark, Stephen, and James R. Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics*, 33, 4.

Clark, Stephen, Julia Hockenmaier, and Mark Steedman. 2002. Building deep dependency structures with a wide-coverage CCG parser. In *Proceedings of the 40th Meeting of the ACL*, 327–334. Philadelphia, PA.

Collins, Michael. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Copestake, Ann, Alex Lascarides, and Dan Flickinger. 2001. An algebra for semantic construction in constraint-based grammars. In *Proceedings of the 39th Annual Meeting of the Association of Computational Linguistics*, 132–139. Toulouse, France.

Crain, Stephen, and Mark Steedman. 1985. On not being led up the garden path: the use of context by the psychological parser. In Lauri Kartunnen David Dowty and Arnold Zwicky, eds., *Natural Language Parsing: Psychological, Computational and Theoretical Perspectives*, 320–358. Cambridge: Cambridge University Press.

Cresswell, Max. 1985. *Structured Meanings*. Cambridge MA: MIT Press.

Cresswell, M.J. 1973. *Logics and Languages*. London: Methuen.

Croft, William. 2001. *Radical Construction Grammar: Syntactic Theory in Typological Perspective*. Oxford: Oxford University Press.

Curry, Haskell B., and Robert Feys. 1958. *Combinatory Logic: Vol. I*. Amsterdam: North Holland.

Dowty, David. 1979. Dative movement and Thomason's extensions of Montague grammar. In Steven Davis and Marianne Mithun, eds., *Linguistics, Philosophy, and Montague Grammar*, 153–222. Austin, TX: University of Texas Press.

Dowty, David. 1988. Type-raising, functional composition, and nonconstituent coordination. In Richard T. Oehrle, Emmon Bach, and Deirdre Wheeler, eds., *Categorial Grammars and Natural Language Structures*, 153–198. Dordrecht: Reidel.

Dowty, David. 1997. Nonconstituent coordination, wrapping, and multimodal categorial grammars: Syntactic form as logical form. In Maria Luisa Dalla Chiara, ed., *Proceedings of the Tenth International Congress of Logic, Methodology, and Philosophy of Science, 1995*, 347–368. Amsterdam: North Holland. Extended version at `http://www.ling.ohio-state.edu/~dowty/`.

Eisner, Jason. 1996. Efficient normal-form parsing for combinatory categorial grammar. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, Santa Cruz, CA*. San Francisco, CA: Morgan Kaufmann.

Enrico, John, and Jason Baldridge. ???? Possessor raising, demonstrative raising, quantifier float and number float in Haida. *International Journal of American Linguistics*. To appear.

Epstein, Samuel D., Erich Groat, Ruriko Kawashima, and Hisatsugu Kitahara. 1998. *A Derivational Approach to Syntactic Relations*. Oxford: Oxford University Press.

Espinosa, Dominic, Michael White, and Dennis Mehay. 2008. Hypertagging: Supertagging for surface realization with ccg. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-08: HLT)*, 183–191. Columbus, OH.

Firbas, Jan. 1964. On defining the theme in functional sentence analysis. *Travaux Linguistiques de Prague*, 1, 267–280.

Firbas, Jan. 1966. Non-thematic subjects in contemporary english. *Travaux Linguistiques de Prague*, 2, 229–236.

Foster, John. 1990. *A Theory of Word-Order in Categorial Grammar, with Special Reference to Spanish*. Ph.D. thesis, University of York.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry*, 12, 155–184.

Gazdar, Gerald. 1988. Applicability of indexed grammars to natural languages. In Uwe Reyle and Christian Rohrer, eds., *Natural Language Parsing and Linguistic Theories*, 69–94. Dordrecht: Reidel.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum, and Ivan Sag. 1985. *Generalized Phrase Structure Grammar*. Oxford: Blackwell.

Gibson, James. 1966. *The Senses Considered as Perceptual Systems*. Boston, MA: Houghton-Mifflin Co.

Gildea, Dan, and Julia Hockenmaier. 2003. Identifying semantic roles using combinatory categorial grammar. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, 57–64. Sapporo, Japan.

Goldberg, Adèle. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago, IL: Chicago University Press.

Goldberg, Adèle. 2006. *Constructions at Work*. Oxford: Oxford University Press.

Goldsmith, John. 1985. A principled exception to the coordinate structure constraint. In *Proceedings of the 21st Regional Meeting of the Chicago Linguistic Society*. Chicago: CLS, University of Chicago.

Gorrell, Paul. 1995a. Japanese trees and the garden path. In Reiko Mazuka and Noriko Nagai, eds., *Japanese Sentence Processing*, 331–350. Hillsdale, NJ: Erlbaum.

Gorrell, Paul. 1995b. *Syntax and Parsing*. Cambridge: Cambridge University Press.

Harrison, Michael. 1978. *Introduction to Formal Language Theory*. Reading MA: Addison-Wesley.

Hassan, Hany, Khalil Sima'an, and Andy Way. 2009. A syntactified direct translation model with linear-time decoding. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 1182–1191. Singapore: Association for Computational Linguistics.

Hausser, Roland. 1986. *NEWCAT*. Berlin: Springer-Verlag.

Hepple, Mark. 1990. *The Grammar and Processing of Order and Dependency: A Categorial Approach*. Ph.D. thesis, University of Edinburgh.

Hockenmaier, Julia. 2003a. *Data and models for statistical parsing with CCG*. Ph.D. thesis, School of Informatics, University of Edinburgh.

Hockenmaier, Julia. 2003b. Parsing with generative models of predicate-argument structure. In *Proceedings of the 41st Meeting of the ACL, Sapporo*, 359–366. San Francisco: Morgan-Kaufmann.

Hockenmaier, Julia. 2006. Creating a ccgbank and a wide-coverage ccg lexicon for german. In *Proceedings of the 44th Annual Meeting of the ACL, Sydney*, 505–512. New Brunswick: ACL.

Hockenmaier, Julia, Gann Bierner, and Jason Baldridge. 2004. Extending the coverage of a CCG system. *Research in Logic and Computation*, 2, 165–208.

Hockenmaier, Julia, and Mark Steedman. 2002a. Acquiring compact lexicalized grammars from a cleaner treebank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, 1974–1981. Las Palmas, Spain.

Hockenmaier, Julia, and Mark Steedman. 2002b. Generative models for statistical parsing with Combinatory Categorial Grammar. In *Proceedings of the 40th Meeting of the ACL*, 335–342. Philadelphia, PA.

Hockenmaier, Julia, and Mark Steedman. 2007. CCGbank: a corpus of CCG derivations and dependency structures extracted from the Penn Treebank. *Computational Linguistics*. 355-396.

Hockenmaier, Julia, and Peter Young. 2008. Non-local scrambling: the equivalence of TAG and CCG revisited. In *Proceedings of The Ninth International Workshop on Tree Adjoining Grammars and Related Formalisms (TAG+9)*. Tübingen, Germany.

Hoffman, Beryl. 1995. *Computational Analysis of the Syntax and Interpretation of "Free" Word-Order in Turkish*. Ph.D. thesis, University of Pennsylvania. IRCS Report 95-17.

Hoyt, Frederick, and Jason Baldridge. 2008. A logical basis for the D combinator and normal form in CCG. In *Proceedings of ACL-08: HLT*, 326–334. Columbus, Ohio: Association for Computational Linguistics.

Huddleston, Rodney, and Geoffrey K. Pullum. 2002. *Cambridge Grammar of English*. Cambridge: Cambridge University Press.

Huybregts, Riny. 1976. Overlapping dependencies in Dutch. *Utrecht Working Papers in Linguistics*, 1, 24–65.

Huybregts, Riny. 1984. The weak inadequacy of context-free phrase-structure grammars. In Ger de Haan, Mieke Trommelen, and Wim Zonneveld, eds., *Van Periferie naar Kern*. Dordrecht: Foris.

Inoue, A., and Janet Dean Fodor. 1995. Information-paced parsing of Japanese. In R. Masuka and N. Nagai, eds., *Japanese Sentence Processing*. Hillsdale, NJ: Erlbaum.

Jackendoff, Ray. 1972. *Semantic Interpretation in Generative Grammar*. Cambridge, MA: MIT Press.

Jackendoff, Ray. 1990. *Semantic Structures*. Cambridge: MIT Press.

Jacobson, Pauline. 1990. Raising as function composition. *Linguistics and Philosophy*, 13, 423–476.

Jacobson, Pauline. 1992a. Flexible categorial grammars: Questions and prospects. In Robert Levine, ed., *Formal Grammar*, 129–167. Oxford: Oxford University Press.

Jacobson, Pauline. 1992b. The lexical entailment theory of control and the tough construction. In Ivan Sag and Anna Szabolcsi, eds., *Lexical Matters*, 269–300. Stanford, CA: CSLI Publications.

Joshi, Aravind. 1988. Tree adjoining grammars. In David Dowty, Lauri Karttunen, and Arnold Zwicky, eds., *Natural Language Parsing*, 206–250. Cambridge: Cambridge University Press.

Joshi, Aravind, Owen Rambow, and Tilman Becker. 2000. Complexity of scrambling—a new twist to the competence-performance distinction. In Anne Abeille and Owen Rambow, eds., *Tree Adjoining Grammars: Formalisms, Linguistic Analysis and Processing*. Stanford, CA: CSLI Publications.

Kamide, Yuki, and Don Mitchell. 1999. Incremental pre-head attachment in Japanese parsing. *Language and Cognitive Processes*, 11, 631–662.

Karamanis, Nikiforos. 2000. Ordered set-CCG. Ms. University of Edinburgh.

Karlgren, Hans. 1974. *Categorial Grammar Calculus*. No. 2 in Kval publ. Stockholm: Språkförlaget Skriptor AB.

Karttunen, Lauri. 1989. Radical lexicalism. In Mark Baltin and Anthony Kroch, eds., *Alternative Conceptions of Phrase Structure*. Chicago: University of Chicago Press.

Kay, Paul. 2002. An informal sketch of a formal architecture for construction grammar. *Grammars*, 5, 1–19.

Kehler, Andrew. 2002. *Coherence, Reference, and the Theory of Grammar*. Stanford CA: CSLI Publications.

Kitahara, Hisatsugu. 1995. Target α: Deducing strict cyclicity from principles of derivational economy. *Linguistic Inquiry*, 26, 47–77.

Koller, Alexander, and Marco Kuhlmann. 2009. Dependency trees and the strong generative capacity of CCG. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, 460–468. Athens, Greece: Association for Computational Linguistics.

Komagata, Nobo. 1999. *Information Structure in Texts: A Computational Analysis of Contextual Appropriateness in English and Japanese*. Ph.D. thesis, University of Pennsylvania.

Konieczny, Lars, Barbara Hemforth, Christoph Scheepers, and Gerhard Strube. 1997. The role of lexical heads in parsing: Evidence from German. *Language and Cognitive Processes*, 12, 307–348.

Kruijff, Geert-Jan. 2001. *Categorial-Modal Architecture of Informativity: Dependency Grammar Logic and Information Structure*. Ph.D. thesis, Charles University, Prague.

Kruijff, Geert-Jan M., Pierre Lison, Trevor Benjamin, Henrik Jacobsson, and Nick Hawes. 2007. Incremental, multi-level processing for comprehending situated dialogue in human-robot interaction. In Luis Seabra Lopes, Tony Belpaeme, and Stephen J. Cowley, eds., *Symposium on Language and Robots (LangRo 2007)*. Aveiro, Portugal.

Lakoff, George. 1986. Frame semantic control of the coordinate structure constraint. In Anne Farley, Peter Farley, and Karl-Eric McCullough, eds., *Papers from the Parasession on Pragmatics and Grammatical Theory at the 22nd Regional Meeting of the Chicago Linguistics Society*, 152–167. Chicago: CLS.

Lambek, Joachim. 1958. The mathematics of sentence structure. *American Mathematical Monthly*, 65, 154–170.

Landsbergen, Jan. 1982. Machine translation based on logically isomorphic Montague grammars. In *Proceedings of the 9th International Conference on Computational Linguistics*, 175–181. CoLing-82, Prague, ACL.

Lasnik, Howard, and Mamoru Saito. 1984. On the nature of proper government. *Linguistic Inquiry*, 15, 235–289.

Mathesius, Vilem. 1929. Functional linguistics. In Josef Vachek, ed., *Praguiana: Some Basic and Less Well-known Aspects of the Prague Linguistics School*, 121–142. Amsterdam: John Benjamins. Publ. 1983.

Mazuko, Reika, and Kenji Itoh. 1995. Japanese trees and the garden path. In Reiko Mazuka and Noriko Nagai, eds., *Can Japanese Speakers be Led Down the Garden Path*, 295–329. Hillsdale, NJ: Erlbaum.

McCloskey, James. 1978. *A Fragment of a Grammar of Modern Irish*. Ph.D. thesis, University of Texas at Austin. Publ. as Texas Linguistic Forum, 12.

McConville, Mark. 2006. Inheritance and the ccg lexicon. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, 1–8. ACL.

McConville, Mark. 2007. *Inheritance and the Categorial Lexicon*. Ph.D. thesis, University of Edinburgh.

Montague, Richard. 1974. *Formal Philosophy: Papers of Richard Montague*. New Haven, CT: Yale University Press. Richmond H. Thomason, ed.

Moortgat, Michael. 1988. *Categorial Investigations*. Ph.D. thesis, Universiteit van Amsterdam. Published by Foris, Dordrecht, 1989.

Morrill, Glyn. 1988. *Extraction and Coordination in Phrase Structure Grammar and Categorial Grammar*. Ph.D. thesis, University of Edinburgh.

Morrill, Glyn. 1994. *Type-Logical Grammar*. Dordrecht: Kluwer.

Oehrle, Richard. 1975. *The Grammatical Status of the English Dative Alternation*. Ph.D. thesis, MIT.

Oehrle, Richard T. to appear. Multi-modal type logical grammar. In Robert Borsley and Kersti Borjars, eds., *Constraint-based approaches to grammar: alternatives to transformational syntax*. Oxford: Blackwell.

Partee, Barbara, ed. 1976. *Montague Grammar*. New York: Academic Press.

Pentus, Mati. 1993. Lambek grammars are context-free. In *Proceedings of the IEEE Symposium on Logic in Computer Science, Montreal*, 429–433.

Pentus, Mati. 2003. Lambek calculus is NP-complete. Tech. Rep. TR-2003005, Graduate Center, City University of New York.

Peters, Stanley, and Robert Ritchie. 1973. On the generative power of transformational grammars. *Information Science*, 6, 49–83.

Phillips, Colin. 1996. *Order and Structure*. Ph.D. thesis, MIT.

Phillips, Colin. 2003. Linear order and constituency. *Linguistic Inquiry*, 34, 37–90.

Piantadosi, Steven T., Noah D. Goodman, Benjamin A. Ellis, and Joshua B. Tenenbaum. 2008. A bayesian model of the acquisition of compositional semantics. In *Proceedings of the 30th Annual Meeting of the Cognitive Science Society*, 1620–1625. Cognitive Science Society.

Pickering, Martin, and Guy Barry. 1993. Dependency categorial grammar and coordination. *Linguistics*, 31, 855–902.

Pierrehumbert, Janet. 1980. *The Phonology and Phonetics of English Intonation*. Ph.D. thesis, MIT. Distributed by Indiana University Linguistics Club, Bloomington.

Pierrehumbert, Janet, and Mary Beckman. 1988. *Japanese Tone Structure*. Cambridge, MA: MIT Press.

Pollard, Carl. 1984. *Generalized Phrase Structure Grammars, Head Grammars, and Natural Languages*. Ph.D. thesis, Stanford University.

Pollard, Carl, and Ivan Sag. 1992. Anaphors in English and the scope of binding theory. *Linguistic Inquiry*, 23, 261–303.

Pollard, Carl, and Ivan Sag. 1994. *Head Driven Phrase Structure Grammar*. Stanford, CA: CSLI Publications.

Prevost, Scott. 1995. *A Semantics of Contrast and Information Structure for Specifying Intonation in Spoken Language Generation*. Ph.D. thesis, University of Pennsylvania.

Prevost, Scott, and Mark Steedman. 1994. Specifying intonation from context for speech synthesis. *Speech Communication*, 15, 139–153.

Rambow, Owen. 1994. *Formal and Computational Aspects of Natural Language Syntax*. Ph.D. thesis, University of Pennsylvania.

Reinhart, Tanya, and Eric Reuland. 1991. Anaphors and logophors: An argument structure perspective. In Jan Koster and Eric Reuland, eds., *Long Distance Anaphora*, 283–322. Cambridge: Cambridge University Press.

Reinhart, Tanya, and Eric Reuland. 1993. Reflexivity. *Linguistic Inquiry*, 24, 657–720.

Rooth, Mats. 1985. *Association with Focus*. Ph.D. thesis, University of Massachusetts, Amherst.

Rooth, Mats. 1992. A theory of focus interpretation. *Natural Language Semantics*, 1, 75–116.

Rosenbaum, Harvey. 1977. Zapotec gapping as counterevidence to some universal proposals. *Linguistic Inquiry*, 8, 379–395.

Ross, John Robert. 1967. *Constraints on Variables in Syntax*. Ph.D. thesis, MIT. Published as *Infinite Syntax!*, Ablex, Norton, NJ, 1986.

Ross, John Robert. 1970. Gapping and the order of constituents. In Manfred Bierwisch and Karl Heidolph, eds., *Progress in Linguistics*, 249–259. The Hague: Mouton.

Sag, Ivan, and Carl Pollard. 1991. An integrated theory of control. *Language*, 67, 63–113.

Sedivy, Julie, and Michael Spivey-Knowlton. 1993. The effect of NP definiteness on parsing attachment ambiguity. In *NELS 23*. Amherst: University of Massachusetts, GLSA.

Shieber, Stuart. 1985. Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8, 333–343.

Shieber, Stuart. 1986. *An Introduction to Unification-Based Approaches to Grammar*. Stanford, CA: CSLI Publications.

Spivey-Knowlton, Michael, and Julie Sedivy. 1995. Resolving attachment ambiguities with multiple constraints. *Cognition*, 55, 227–267.

von Stechow, Arnim. 1991. Focusing and backgrounding operators. In Werner Abraham, ed., *Discourse Particles: Descriptive and Theoretical Investigations on the Logical, Syntactic and Pragmatic Properties of Discourse Particles in German*, 37–84. Amsterdam: John Benjamins.

Steedman, Mark. 1985. Dependency and coordination in the grammar of Dutch and English. *Language*, 61, 523–568.

Steedman, Mark. 1987. Combinatory grammars and parasitic gaps. *Natural Language and Linguistic Theory*, 5, 403–439.

Steedman, Mark. 1990. Gapping as constituent coordination. *Linguistics and Philosophy*, 13, 207–263.

Steedman, Mark. 1991. Structure and intonation. *Language*, 67, 262–296.

Steedman, Mark. 1996. *Surface Structure and Interpretation*. No. 30 in Linguistic Inquiry Monographs. Cambridge, MA: MIT Press.

Steedman, Mark. 2000a. Information structure and the syntax-phonology interface. *Linguistic Inquiry*, 34, 649–689.

Steedman, Mark. 2000b. *The Syntactic Process*. Cambridge, MA: MIT Press.

Steedman, Mark. 2002. Plans, affordances, and combinatory grammar. *Linguistics and Philosophy*, 25, 723–753.

Steedman, Mark. 2007. On “the computation”. In Gillian Ramchand and Charles Reiss, eds., *The Oxford Handbook of Linguistic Interfaces*, 575–611. Oxford.

Sturt, Patrick, and Matthew Crocker. 1996. Monotonic syntactic processing: a cross-linguistic study. *Language and Cognitive Processes*, 11, 449–494.

Szabolcsi, Anna. 1987. On combinatory categorial grammar. In *Proceedings of the Symposium on Logic and Language, Debrecen*, 151–162. Budapest: Akadémiai Kiadó.

Szabolcsi, Anna. 1989. Bound variables in syntax: Are there any? In Renate Bartsch, Johan van Benthem, and Peter van Emde Boas, eds., *Semantics and Contextual Expression*, 295–318. Dordrecht: Foris.

Tanenhaus, Michael. to appear. Sentence processing. In Robert Borsley and Kersti Borjars, eds., *Constraint-based approaches to grammar: alternatives to transformational syntax*. Oxford: Blackwell.

Tanenhaus, Michael, and John Trueswell. 1995. Sentence comprehension. In Jim Miller and Peter Eimas, eds., *Speech, Language and Communication*, vol. 11 of *Handbook of Perception and Cognition*, 217–262. New York: Academic Press.

Trechsel, Frank. 2000. A CCG account of Tzotzil pied piping. *Natural Language and Linguistic Theory*, 18, 611–663.

Vijay-Shanker, K., and David Weir. 1990. Polynomial time parsing of combinatory categorial grammars. In *Proceedings of the 28th Annual Meeting of the Association for Computational Linguistics, Pittsburgh*, 1–8. San Francisco, CA: Morgan Kaufmann.

Vijay-Shanker, K., and David Weir. 1993. Parsing some constrained grammar formalisms. *Computational Linguistics*, 19, 591–636.

Vijay-Shanker, K., and David Weir. 1994. The equivalence of four extensions of context-free grammar. *Mathematical Systems Theory*, 27, 511–546.

Villavicencio, Aline. 2002. *The Acquisition of a Unification-Based Generalised Categorial Grammar*. Ph.D. thesis, University of Cambridge.

White, Michael. 2006. Efficient realization of coordinate structures in Combinatory Categorial Grammar. *Research on Language and Computation*, 4, 1, 39–75.

White, Michael, and Jason Baldridge. 2003. Adapting chart realization to CCG. In *Proceedings of the 9th European Workshop on Natural Language Generation, held in conjunction with the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Budapest, April 2003*. San Francisco, CA: Morgan Kaufmann. To appear.

Williams, Edwin. 1978. Across-the-board rule application. *Linguistic Inquiry*, 9, 31–43.

Wood, Mary McGee. 1993. *Categorial Grammar*. London: Routledge.

Zeevat, Henk. 1988. Combining categorial grammar and unification. In Uwe Reyle and Christian Rohrer, eds., *Natural Language Parsing and Linguistic Theories*, 202–229. Dordrecht: Reidel.

Zettlemoyer, Luke, and Michael Collins. 2005. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars. In *Proceedings of the 21st Conference on Uncertainty in AI (UAI)*, 658–666. ACL.