Chapter 31

HPSG and Categorial Grammar

Yusuke Kubota
National Institute for Japanese Language and Linguistics

This chapter aims to offer an up-to-date comparison of HPSG and Categorial Grammar (CG). Since the CG research itself consists of two major types of approaches with overlapping but distinct goals and research strategies, I start by giving an overview of these two variants of CG (Section 2). This is followed by a comparison of HPSG and CG at a broad level, in terms of the general architecture of the theory (Section 3), and then, by a more focused comparison of specific linguistic analyses of some selected phenomena (Section 4). The chapter ends by briefly touching on issues related to computational implementation and human sentence processing (Section 5). Thoughtout the discussion, I attempt to highlight both the similarities and differences between HPSG and CG research, in the hope of stimulating further research in the two research communities on their respective open questions, and so that the two communities can continue to learn from each other.

1 Introduction

The goal of this chapter is to provide a comparison between HPSG and *Categorial Grammar* (CG). The two theories share certain important insights, mostly due to the fact that they are among the so-called "lexicalist", "non-transformational" theories of syntax that were proposed as major alternatives to the mainstream transformational syntax in the 1980s (see Borsley & Börjars 2011 and Müller 2019 for overviews of these theories). However, due to the differences in the main research goals in the respective communities in which these approaches have been developed, there are certain nontrivial differences between them as well. The present chapter assumes researchers working in HPSG or other non-CG theories of syntax as its main audience, and aims to inform them of key aspects of CG

Yusuke Kubota. 2020. HPSG and Categorial Grammar. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 967–1010. Berlin: Language Science Press. DOI:??

which make it distinct from other theories of syntax. While computational implementation and investigations of the formal properties of grammatical theory have been important in both HPSG and CG research, I will primarily focus on the linguistic aspects in the ensuing discussion, with pointers (where relevant) to literature on mathematical and computational issues. Throughout the discussion, I presuppose basic familiarity with HPSG (with pointers to relevant chapters in the handbook). The present handbook contains chapters that compare HPSG with other grammatical theories, including the present one. I encourage the reader to take a look at the other theory comparison chapters too (as well as other chapters dealing with specific aspects of HPSG in greater detail), in order to obtain a fuller picture of the theoretical landscape in current (non-transformational) generative syntax research.

2 Two varieties of CG

CG is actually not a monolithic theory, but is a family of related approaches—or, perhaps more accurately, it is much *less of* a monolithic theory than either HPSG or Lexical Functional Grammar (LFG; Kaplan & Bresnan 1982; Bresnan et al. 2016) is. For this reason, I will start my discussion by sketching some important features of two major varieties of CG, *Combinatory Categorial Grammar* (CCG; Steedman 2000; 2012) and *Type-Logical Categorial Grammar* (TLCG; or "Type-Logical Grammar"; Morrill 1994; Moortgat 2011; Kubota & Levine 2020).[1] After presenting the "core" component of CG that is shared between the two approaches—which is commonly referred to as the *AB Grammar*—I introduce aspects of the respective approaches in which they diverge from each other.

2.1 Notation and presentation

Before getting started, some comments are in order as to the notation and the mode of presentation adopted. Two choices are made for the notation. First, CCG and TLCG traditionally adopt different notations of the slash. I stick to the TLCG notation throughout this chapter for notational consistency. Second, I present all the fragments below in the so-called *labelled deduction* notation of (Prawitz-style) natural deduction. In particular, I follow Oehrle (1994) and Morrill (1994) in the use of "term labels" in labelled deduction to encode prosodic and semantic information of linguistic expressions. This involves writing linguistic expressions

[1] For more detailed introductions to these different variants of CG, see Steedman & Baldridge (2011) (on CCG) and Oehrle (2011) (on TLCG), both included in Borsley & Börjars (2011).

as *tripartite signs*, formally, tuples of prosodic form, semantic interpretation and syntactic category (or syntactic type). Researchers familiar with HPSG should find this notation easy to read and intuitive; the idea is essentially the same as how linguistic signs are conceived of in HPSG. In the CG literature, this notation has its roots in the conception of "multidimensional" linguistic signs in earlier work by Dick Oehrle (1988). But the reader should be aware that this is *not* the standard notation in which either CCG or TLCG is typically presented.[2] Also, logically savvy readers may find this notation somewhat confusing since it (unfortunately) obscures certain aspects of CG pertaining to its logical properties. In any event, it is important to keep in mind that different notations co-exist in the CG literature (and the logic literature behind it), and that, just as in mathematics in general, different notations can be adopted for the same formal system to highlight different aspects of it in different contexts. As noted in the introduction, for the mode of presentation, the emphasis is consistently on linguistic (rather than computational or logical) aspects. Moreover, I have taken the liberty to gloss over certain minor differences among different variants of CG for the sake of presentation. The reader is therefore encouraged to consult primary sources as well, especially when details matter.

2.2 The AB grammar

I start with a simple fragment of CG called the *AB grammar*, consisting of just two syntactic rules in (1) (here, $\circ$ designates string concatenation):

(1) a. Forward Slash Elimination

$$\frac{\mathsf{a}; A/B \qquad \mathsf{b}; B}{\mathsf{a} \circ \mathsf{b}; A}\ /\mathrm{E}$$

b. Backward Slash Elimination

$$\frac{\mathsf{b}; B \qquad \mathsf{a}; B\backslash A}{\mathsf{b} \circ \mathsf{a}; A}\ \backslash\mathrm{E}$$

With the somewhat minimal lexicon in (2), the sentence *John loves Mary* can be licensed as in (3). The two slashes / and \ are used to form "complex" syntactic categories (more on this below) indicating valence information: the transitive verb *loves* is assigned the category (NP\S)/NP since it first combines with an NP to its right (i.e. the direct object) and then another NP to its left (i.e. the subject).

(2) a. john; NP

[2] CCG derivations are typically presented as upside-down parsetrees (see, for example, Steedman 2000; 2012) whereas TLCG derivations are typically presented as proofs in Gentzen sequent calculus (see, for example, Moortgat 2011; Barker & Shan 2015).

b. mary; NP
c. ran; NP\S
d. loves; (NP\S)/NP

(3)
$$\dfrac{\text{john; NP} \qquad \dfrac{\text{mary; NP} \qquad \text{loves; (NP\textbackslash S)/NP}}{\text{loves} \circ \text{mary; NP\textbackslash S}}\,/\text{E}}{\text{john} \circ \text{loves} \circ \text{mary; S}}\,\backslash\text{E}$$

At this point, this is just like the familiar PSG analysis of the following form, except that the symbol VP is replaced by NP\S:

(4) [S [NP John] [VP [V loves] [NP Mary]]]

Things will start looking more interesting as one makes the fragment more complex (and also by adding the semantics), but before doing so, I first introduce some basic assumptions, first about syntactic categories (below) and then about semantics (next section).

Syntactic categories (or *syntactic types*) are defined recursively in CG. This can be concisely written using the so-called "BNC notation" as follows:[3,4]

(5) a. BaseType := { N, NP, PP, S }
b. Type := BaseType | (Type\Type) | (Type/Type)

[3] See Section 3.3 below for the treatment of syntactic features (such as those used for agreement). I ignore this aspect for the fragment developed below for the sake of exposition. The treatment of syntactic features (or its analog) is a relatively underdeveloped aspect of CG syntax literature, as compared to HPSG research (where the whole linguistic theory is built on the basis of a theory/formalism of complex feature structures). CCG seems to assume something similar to feature unification in HPSG, though details are typically not worked out explicitly. In TLCG, there are occasional suggestions in the literature (see, for example, Morrill 1994; Pogodalla & Pompigne 2012) that syntactic features can be formalized in terms of dependent types (Martin-Löf 1984; Ranta 1994), but there is currently no in-depth study working out a theory of syntactic features along these lines.

[4] Recognizing PP as a basic type is somewhat non-standard, although there does not seem to be any consensus on what should be regarded as a (reasonably complete) set of basic syntactic types for natural language syntax.

In words, anything that is a BaseType is a Type, and any complex expression of form $A\backslash B$ or A/B where A and B are both Types is a Type. To give some examples, the following expressions are syntactic types according to the definition in (5):[5]

(6) a. S\S
b. (NP\S)/NP/NP
c. (S/(NP\S))\(S/NP)
d. ((NP\S)\(NP\S))\((NP\S)\(NP\S))

One important feature of CG is that, like HPSG, it lexicalizes the valence (or subcategorization) properties of linguistic expressions. Unlike HPSG, where this is done by a list (or set) valued syntactic feature, in CG, complex syntactic categories directly represent the combinatoric (i.e. valence) properties of lexical items. For example, lexical entries for intransitive and transitive verbs in English will look like the following (semantics is omitted here but will be supplied later):

(7) a. ran; NP\S
b. read; (NP\S)/NP
c. introduces; (NP\S)/PP/NP

(7a) says that the verb *ran* combines with its argument NP *to its left* to become an S. Likewise, (7b) says that *read* first combines with an NP *to its right* and then another NP to its left to become an S.

One point to keep in mind (though it may not seem to make much difference at this point) is that in CG, syntactic rules are thought of as logical rules and the derivations of sentences like (3) as *proofs* of the well-formedness of particular strings as sentences. From this "logical" point of view, the two slashes should really be thought of as directional variants of implication (that is, both A/B and $B\backslash A$ essentially mean '*if* there is a B, *then* there is an A'), and the two rules of Slash Elimination introduced in (1) should be thought of as directional variants of *modus ponens* ($B \rightarrow A, B \vdash A$). This analogy between natural language syntax and logic is emphasized in particular in TLCG research.

2.3 Syntax-semantics interface in CG

One attractive property of CG as a theory of natural language syntax is its straightforward syntax-semantics interface. In particular, there is a functional mapping

[5] I omit parentheses for a sequence of the same type of slash, for which disambiguation is obvious—for example, A\A\A is an abbreviation for (A\(A\A)).

from syntactic categories to semantic types.[6] For the sake of exposition, I assume an extensional fragment of Montagovian model-theoretic semantics in what follows, but it should be noted that the CG syntax is mostly neutral to the choice of the specific variant of semantic theory to go with it.[7]

Assuming the standard recursive definition of semantic types as in (8) (with basic types e for individuals and t for truth values), the function Sem (which returns, for each syntactic category given as input, its semantic type) can be defined as in (9) and (10).

(8) a. BaseSemType := { e, t }

b. SemType := BaseSemType | SemType $\rightarrow$ SemType

(9) (Base Case)

a. $\mathsf{Sem}(\mathrm{NP}) = \mathsf{Sem}(\mathrm{PP}) = e$

b. $\mathsf{Sem}(\mathrm{N}) = e \rightarrow t$

c. $\mathsf{Sem}(\mathrm{S}) = t$

(10) (Recursive Clause)

For any complex syntactic category of the form A/B (or $B\backslash A$),

$\mathsf{Sem}(A/B)$ (= $\mathsf{Sem}(B\backslash A)$) = $\mathsf{Sem}(B) \rightarrow \mathsf{Sem}(A)$

For example, $\mathsf{Sem}(\mathrm{S}/(\mathrm{NP}\backslash\mathrm{S})) = (e \rightarrow t) \rightarrow t$ (for subject position quantifier in CCG).

Syntactic rules with semantics can then be written as in (11) (where the semantic effect of these rules is *function application*) and a sample derivation with semantic annotation is given in (12).

(11) a. Forward Slash Elimination

$$\frac{\mathsf{a}; \mathcal{F}; A/B \qquad \mathsf{b}; \mathcal{G}; B}{\mathsf{a} \circ \mathsf{b}; \mathcal{F}(\mathcal{G}); A}\,/\mathrm{E}$$

b. Backward Slash Elimination

$$\frac{\mathsf{b}; \mathcal{G}; B \qquad \mathsf{a}; \mathcal{F}; B\backslash A}{\mathsf{b} \circ \mathsf{a}; \mathcal{F}(\mathcal{G}); A}\,\backslash\mathrm{E}$$

(12)

$$\frac{\mathsf{john}; \mathbf{j}; \mathrm{NP} \qquad \dfrac{\mathsf{mary}; \mathbf{m}; \mathrm{NP} \qquad \mathsf{loves}; \mathbf{love}; (\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}}{\mathsf{loves} \circ \mathsf{mary}; \mathbf{love}(\mathbf{m}); \mathrm{NP}\backslash\mathrm{S}}\,/\mathrm{E}}{\mathsf{john} \circ \mathsf{loves} \circ \mathsf{mary}; \mathbf{love}(\mathbf{m})(\mathbf{j}); \mathrm{S}}\,\backslash\mathrm{E}$$

[6] Technically, this is ensured in TLCG by the homomorphism from the syntactic type logic to the semantic type logic (the latter of which is often implicit) and the so-called Curry-Howard correspondence between proofs and terms (van Benthem 1988).

[7] See, for example, Martin (2013) and Bekki & Mineshima (2017) for recent proposals on adopting compositional variants of (hyper)intensional dynamic semantics and proof theoretic semantics, respectively, for the semantic component of CG-based theories of natural language.

A system of CG with only the Slash Elimination rules like the fragment above is called the *AB grammar*, so called because it corresponds to the earliest form of CG formulated by Ajdukiewicz (1935) and Bar-Hillel (1953).

2.4 Combinatory Categorial Grammar

2.4.1 An "ABC" fragment: AB grammar with order-preserving combinatory rules

Some more machinery is needed to do some interesting linguistic analysis. I now extend the AB fragment above by adding two types of rules: *Type-Raising* and (Harmonic) *Function Composition*. These are a subset of rules typically entertained in CCG. I call the resultant system *ABC Grammar* (AB + Function Composition).[8] Though it is an impoverished version of CCG, the ABC fragment already enables an interesting and elegant analysis of *nonconstituent coordination* (NCC), originally due to Steedman (1985) and Dowty (1988), which is essentially identical to the analysis of NCC in the current versions of both CCG and TLCG. I will then discuss the rest of the rules constituting CCG in the next section. The reason for drawing a distinction between the "ABC" fragment and (proper) CCG is just for the sake of exposition. The rules introduced in the present section have the property that they are all derivable as *theorems* in the (associative) Lambek calculus, the calculus that underlies most variants of TLCG. For this reason, separating the two sets of rules helps clarify the similarities and differences between CCG and TLCG.

The *Type Raising* and *Function Composition* rules are defined as in (13) and (14), respectively.

(13) a. Forward Function Composition

$$\frac{\mathsf{a}; \mathcal{F}; A/B \qquad \mathsf{b}; \mathcal{G}; B/C}{\mathsf{a} \circ \mathsf{b}; \lambda x.\mathcal{F}(\mathcal{G}(x)); A/C}\text{FC}$$

b. Backward Function Composition

$$\frac{\mathsf{b}; \mathcal{G}; C\backslash B \qquad \mathsf{a}; \mathcal{F}; B\backslash A}{\mathsf{b} \circ \mathsf{a}; \lambda x.\mathcal{F}(\mathcal{G}(x)); C\backslash A}\text{FC}$$

(14) a. Forward Type Raising

$$\frac{\mathsf{a}; \mathcal{F}; A}{\mathsf{a}; \lambda v.v(\mathcal{F}); B/(A\backslash B)}\text{TR}$$

b. Backward Type Raising

$$\frac{\mathsf{a}; \mathcal{F}; A}{\mathsf{a}; \lambda v.v(\mathcal{F}); (B/A)\backslash B}\text{TR}$$

The Type-Raising rules are essentially rules of "type lifting" familiar in the formal semantics literature, except that they specify the "syntactic effect" of type

[8] This is not a standard terminology, but giving a name to this fragment is convenient for the purpose of the discussion below.

lifting explicitly (such that the function-argument relation is reversed). Similarly Function Composition rules can be understood as function composition in the usual sense (as in mathematics and functional programming), except, again, that the syntactic effect is explicitly specified.

As noted by Steedman (1985), with Type Raising and Function Composition, a string of words such as *John loves* can be analyzed as a constituent of type S/NP, that is, an expression that is looking for an NP to its right to become an S:[9]

(15)
$$\dfrac{\dfrac{\textsf{john}; \mathbf{j}; \mathrm{NP}}{\textsf{john}; \lambda f.f(\mathbf{j}); \mathrm{S/(NP\backslash S)}}\,\mathrm{TR} \qquad \textsf{loves}; \mathbf{love}; \mathrm{(NP\backslash S)/NP}}{\textsf{john} \circ \textsf{loves}; \lambda x.\mathbf{love}(x)(\mathbf{j}); \mathrm{S/NP}}\,\mathrm{FC}$$

Intuitively, Function Composition has the effect of delaying the application of a function. The verb is looking for a direct object to its right before it can be taken as an argument (of type NP\S) of the type-raised subject NP. Function Composition directly combines the subject and the verb before the direct object argument of the latter is saturated. The resultant category inherits the unsaturated argument both in the syntactic category (of type NP\S) and semantics (of type $e \to t$).

Assuming generalized conjunction (with the standard definition for the generalized conjunction operator $\sqcap$ *à la* Partee & Rooth (1983) and the polymorphic syntactic category $(X\backslash X)/X$ for *and*), the analysis for a *right-node raising* (RNR) sentence such as (16) is straightforward, as in (17).

(16) John loves, and Bill hates, Mary.

(17)
$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \textsf{john} \circ \textsf{loves};\\ \lambda x.\mathbf{love}(x)(\mathbf{j}); \mathrm{S/NP}\end{array} \qquad \dfrac{\begin{array}{c}\textsf{and};\\ \sqcap; (X\backslash X)/X\end{array} \quad \begin{array}{c}\vdots\\ \textsf{bill} \circ \textsf{hates};\\ \lambda x.\mathbf{hate}(x)(\mathbf{b}); \mathrm{S/NP}\end{array}}{\begin{array}{c}\textsf{and} \circ \textsf{bill} \circ \textsf{hates};\\ \sqcap(\lambda x.\mathbf{hate}(x)(\mathbf{b})); \mathrm{(S/NP)\backslash(S/NP)}\end{array}}\,\mathrm{FA}}{\textsf{john} \circ \textsf{loves} \circ \textsf{and} \circ \textsf{bill} \circ \textsf{hates}; (\lambda x.\mathbf{love}(x)(\mathbf{j})) \sqcap (\lambda x.\mathbf{hate}(x)(\mathbf{b})); \mathrm{S/NP}}\,\mathrm{FA} \qquad \begin{array}{c}\textsf{mary};\\ \mathbf{m}; \mathrm{NP}\end{array}}{\textsf{john} \circ \textsf{loves} \circ \textsf{and} \circ \textsf{bill} \circ \textsf{hates} \circ \textsf{mary}; \mathbf{love}(\mathbf{m})(\mathbf{j}) \wedge \mathbf{hate}(\mathbf{m})(\mathbf{b}); \mathrm{S}}\,\mathrm{FA}$$

Dowty (1988) showed that this analysis extends straightforwardly to the (slightly) more complex case of *argument cluster coordination* (ACC), such as (18), as in (19) (here, VP, TV and DTV are abbreviations of NP\S, (NP\S)/NP and (NP\S)/NP/NP, respectively).

[9] **love** is a function of type $e \to e \to t$, where the first argument corresponds to the direct object. Thus, $\mathbf{love}(x)(y)$ is equivalent to the two-place relation notation $\mathbf{love}(y, x)$ in which the subject argument is written first.

(18) Mary gave Bill the book and John the record.

(19)

$$\dfrac{\begin{array}{l}\text{mary;}\\ \mathbf{m};\\ \mathrm{NP}\end{array} \quad \dfrac{\begin{array}{l}\text{gave;}\\ \mathbf{give};\\ \mathrm{DTV}\end{array} \quad \dfrac{\dfrac{\dfrac{\begin{array}{l}\text{bill;}\\ \mathbf{b};\mathrm{NP}\end{array}}{\begin{array}{l}\text{bill;}\\ \lambda P.P(\mathbf{b});\\ \mathrm{DTV}\backslash\mathrm{TV}\end{array}}\mathrm{TR} \quad \dfrac{\begin{array}{l}\text{the} \circ \text{book;}\\ \iota(\mathbf{bk});\mathrm{NP}\end{array}}{\begin{array}{l}\text{the} \circ \text{book;}\\ \lambda Q.Q(\iota(\mathbf{bk}));\\ \mathrm{TV}\backslash\mathrm{VP}\end{array}}\mathrm{TR}}{\begin{array}{l}\text{bill} \circ \text{the} \circ \text{book;}\\ \lambda R.R(\mathbf{b})(\iota(\mathbf{bk}));\mathrm{DTV}\backslash\mathrm{VP}\end{array}}\mathrm{FC} \quad \dfrac{\begin{array}{l}\text{and;}\\ \sqcap;\\ (X\backslash X)/X\end{array} \quad \begin{array}{l}\vdots\\ \text{john} \circ \text{the} \circ \text{record;}\\ \lambda R.R(\mathbf{j})(\iota(\mathbf{rc}));\\ \mathrm{DTV}\backslash\mathrm{VP}\end{array}}{\begin{array}{l}\text{and} \circ \text{john} \circ \text{the} \circ \text{record;}\\ \sqcap(\lambda R.R(\mathbf{j})(\iota(\mathbf{rc})));\\ (\mathrm{DTV}\backslash\mathrm{VP})\backslash(\mathrm{DTV}\backslash\mathrm{VP})\end{array}}}{\begin{array}{l}\text{bill} \circ \text{the} \circ \text{book} \circ \text{and} \circ \text{john} \circ \text{the} \circ \text{record;}\\ \lambda R.R(\mathbf{j})(\iota(\mathbf{rc})) \sqcap \lambda R.R(\mathbf{b})(\iota(\mathbf{bk}));\mathrm{DTV}\backslash\mathrm{VP}\end{array}}}{\begin{array}{l}\text{gave} \circ \text{bill} \circ \text{the} \circ \text{book} \circ \text{and} \circ \text{john} \circ \text{the} \circ \text{record;}\\ \mathbf{give}(\mathbf{j})(\iota(\mathbf{rc})) \sqcap \mathbf{give}(\mathbf{b})(\iota(\mathbf{bk}));\mathrm{VP}\end{array}}}{\begin{array}{l}\text{mary} \circ \text{gave} \circ \text{bill} \circ \text{the} \circ \text{book} \circ \text{and} \circ \text{john} \circ \text{the} \circ \text{record;}\\ \mathbf{give}(\mathbf{j})(\iota(\mathbf{rc}))(\mathbf{m}) \wedge \mathbf{give}(\mathbf{b})(\iota(\mathbf{bk}))(\mathbf{m});\mathrm{S}\end{array}}$$

Here, by Type Raising, the indirect and direct objects become functions that can be combined via Function Composition, to form a non-standard constituent that can then be coordinated. After two such expressions are conjoined, the verb is fed as an argument to return a VP. Intuitively, the idea behind this analysis is that *Bill the book* is of type DTV\VP since if it were to combine with an actual ditransitive verb (such as *gave*), a VP (*gave Bill the book*) would be obtained. Note that in both the RNR and ACC examples above, the right semantic interpretation for the whole sentence is assigned compositionally via the rules given above in (13) and (14).

2.4.2 From ABC to CCG

CCG is a version of CG developed by Mark Steedman since the 1980s with extensive linguistic application. The best sources for CCG are the three books by Steedman (Steedman 1997; 2000; 2012), which present treatments of major linguistic phenomena in CCG and give pointers to earlier literature. CCG is essentially a rule-based extension of the AB grammar. The previous section has already introduced two key components that constitute this extension: Type Raising and (Harmonic) Function Composition.[10] There are aspects of natural language syntax that cannot be handled adequately in this simple system, and in such situa-

[10]There is actually a subtle point about Type Raising rules. Recent versions of CCG (Steedman 2012) do not take them to be syntactic rules, but rather assume that Type Raising is an operation in the lexicon. This choice seems to be motivated by parsing considerations (so as to eliminate as many unary rules as possible from the syntax). It is also worth noting in this connection that the CCG-based syntactic fragment that Jacobson (1999; 2000) assumes for her Variable-Free Semantics is actually a quite different system from Steedman's version of CCG in that it

tions, CCG makes (restricted) use of additional rules. This point can be illustrated nicely with two issues that arise in connection with the analysis of long-distance dependencies.

The basic idea behind the CCG analysis of long-distance dependencies, due originally to Ades & Steedman (1982), is very simple and is similar in spirit to the HPSG analysis in terms of SLASH feature percolation (see Borsley & Crysmann 2020, Chapter 14 of this volume for the treatment of long-distance dependencies in HPSG). Specifically, CCG analyzes extraction dependencies via a chain of Function Composition, as illustrated by the derivation for (20) in (21).

(20) This is the book that John thought that Mary read _.

(21)

$$
\begin{prooftree}
\AxiomC{$\begin{array}{l}\text{that;}\\ \lambda P \lambda Q \lambda x.\\ \quad Q(x) \wedge P(x);\\ (\mathrm{N}\backslash\mathrm{N})/(\mathrm{S}/\mathrm{NP})\end{array}$}
\AxiomC{$\begin{array}{l}\text{john;}\\ \mathbf{j}; \mathrm{NP}\end{array}$}
\RightLabel{TR}
\UnaryInfC{$\begin{array}{l}\text{john;}\\ \lambda P.P(\mathbf{j});\\ \mathrm{S}/(\mathrm{NP}\backslash\mathrm{S})\end{array}$}
\AxiomC{$\begin{array}{l}\text{thought;}\\ \mathbf{think};\\ (\mathrm{NP}\backslash\mathrm{S})/\mathrm{S}'\end{array}$}
\AxiomC{$\begin{array}{l}\text{that;}\\ \lambda p.p;\\ \mathrm{S}'/\mathrm{S}\end{array}$}
\AxiomC{$\begin{array}{l}\text{read;}\\ \mathbf{read};\\ (\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}\end{array}$}
\AxiomC{$\begin{array}{l}\text{mary;}\\ \mathbf{m}; \mathrm{NP}\end{array}$}
\RightLabel{TR}
\UnaryInfC{$\begin{array}{l}\text{mary;}\\ \lambda P.P(\mathbf{m});\\ \mathrm{S}/(\mathrm{NP}\backslash\mathrm{S})\end{array}$}
\RightLabel{FC}
\BinaryInfC{$\begin{array}{l}\text{mary} \circ \text{read;}\\ \lambda x.\mathbf{read}(x)(\mathbf{m}); \mathrm{S}/\mathrm{NP}\end{array}$}
\RightLabel{FC}
\BinaryInfC{$\begin{array}{l}\text{that} \circ \text{mary} \circ \text{read;}\\ \lambda x.\mathbf{read}(x)(\mathbf{m}); \mathrm{S}'/\mathrm{S}\end{array}$}
\RightLabel{FC}
\BinaryInfC{$\begin{array}{l}\text{thought} \circ \text{that} \circ \text{mary} \circ \text{read;}\\ \lambda x.\mathbf{think}(\mathbf{read}(x)(\mathbf{m})); (\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}\end{array}$}
\RightLabel{FC}
\BinaryInfC{$\begin{array}{l}\text{john} \circ \text{thought} \circ \text{that} \circ \text{mary} \circ \text{read;}\\ \lambda x.\mathbf{think}(\mathbf{read}(x)(\mathbf{m}))(\mathbf{j}); \mathrm{S}/\mathrm{NP}\end{array}$}
\RightLabel{FA}
\BinaryInfC{$\begin{array}{l}\text{that} \circ \text{john} \circ \text{thought} \circ \text{that} \circ \text{mary} \circ \text{read;}\\ \lambda Q \lambda x.Q(x) \wedge \mathbf{think}(\mathbf{read}(x)(\mathbf{m}))(\mathbf{j}); \mathrm{N}\backslash\mathrm{N}\end{array}$}
\end{prooftree}
$$

Like (many versions of) HPSG, CCG does not assume any empty expression at the gap site. Instead, the information that the subexpressions (constituting extraction pathway) such as *Mary read* and *thought that Mary read* are missing an NP on the right edge is encoded in the syntactic category of the linguistic expression. *Mary read* is assigned the type S/NP, since it is a sentence missing an NP on its right edge. *thought that Mary read* is of type VP/NP since it is a VP missing an NP on its right edge, etc. Expressions that are not originally functions (such as the subject NPs in the higher and lower clauses inside the relative clause in (20)) are first type-raised. Then, Function Composition effectively "delays" the saturation of the object NP argument of the embedded verb, until the whole

crucially assumes Geach rules, another type of unary rules likely to have similar computational consequences as Type-Raising rules, in the syntactic component. (Incidentally, the Geach rules are often attributed to Geach 1970, but Humberstone's 2005 careful historical study suggests that this attribution is highly misleading, if not totally groundless.)

relative clause meets the relative pronoun, which itself is a higher-order function that takes a sentence missing an NP (of type S/NP) as an argument.

The successive passing of the /NP specification to larger structures is essentially analogous to the treatment of extraction via the SLASH feature in HPSG. However, unlike HPSG, which has a dedicated feature that handles this information passing, CCG achieves the effect via the ordinary slash that is also used for local syntactic composition.

This difference immediately raises some issues for the CCG analysis of extraction. First, in (20), the NP gap happens to be on the right edge of the sentence, but this is not always the case. Harmonic Function Composition alone cannot handle non-peripheral extraction of the sort found in examples such as the following:

(22) This is the book that John thought that [Mary read _ at school].

Assuming that *at school* is a VP modifier of type (NP\S)\(NP\S), what is needed here is a mechanism that assigns the type (NP\S)/NP to the string *read _ at school*, despite the fact that the missing NP is not on the right edge. CCG employs a special rule of "Crossed" Function Composition for this purpose, defined as follows:

(23) Crossed Function Composition

$$\frac{\mathsf{a}; \mathcal{G}; A/B \qquad \mathsf{b}; \mathcal{F}; A\backslash C}{\mathsf{a} \circ \mathsf{b}; \lambda x.\mathcal{F}(\mathcal{G}(x)); C/B}\ \mathrm{xFC}$$

Unlike its harmonic counterpart (in which a has the type $B\backslash A$), in (23) the directionality of the slash is different in the two premises, and the resultant category inherits the slash originally associated with the inherited argument (i.e. $/B$).

Once this non-order-preserving version of Function Composition is introduced in the grammar, the derivation for (22) is straightforward, as in (24):

(24)
$$\dfrac{\dfrac{\mathsf{mary}; \mathbf{m}; \mathrm{NP}}{\begin{array}{l}\mathsf{mary};\\ \lambda P.P(\mathbf{m});\\ \mathrm{S}/(\mathrm{NP}\backslash\mathrm{S})\end{array}}\ \mathrm{TR} \qquad \dfrac{\begin{array}{l}\mathsf{read};\\ \mathbf{read}; (\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}\end{array} \qquad \begin{array}{l}\mathsf{at} \circ \mathsf{school};\\ \textbf{at-school}; (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S})\end{array}}{\mathsf{read} \circ \mathsf{at} \circ \mathsf{school}; \lambda x.\textbf{at-school}(\mathbf{read}(x)); (\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}}\ \mathrm{xFC}}{\mathsf{mary} \circ \mathsf{read} \circ \mathsf{at} \circ \mathsf{school}; \lambda x.\textbf{at-school}(\mathbf{read}(x))(\mathbf{m}); \mathrm{S}/\mathrm{NP}}\ \mathrm{FC}$$

Unless appropriately constrained, the addition of the crossed composition rule leads to potential overgeneration, since non-extracted expressions cannot change word order so freely in English. For example, without additional restrictions, the simple CCG fragment above overgenerates examples such as the following (see, for example, Kuhlmann et al. 2015):

(25) * $\mathrm{a}_{NP/N}$ $[_{N/N}$ $\mathrm{powerful}_{N/N}$ by $\mathrm{Rivaldo}_{N\backslash N}]$ shot_{N}

Here, I will not go into the technical details of how this issue is addressed in the CCG literature. In contemporary versions of CCG, the application of special rules such as crossed composition in (23) is regulated by the notion of "structural control" borrowed into CCG from the "multi-modal" variant of TLCG (see Baldridge (2002) and Steedman & Baldridge (2011)).

Another issue that arises in connection with extraction is how to treat multiple gaps corresponding to a single filler. The simple fragment developed above cannot license examples involving *parasitic gaps* such as the following:[11]

(26)
a. This is the article that I filed _ without reading _.
b. Peter is a guy who even the best friends of _ think _ should be closely watched.

Since neither Type Raising nor Function Composition changes the number of "gaps" passed on to a larger expression, a new mechanism is needed here. Steedman (1987) proposes the following rule to deal with this issue:

(27) Substitution

$$\frac{\mathsf{a}; \mathcal{G}; A/B \qquad \mathsf{b}; \mathcal{F}; (A\backslash C)/B}{\mathsf{a} \circ \mathsf{b}; \lambda x.\mathcal{F}(x)(\mathcal{G}(x)); C/B}\ \mathrm{S}$$

This rule has the effect of "collapsing" the arguments of the two inputs into one, to be saturated by a single filler. The derivation for the adjunct parasitic gap example in (26a) then goes as follows (where VP is an abbreviation for NP\S):

(28)

$$\dfrac{\begin{array}{c}\mathsf{filed};\\ \mathbf{file}; \mathrm{VP/NP}\end{array} \qquad \dfrac{\begin{array}{c}\mathsf{without};\\ \mathbf{wo}; (\mathrm{VP\backslash VP})/\mathrm{VP}\end{array} \qquad \begin{array}{c}\mathsf{reading};\\ \mathbf{read}; \mathrm{VP/NP}\end{array}}{\begin{array}{c}\mathsf{without} \circ \mathsf{reading};\\ \lambda x.\mathbf{wo}(\mathbf{read}(x)); (\mathrm{VP\backslash VP})/\mathrm{NP}\end{array}}\ \mathrm{FC}}{\begin{array}{c}\mathsf{filed} \circ \mathsf{without} \circ \mathsf{reading};\\ \lambda x.\mathbf{wo}(\mathbf{read}(x))(\mathbf{file}(x)); \mathrm{VP/NP}\end{array}}\ \mathrm{S}$$

Like the crossed composition rule, the availability of the substitution rule should be restricted to extraction environments. In earlier versions of CCG, this

[11]Multiple gaps in coordination (i.e. ATB extraction) is not an issue, since these cases can be handled straightforwardly via the polymorphic definition of generalized conjunction in CCG, in just the same way that unsaturated shared arguments in each conjunct are identified with one another.

was done by a stipulation on the rule itself. Baldridge (2002) proposed an improvement of the organization of the CCG rule system in which the applicability of particular rules is governed by lexically specified "modality" encodings. See Steedman & Baldridge (2011) for this relatively recent development in CCG.

2.5 Type-Logical Categorial Grammar

The "rule-based" nature of CCG should be clear from the above exposition. Though superficially similar in many respects, TLCG takes a distinctly different perspective on the underlying architecture of the grammar of natural language. Specifically, in TLCG, the rule system of grammar is literally taken to be a kind of logic. Consequently, all (or almost all) grammar rules are logical inference rules reflecting the properties of (typically a small number of) logical connectives such as / and \ (which are, as noted in Section 2.2, viewed as directional variants of implication). It is important to keep in mind that this leads to an inherently much more abstract view on the organization of the grammar of natural language than the "surface-oriented" perspective that HPSG and CCG share at a broad level. This conceptual shift can be best illustrated by first replacing the ABC grammar introduced in Section 2.4.1 by the *Lambek calculus*, where all the rules posited as primitive rules in the former are derived as *theorems* (in the technical sense of the term) in the latter.

Before moving on, I should hasten to note that the TLCG literature is more varied than the CCG literature, consisting of several related but distinct lines of research. I choose to present one particular variant called Hybrid Type-Logical Categorial Grammar (Kubota & Levine 2020) in what follows, in line with the present chapter's linguistic emphasis (for a more in-depth discussion on the linguistic application of TLCG, see Carpenter 1998 and Kubota & Levine 2020). A brief comparison with major alternatives can be found in Chapter 12 of Kubota & Levine (2020). Other variants of TLCG, most notably, the *Categorial Type Logics* (Moortgat 2011) and *Displacement Calculus* (Morrill 2010) emphasize logical and computational aspects. Moot & Retoré (2012) is a good introduction to TLCG with emphasis on these latter aspects.

2.5.1 The Lambek calculus

In addition to the *Slash Elimination* rules (reproduced here as (29)), which are identical to the two rules in the AB grammar from Section 2.2, the Lambek calculus posits the *Slash Introduction* rules, which can be written in the current labelled deduction format as in (30) (the vertical dots around the hypothesis abbreviate an arbitrarily complex proof structure).[12]

(29) a. *Forward Slash Elimination*

$$\frac{\mathsf{a}; \mathcal{F}; A/B \qquad \mathsf{b}; \mathcal{G}; B}{\mathsf{a} \circ \mathsf{b}; \mathcal{F}(\mathcal{G}); A}\ /\mathrm{E}$$

b. *Backward Slash Elimination*

$$\frac{\mathsf{b}; \mathcal{G}; B \qquad \mathsf{a}; \mathcal{F}; B\backslash A}{\mathsf{b} \circ \mathsf{a}; \mathcal{F}(\mathcal{G}); A}\ \backslash\mathrm{E}$$

(30) a. *Forward Slash Introduction*

$$\dfrac{\dfrac{\begin{array}{ccc} \vdots & \underline{[\varphi; x; A]^n} & \vdots \\ \vdots & \vdots & \vdots \end{array}}{\mathsf{b} \circ \varphi; \mathcal{F}; B}}{\mathsf{b}; \lambda x.\mathcal{F}; B/A}\ /\mathrm{I}^n$$

b. *Backward Slash Introduction*

$$\dfrac{\dfrac{\begin{array}{ccc} \vdots & \underline{[\varphi; x; A]^n} & \vdots \\ \vdots & \vdots & \vdots \end{array}}{\varphi \circ \mathsf{b}; \mathcal{F}; B}}{\mathsf{b}; \lambda x.\mathcal{F}; A\backslash B}\ \backslash\,\mathrm{I}^n$$

The key idea behind the Slash Introduction rules in (30) is that they allow one to derive linguistic expressions by *hypothetically* assuming the existence of words and phrases that are not (necessarily) overtly present. For example, (30a) can be understood as consisting of two steps of inference: one first draws a (tentative) conclusion that the string of words $\mathsf{b} \circ \varphi$ is of type B, by hypothetically assuming the existence of an expression φ of type A (where a hypothesis is enclosed in square brackets to indicate its status as such). At that point, one can draw the (real) conclusion that b alone is of type B/A since it was just shown to be an expression that yields B *if* there is an A (namely, φ) to its right. Note that the final conclusion no longer depends on the hypothesis that there is an expression φ of type A. More technically, the hypothesis is *withdrawn* at the final step.

One consequence that immediately follows in this system is that Type Raising and Function Composition (as well as other theorems; see, for example, Jäger 2005: Section 2.2.5, pp. 46–49) are now derivable as theorems. As an illustration, the proofs for (14a) and (13a) are shown in (31) and (32), respectively.

(31)
$$\dfrac{\dfrac{[\varphi; v; A\backslash B]^1 \qquad \mathsf{a}; \mathcal{F}; A}{\mathsf{a} \circ \varphi; v(\mathcal{F}); B}\ \backslash\mathrm{E}}{\mathsf{a}; \lambda v.v(\mathcal{F}); B/(A\backslash B)}\ /\mathrm{I}^1$$

[12] Morrill (1994) was the first to recast the Lambek calculus in this labelled deduction format.

(32)
$$\dfrac{\dfrac{a;\mathcal{F};A/B \qquad \dfrac{[\varphi;x;C]^1 \qquad b;\mathcal{G};B/C}{b\circ\varphi;\mathcal{G}(x);B}\,/\mathrm{E}}{a\circ b\circ\varphi;\mathcal{F}(\mathcal{G}(x));A}\,/\mathrm{E}}{a\circ b;\lambda x.\mathcal{F}(\mathcal{G}(x));A/C}\,/\mathrm{I}^1$$

These are formal theorems, but they intuitively make sense. For example, what's going on in (32) is simple. Some expression of type C is hypothetically assumed first, which is then combined with B/C. This produces a larger expression of type B, which can then be fed as an argument to A/B. At that point, the initial hypothesis is withdrawn and it is concluded that what one really had was just something that would become an A *if* there is a C to its right, namely, an expression of type A/C. Thus, a sequence of expression of types A/B and B/C is proven to be of type A/C. This type of proof is known as *hypothetical reasoning*, since it involves a step of positing a hypothesis initially and withdrawing that hypothesis at a later point.

Getting back to some notational issues, there are two crucial things to keep in mind about the notational convention adopted here (which I implicitly assumed above). First, the connective $\circ$ in the prosodic component designates string concatenation and is associative in both directions (i.e. $(\varphi_1 \circ \varphi_2) \circ \varphi_3 \equiv \varphi_1 \circ (\varphi_2 \circ \varphi_3)$). In other words, hierarchical structure is irrelevant for the prosodic representation. Thus, the applicability condition on the Forward Slash Introduction rule (30a) is simply that the prosodic variable φ of the hypothesis appears as the rightmost element of the string prosody of the input expression (i.e. $b \circ \varphi$). Since the penultimate step in (32) satisfies this condition, the rule is applicable here. Second, note in this connection that the application of the Introduction rules is conditioned on the position of the prosodic variable, and *not* on the position of the hypothesis itself in the proof tree (this latter convention is more standardly adopted when the Lambek calculus is presented in Prawitz-style natural deduction, though the two presentations are equivalent—see, for example, Carpenter 1998 and Jäger 2005).

Hypothetical reasoning with Slash Introduction makes it possible to recast the CCG analysis of nonconstituent coordination from Section 2.4.1 within the logic of / and \. This reformulation fully retains the essential analytic ideas of the original CCG analysis but makes the underlying logic of syntactic composition more transparent.

The following derivation illustrates how the "reanalysis" of the string *Bill the book* as a derived constituent of type (VP/NP/NP)\VP (the same type as in (19)) can be obtained in the Lambek calculus:

(33) $$\dfrac{\dfrac{\dfrac{[\varphi; f; \mathrm{VP/NP/NP}]^1 \qquad \mathsf{bill}; \mathbf{b}; \mathrm{NP}}{\varphi \circ \mathsf{bill}; f(\mathbf{b}); \mathrm{VP/NP}}/\mathrm{E} \qquad \mathsf{the} \circ \mathsf{book}; \iota(\mathbf{bk}); \mathrm{NP}}{\varphi \circ \mathsf{bill} \circ \mathsf{the} \circ \mathsf{book}; f(\mathbf{b})(\iota(\mathbf{bk})); \mathrm{VP}}/\mathrm{E}}{\mathsf{bill} \circ \mathsf{the} \circ \mathsf{book}; \lambda f.f(\mathbf{b})(\iota(\mathbf{bk})); (\mathrm{VP/NP/NP})\backslash\mathrm{VP}}\backslash\mathrm{I}^1$$

At this point, one may wonder what the relationship is between the analysis of nonconstituent coordination via Type Raising and Function Composition in the ABC grammar in Section 2.4.1 and the hypothetical reasoning-based analysis in the Lambek calculus just presented. Intuitively, they seem to achieve the same effect in slightly different ways. The logic-based perspective of TLCG allows us to obtain a deeper understanding of the relationship between them. To facilitate comparison, I first recast the Type Raising + Function Composition analysis from Section 2.4.1 in the Lambek calculus. The relevant part is the part that derives the "noncanonical constituent" *Bill the book*:

(34) $$\dfrac{\dfrac{[\varphi_3; R; \mathrm{DTV}]^3 \qquad \dfrac{\dfrac{[\varphi_2; P; \mathrm{DTV}]^2 \qquad \mathsf{bill}; \mathbf{b}; \mathrm{NP}}{\varphi_2 \circ \mathsf{bill}; P(\mathbf{b}); \mathrm{TV}}/\mathrm{E}}{\mathsf{bill}; \lambda P.P(\mathbf{b}); \mathrm{DTV}\backslash\mathrm{TV}}\backslash\mathrm{I}^2}{\varphi_3 \circ \mathsf{bill}; R(\mathbf{b}); \mathrm{TV}}\backslash\mathrm{E} \qquad \dfrac{\dfrac{[\varphi_1; Q; \mathrm{TV}]^1 \qquad \mathsf{the} \circ \mathsf{book}; \iota(\mathbf{bk}); \mathrm{NP}}{\varphi_1 \circ \mathsf{the} \circ \mathsf{book}; Q(\iota(\mathbf{bk})); \mathrm{VP}}/\mathrm{E}}{\mathsf{the} \circ \mathsf{book}; \lambda Q.Q(\iota(\mathbf{bk})); \mathrm{TV}\backslash\mathrm{VP}}\backslash\mathrm{I}^1}{\dfrac{\varphi_3 \circ \mathsf{bill} \circ \mathsf{the} \circ \mathsf{book}; R(\mathbf{b})(\iota(\mathbf{bk})); \mathrm{VP}}{\mathsf{bill} \circ \mathsf{the} \circ \mathsf{book}; \lambda R.R(\mathbf{b})(\iota(\mathbf{bk})); \mathrm{DTV}\backslash\mathrm{VP}}\backslash\mathrm{I}^3}\backslash\mathrm{E}$$

By comparing (34) and (33), one can see that (34) contains some redundant steps. First, hypothesis 2 (φ_2) is introduced only to be replaced by hypothesis 3 (φ_3). This is completely redundant, since one could have obtained exactly the same result by directly combining hypothesis 3 with the NP *Bill*. Similarly, hypothesis 1 can be eliminated by replacing it with the TV $\varphi_3 \circ$ book on the left-hand side of the third line from the bottom. By making these two simplifications, the derivation in (33) is obtained.

The relationship between the more complex proof in (34) and the simpler one in (33) is parallel to the relationship between an unreduced lambda term (such as $\lambda R[\lambda Q[Q(\iota(\mathbf{bk}))](\lambda P[P(\mathbf{b})](R))]$) and its β-normal form (i.e. $\lambda R.R(\mathbf{b})(\iota(\mathbf{bk}))$). In fact, there is a formally precise one-to-one relationship between linear logic (of which the Lambek calculus is known to be a straightforward extension) and the typed lambda calculus known as the *Curry-Howard Isomorphism* (Howard 1969), according to which the lambda term that represents the proof (34) β-reduces to the term that represents the proof (33).[13] Technically, this is known as *proof normalization* (Jäger 2005 contains a particularly useful discussion on this notion).

[13] There is a close relationship between these lambda terms representing proofs (i.e. syntactic

Thus, the logic-based architecture of the Lambek calculus (and various versions of TLCG, which are all extensions of the Lambek calculus) enables us to say, in a technically precise way, how (34) and (33) are the "same" (or, more precisely, equivalent), by building on independently established results in mathematical logic and computer science. This is one big advantage of taking seriously the view, advocated by the TLCG research, that "language *is* logic".

2.5.2 Extending the Lambek calculus

Hypothetical reasoning is a very powerful (yet systematic) tool, but with forward and backward slashes, it is only good for analyzing expressions missing some material at the (right or left) periphery. This is problematic in the analyses of many linguistic phenomena, such as *wh*-extraction (where the "gap" can be in a sentence-medial position—recall the discussion about crossed composition rules in CCG in Section 2.4.2) and quantifier scope (where the quantifier needs to covertly move from a sentence-medial position), as well as various kinds of discontinuous constituency phenomena (see, for example, Morrill et al. 2011, which contains analyses of various types of discontinuous constituency phenomena in a recent version of TLCG known as "Displacement Calculus"). In what follows, I sketch one particular, relatively recent approach to this problem, known as *Hybrid Type-Logical Categorial Grammar* (Hybrid TLCG; Kubota 2010; 2015; Kubota & Levine 2015; 2020). This approach combines the Lambek calculus with Oehrle's (1994) term-labelled calculus, which deals with discontinuity by employing λ-binding in the prosodic component.

Hybrid TLCG extends the Lambek calculus with the Elimination and Introduction rules for the *vertical slash*:

(35) a. *Vertical Slash Introduction*

$$\dfrac{\dfrac{\begin{array}{ccc} \vdots & \dfrac{}{[\varphi; x; A]^n} & \vdots \\ \vdots & \vdots & \vdots \end{array}}{\mathsf{b}; \mathcal{F}; B}}{\lambda\varphi.\mathsf{b}; \lambda x.\mathcal{F}; B\upharpoonright A}\ \upharpoonright\mathrm{I}^n$$

b. *Vertical Slash Elimination*

$$\dfrac{\mathsf{a}; \mathcal{F}; A\upharpoonright B \qquad \mathsf{b}; \mathcal{G}; B}{\mathsf{a}(\mathsf{b}); \mathcal{F}(\mathcal{G}); A}\ \upharpoonright\mathrm{E}$$

These rules make it possible to model what (roughly) corresponds to syntactic

derivations) and the lambda terms that one writes to notate semantic translations, especially if the latter is written at each step of derivation *without* performing β-reduction. But it is important to keep in mind that lambda terms representing syntactic proofs and lambda terms notating semantic translations are distinct things.

movement operations in mainstream generative grammar. This is illustrated in (36) for the $\forall > \exists$ reading for the sentence *Someone talked to everyone today.*

(36)

$$\dfrac{[\varphi_2; x_2; \mathrm{NP}]^2 \qquad \dfrac{\mathsf{talked} \circ \mathsf{to}; \mathbf{talked\text{-}to}; (\mathrm{NP}\backslash \mathrm{S})/\mathrm{NP} \qquad [\varphi_1; x_1; \mathrm{NP}]^1}{\mathsf{talked} \circ \mathsf{to} \circ \varphi_1; \mathbf{talked\text{-}to}(x_1); \mathrm{NP}\backslash \mathrm{S}}\,/\mathrm{E}}{\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1; \mathbf{talked\text{-}to}(x_1)(x_2); \mathrm{S}}\,\backslash \mathrm{E}$$

$$\dfrac{\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1; \mathbf{talked\text{-}to}(x_1)(x_2); \mathrm{S} \qquad \mathsf{today}; \mathbf{tdy}; \mathrm{S}\backslash \mathrm{S}}{\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2)); \mathrm{S}}\,\backslash \mathrm{E}$$

$$① \dfrac{\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2)); \mathrm{S}}{\lambda\varphi_2.\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2)); \mathrm{S}{\upharpoonright}\mathrm{NP}}\,{\upharpoonright}\mathrm{I}^2$$

$$② \dfrac{\lambda\sigma.\sigma(\mathsf{everyone}); \mathbf{V}_{\mathbf{person}}; \mathrm{S}{\upharpoonright}(\mathrm{S}{\upharpoonright}\mathrm{NP}) \qquad \lambda\varphi_2.\varphi_2 \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2)); \mathrm{S}{\upharpoonright}\mathrm{NP}}{\mathsf{someone} \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \mathbf{\exists}_{\mathbf{person}}(\lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2))); \mathrm{S}}\,{\upharpoonright}\mathrm{E}$$

$$\dfrac{\mathsf{someone} \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \mathbf{\exists}_{\mathbf{person}}(\lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2))); \mathrm{S}}{\lambda\varphi_1.\mathsf{Someone} \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \lambda x_1.\mathbf{\exists}_{\mathbf{person}}(\lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2))); \mathrm{S}{\upharpoonright}\mathrm{NP}}\,{\upharpoonright}\mathrm{I}^1$$

$$\dfrac{\lambda\sigma.\sigma(\mathsf{someone}); \mathbf{\exists}_{\mathbf{person}}; \mathrm{S}{\upharpoonright}(\mathrm{S}{\upharpoonright}\mathrm{NP}) \qquad \lambda\varphi_1.\mathsf{Someone} \circ \mathsf{talked} \circ \mathsf{to} \circ \varphi_1 \circ \mathsf{today}; \lambda x_1.\mathbf{\exists}_{\mathbf{person}}(\lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2))); \mathrm{S}{\upharpoonright}\mathrm{NP}}{\mathsf{someone} \circ \mathsf{talked} \circ \mathsf{to} \circ \mathsf{everyone} \circ \mathsf{today}; \mathbf{V}_{\mathbf{person}}(\lambda x_1.\mathbf{\exists}_{\mathbf{person}}(\lambda x_2.\mathbf{tdy}(\mathbf{talked\text{-}to}(x_1)(x_2)))); \mathrm{S}}\,{\upharpoonright}\mathrm{E}$$

A quantifier has the ordinary GQ meaning ($\mathbf{\exists}_{\mathbf{person}}$ and $\mathbf{V}_{\mathbf{person}}$ abbreviate the terms $\lambda P.\exists x[\mathbf{person}(x) \land P(x)]$ and $\lambda P.\forall x[\mathbf{person}(x) \rightarrow P(x)]$, respectively), but its phonology is a function of type **(st→st)→st** (where **st** is the type of string). By abstracting over the position in which the quantifier "lowers into" in an S via the Vertical Slash Introduction rule (35a), an expression of type $\mathrm{S}{\upharpoonright}\mathrm{NP}$ (phonologically **st→st**) is obtained (①), which is then given as an argument to the quantifier. Then, by function application via ${\upharpoonright}\mathrm{E}$ (②), the subject quantifier *someone* semantically scopes over the sentence and lowers its phonology to the "gap" position kept track of by λ-binding in phonology (note that this result obtains by function application and beta-reduction of the prosodic term). The same process takes place for the object quantifier *everyone* to complete the derivation. The scopal relation between multiple quantifiers depends on the order of application of this hypothetical reasoning. The surface scope reading is obtained by switching the order of the hypothetical reasoning for the two quantifiers (which results in the same string of words, but with the opposite scope relation).

This formalization of quantifying-in by Oehrle (1994) has later been extended by Barker (2007) for more complex types of scope-taking phenomena known as *parasitic scope* in the analysis of symmetrical predicates (such as *same* and *different*).[14] Empirical application of parasitic scope include "respective" readings

[14] "Parasitic scope" is a notion coined by Barker (2007) where, in transformational terms, some

(Kubota & Levine 2016b), "split scope" of negative quantifiers (Kubota & Levine 2016a) and modified numerals such as *exactly N* (Pollard 2014).

Hypothetical reasoning with prosodic λ-binding enables a simple analysis of *wh*-extraction too, as originally noted by Muskens (2003). The key idea is that sentences with medial gaps can be analyzed as expressions of type $S{\upharpoonright}NP$, as in the derivation for (37) in (38).

(37) Bagels$_i$, Kim gave $\_\,_i$ to Chris.

(38)

$$
\dfrac{\begin{matrix}\textsf{bagels};\\ \mathbf{b}; NP\end{matrix}\ ② \quad \dfrac{\begin{matrix}\lambda\sigma\lambda\varphi.\varphi \circ \sigma(\epsilon);\\ \lambda\mathcal{F}.\mathcal{F};\\ (S{\upharpoonright}X){\upharpoonright}(S{\upharpoonright}X)\end{matrix}\ ① \quad \dfrac{\dfrac{\begin{matrix}\textsf{kim};\\ \mathbf{k}; NP\end{matrix} \quad \dfrac{\dfrac{\begin{matrix}\textsf{gave};\\ \mathbf{gave};\\ VP/PP/NP\end{matrix} \quad \begin{bmatrix}\varphi;\\ x;\\ NP\end{bmatrix}^1}{\textsf{gave} \circ \varphi; \mathbf{gave}(x); VP/PP}/E \quad \begin{matrix}\textsf{to} \circ \textsf{chris};\\ \mathbf{c}; PP\end{matrix}}{\textsf{gave} \circ \varphi \circ \textsf{to} \circ \textsf{chris}; \mathbf{gave}(x)(\mathbf{c}); VP}/E}{\textsf{kim} \circ \textsf{gave} \circ \varphi \circ \textsf{to} \circ \textsf{chris}; \mathbf{gave}(x)(\mathbf{c})(\mathbf{k}); S}\backslash E}{\begin{matrix}\lambda\varphi.\textsf{kim} \circ \textsf{gave} \circ \varphi \circ \textsf{to} \circ \textsf{chris};\\ \lambda x.\mathbf{gave}(x)(\mathbf{c})(\mathbf{k}); S{\upharpoonright}NP\end{matrix}}{\upharpoonright}I^1}{\lambda\varphi.\varphi \circ \textsf{kim} \circ \textsf{gave} \circ \textsf{to} \circ \textsf{chris}; \lambda x.\mathbf{gave}(x)(\mathbf{c})(\mathbf{k}); S{\upharpoonright}NP}{\upharpoonright}E}{\textsf{bagels} \circ \textsf{kim} \circ \textsf{gave} \circ \textsf{to} \circ \textsf{chris}; \mathbf{gave}(\mathbf{b})(\mathbf{c})(\mathbf{k}); S}{\upharpoonright}E
$$

Here, after deriving an $S{\upharpoonright}NP$, which keeps track of the gap position via the λ-bound variable φ, the topicalization operator fills in the gap with an empty string and concatenates the topicalized NP to the left of the string thus obtained. This way, the difference between "overt" and "covert" movement reduces to a lexical difference in the prosodic specifications of the operators that induce them. A covert movement operator throws in some material in the gap position, whereas an overt movement operator "closes off" the gap with an empty string.

As illustrated above, hypothetical reasoning for the Lambek slashes / and \ and for the vertical slash $\upharpoonright$ have important empirical motivations, but the real strength of a "hybrid" system like Hybrid TLCG which recognizes both types of slashes is that it extends automatically to cases in which "directional" and "non-directional" phenomena interact. A case in point comes from the interaction of nonconstituent coordination and quantifier scope. Examples such as those in (39) allow for at least a reading in which the shared quantifier outscopes conjunction.[15]

expression takes scope at LF by parasitizing on the scope created by a different scopal operator's LF movement. In versions of (TL)CG of the sort discussed here, this corresponds to double lambda-abstraction via the vertical slash.

[15] Whether the other scopal relation (one in which the quantifier meaning is "distributed" to each conjunct, as in the paraphrase "I gave a couple of books to Pat on Monday and I gave a couple

(39) a. I gave a couple of books to Pat on Monday and to Sandy on Tuesday.
b. Terry said nothing to Robin on Thursday or to Leslie on Friday.

I now illustrate how this wide scope reading for the quantifier in NCC sentences like (39) is immediately predicted to be available in the fragment developed so far (Hybrid TLCG actually predicts both scopal relations for all NCC sentences; see Kubota & Levine 2015 for how the distributive scope is licensed). The derivation for (39b) is given in (40).

of books to Sandy on Tuesday" for (39)) is possible seems to depend on various factors. With downward-entailing quantifiers such as (39b), this reading seems difficult to obtain without heavy contextualization and appropriate intonational cues. See Kubota & Levine (2015) for some discussion.

(40)

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{[\varphi_1; P; \mathrm{VP/PP/NP}]^1 \quad [\varphi_2; x; \mathrm{NP}]^2}{\varphi_1 \circ \varphi_2; P(x); \mathrm{VP/PP}}\,/\mathrm{E} \quad \text{to} \circ \text{robin}; \mathbf{r}; \mathrm{PP}}{\varphi_1 \circ \varphi_2 \circ \text{to} \circ \text{robin}; P(x)(\mathbf{r}); \mathrm{VP}}\,/\mathrm{E} \quad \text{on} \circ \text{thursday}; \mathbf{onTh}; \mathrm{VP\backslash VP}}{\varphi_1 \circ \varphi_2 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday}; \mathbf{onTh}(P(x)(\mathbf{r})); \mathrm{VP}}\,\backslash\mathrm{E}}{\varphi_2 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday}; \lambda P.\mathbf{onTh}(P(x)(\mathbf{r})); \mathrm{(VP/PP/NP)\backslash VP}}\,\backslash\,\mathrm{I}^1}{\text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday}; \lambda x \lambda P.\mathbf{onTh}(P(x)(\mathbf{r})); \mathrm{NP\backslash(VP/PP/NP)\backslash VP}}\,\backslash\,\mathrm{I}^2$$

$$\vdots$$

$$\cfrac{\begin{matrix}\vdots \\ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday}; \\ \lambda x \lambda P.\mathbf{onTh}(P(x)(\mathbf{r})); \\ \mathrm{NP\backslash(VP/PP/NP)\backslash VP}\end{matrix} \quad \cfrac{\begin{matrix}\text{or}; \\ \lambda \mathcal{V} \lambda \mathcal{W}.\mathcal{W} \sqcup \mathcal{V}; \\ (X\backslash X)/X\end{matrix} \quad \begin{matrix}\text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda x \lambda P.\mathbf{onFr}(P(x)(\mathbf{l})); \\ \mathrm{NP\backslash(VP/PP/NP)\backslash VP}\end{matrix}}{\begin{matrix}\text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda \mathcal{W}.\mathcal{W} \sqcup [\lambda x \lambda P.\mathbf{onFr}(P(x)(\mathbf{l}))]; \\ \mathrm{(NP\backslash(VP/PP/NP)\backslash VP)\backslash(NP\backslash(VP/PP/NP)\backslash VP)}\end{matrix}}\,/\mathrm{E}}{\begin{matrix}\text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda x \lambda P \lambda z.\mathbf{onTh}(P(x)(\mathbf{r}))(z) \vee \mathbf{onFr}(P(x)(\mathbf{l}))(z); \mathrm{NP\backslash(VP/PP/NP)\backslash VP}\end{matrix}}\,\backslash\mathrm{E}$$

$$\vdots$$

$$\cfrac{\begin{matrix}\lambda\sigma.\sigma(\text{nothing}); \\ \neg\exists_{\mathbf{thing}}; \\ \mathrm{S{\upharpoonright}(S{\upharpoonright}NP)}\end{matrix} \quad \cfrac{\cfrac{\begin{matrix}\text{terry}; \\ \mathbf{t}; \mathrm{NP}\end{matrix} \quad \cfrac{\begin{matrix}\text{said}; \\ \mathbf{said}; \\ \mathrm{VP/NP/PP}\end{matrix} \quad \cfrac{\begin{bmatrix}\varphi_3; \\ x; \mathrm{NP}\end{bmatrix}^3 \quad \begin{matrix}\text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \\ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda x \lambda P \lambda z.\mathbf{onTh}(P(x)(\mathbf{r}))(z) \vee \\ \mathbf{onFr}(P(x)(\mathbf{l}))(z); \\ \mathrm{NP\backslash(VP/PP/NP)\backslash VP}\end{matrix}}{\begin{matrix}\varphi_3 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \\ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda P \lambda z.\mathbf{onTh}(P(x)(\mathbf{r}))(z) \vee \\ \mathbf{onFr}(P(x)(\mathbf{l}))(z); \\ \mathrm{(VP/PP/NP)\backslash VP}\end{matrix}}\,\backslash\mathrm{E}}{\begin{matrix}\text{said} \circ \varphi_3 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \\ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda z.\mathbf{onTh}(\mathbf{said}(x)(\mathbf{r}))(z) \vee \mathbf{onFr}(\mathbf{said}(x)(\mathbf{l}))(z); \mathrm{VP}\end{matrix}}\,\backslash\mathrm{E}}{\begin{matrix}\text{terry} \circ \text{said} \circ \varphi_3 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \\ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \mathbf{onTh}(\mathbf{said}(x)(\mathbf{r}))(\mathbf{t}) \vee \mathbf{onFr}(\mathbf{said}(x)(\mathbf{l}))(\mathbf{t}); \mathrm{S}\end{matrix}}\,\backslash\mathrm{E}}{\begin{matrix}\lambda\varphi_3.\text{terry} \circ \text{said} \circ \varphi_3 \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \\ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \lambda x.\mathbf{onTh}(\mathbf{said}(x)(\mathbf{r}))(\mathbf{t}) \vee \mathbf{onFr}(\mathbf{said}(x)(\mathbf{l}))(\mathbf{t}); \mathrm{S{\upharpoonright}NP}\end{matrix}}\,{\upharpoonright}\mathrm{I}^3}{\begin{matrix}\text{terry} \circ \text{said} \circ \text{nothing} \circ \text{to} \circ \text{robin} \circ \text{on} \circ \text{thursday} \circ \text{or} \circ \text{to} \circ \text{leslie} \circ \text{on} \circ \text{friday}; \\ \neg\exists_{\mathbf{thing}}(\lambda x.\mathbf{onTh}(\mathbf{said}(x)(\mathbf{r}))(\mathbf{t}) \vee \mathbf{onFr}(\mathbf{said}(x)(\mathbf{l}))(\mathbf{t})); \mathrm{S}\end{matrix}}\,{\upharpoonright}\mathrm{E}$$

The key point in this derivation is that, via hypothetical reasoning, the string *to Robin on Thursday or to Leslie on Friday* forms a syntactic constituent with a full-fledged meaning assigned to it in the usual way. Then the quantifier takes scope above this whole coordinate structure, yielding the non-distributive, quantifier wide-scope reading.

Licensing the correct scopal relation between the quantifier and conjunction in the analysis of NCC remains a challenging problem in the HPSG literature. See Section 4.2.1 for some discussion.

3 Architectural similarities and differences

3.1 Broad architecture

One important property common to HPSG and CG is that they are both "lexicalist" theories of syntax in the broader sense.[16] This is partly due to an explicit choice made at an early stage of the development of HPSG to encode valence information in the syntactic categories of linguistic expressions, following CG (see Flickinger, Pollard & Wasow 2020, Chapter 2 of this volume and Davis & Koenig 2020, Chapter 4 of this volume).[17] The two theories share many similarities in the analyses of specific linguistic phenomena due to this basic architectural similarity. For example, many phenomena that are treated by means of local movement operations (or via empty categories) in mainstream generative syntax, such as passivization, raising/control in English and "complex predicate" phenomena in a typologically broad range of languages are generally treated by the sharing of valence information in the lexicon in these theories. For HPSG analyses of these phenomena, see Wechsler, Koenig & Davis (2020), Chapter 9 of this volume, Godard & Samvelian (2020), Chapter 12 of this volume and Abeillé (2020), Chapter 13 of this volume. Steedman & Baldridge (2011) contains a good summary of CG analyses of local dependencies (passivization, raising/control). Kubota (2014) contains a comparison of HPSG and CG analyses of complex predicates. The heavy reliance on "lexicalist" analyses of local dependencies is perhaps the most

[16] I say "broader sense" here since not all variants of either HPSG or CG subscribe to the so-called "lexical integrity hypothesis" (see Davis & Koenig 2020, Chapter 4 of this volume), which says that syntax and morphology are distinct components of grammar. For example, in the CG literature, the treatments of verb clustering in Dutch by Moortgat & Oehrle (1994) and in Japanese by Kubota (2014) seem to go against the tenet of the lexical integrity hypothesis. In HPSG, Gunji (1999) formulates an analysis of Japanese causatives that does not adhere to the lexical integrity hypothesis and which contrasts sharply with the strictly lexicalist analysis by Manning et al. (1999). See also Davis & Koenig (2020), Chapter 4 of this volume, Bruening (2018b,a), Müller (2018) and Müller & Wechsler (2014) for some discussion on lexicalism.

[17] This point is explicitly noted by the founders of HPSG in the following passage in Pollard & Sag (1987):

> A third principle of universal grammar posited by HPSG, the *Subcategorization Principle*, is essentially a generalization of the "argument cancellation" employed in categorial grammar. (Pollard & Sag 1987: 11)

important property that is shared in common in HPSG and various versions of CG.

But emphasizing this commonality too much may be a bit misleading, since the valence features of HPSG and the slash connectives in CG have very different ontological statuses in the respective theories. The valence features in HPSG are primarily specifications, closely tied to the specific phrase structure rules, that dictate the ways in which hierarchical representations are built. To be sure, the lexical specifications of the valence information play a key role in the movement-free analyses of local dependencies along the lines noted above, but still, there is a rather tight connection between these valence specifications originating in the lexicon and the ways in which they are "cancelled" in specific phrase structure rules.

Things are quite different in CG, especially in TLCG. As discussed in Section 2, TLCG views the grammar of natural language *not* as a structure-building system, but as a logical deductive system. The two slashes / and \ are thus not "features" that encode the subcategorization properties of words in the lexicon, but have a much more general and fundamental role within the basic architecture of grammar in TLCG. These connectives are literally implicational connectives within a logical calculus. Thus, in TLCG, "derived" rules such as Type Raising and Function Composition are *theorems*, in just the same way that the transitivity inference is a theorem in classical propositional logic. Note that this is not just a matter of high-level conceptual organization of the theory, since, as discussed in Section 2, the ability to assign "constituent" statuses to non-canonical constituents in the CG analyses of NCC directly exploits this property of the underlying calculus. The straightforward mapping from syntax to semantics discussed in Section 2.3 is also a direct consequence of adopting this "derivation as proof" perspective on syntax, building on the results of the Curry-Howard correspondence (Howard 1969) in setting up the syntax-semantics interface.[18]

Another notable difference between (especially a recent variant of) HPSG and CG is that CG currently lacks a detailed theory of "constructions", that is, patterns and (sub)regularities that are exhibited by linguistic expressions that cannot (at least according to the proponents of "constructionist" approaches) be lexicalized easily. As discussed in Müller (2020b), Chapter 34 of this volume (see also Sag 1997, Fillmore 1999 and Ginzburg & Sag 2000), recent Sign-Based Con-

[18] Although CCG does not embody the idea of "derivation as proof" as explicitly as TLCG does, it remains true to a large extent that the role of the slash connective within the overall theory is different from that of the valence features in HPSG, given that CCG and TLCG share many key ideas in the analyses of actual empirical phenomena.

struction Grammar (SBCG) variants of HPSG (Sag et al. 2012) incorporate ideas from Construction Grammar (Fillmore et al. 1988) and capture such generalizations via a set of constructional templates (or schemata), which are essentially a family of related phrase structure rules that are organized in a type inheritance hierarchy.

Such an architecture seems nearly impossible to implement literally in CG, except via empty operators or lexical operations corresponding to each such constructional schema. In particular, in TLCG, syntactic rules are logical inference rules, so, if one strictly adheres to its slogan "language is logic", there is no option to freely add syntactic rules in the deductive system. The general consensus in the literature seems to be that while many of the phenomena initially adduced as evidence for a constructional approach can be lexicalized (see, for example, Müller & Wechsler 2014 and Müller 2020b, Chapter 34 of this volume; see also Steedman & Baldridge 2011, which discusses ways in which some of the empirical generalizations that Goldberg 1995 adduces to the notion of constructions can be lexicalized within CCG), there remain some real challenges for a strictly lexicalist approach (Müller 2020b, Chapter 34 of this volume identifies the *N after N* construction as an instance of this latter type of phenomenon). It then seems undeniable that the grammar of natural language is equipped with mechanisms for dealing with "peripheral" patterns, but whether such mechanisms should be given a central role in the architecture of grammar is still a highly controversial issue. Whatever position one takes, it is important to keep in mind that this is ultimately an empirical question (a very complex and tough one indeed) that should be settled on the basis of (various types of) evidence.

3.2 Syntax-semantics interface

As should be clear from the exposition in Section 2, both CCG and TLCG (at least in the simplest form) adopt a very rigid, one-to-one correspondence between syntax and semantics. Steedman's work on CCG has demonstrated that this simple and systematic mapping between syntax and semantics enables attractive analyses of a number of empirical phenomena at the syntax-semantics interface, including some notorious problems such as the scope parallelism issue in right-node raising known as the Geach paradigm (*Every boy loves, and every girl detests, some saxophonist*; Geach 1970). Other important work on issues at the syntax-semantics interface includes Jacobson's (1999; 2000) work on pronominal anaphora in Variable-Free Semantics (covering a wide range of phenomena including the paycheck/Bach-Peters paradigms and binding parallelism in right-node raising), Barker & Shan's (2015) work on "continuation-based" semantics (weak crossover, superiority effects and "parasitic scope" treatments of symmetrical predicates and sluicing) and Kubota and Levine's (2015; 2017; 2020) Hybrid TLCG, dealing with interactions between coordination, ellipsis and scopal phenomena.

As discussed in Koenig & Richter (2020), Chapter 23 of this volume, recent HPSG work on complex empirical phenomena at the syntax-semantics interface makes heavy use of underspecification. For example, major analyses of nonconstituent coordination in recent HPSG use some version of an underspecification framework to deal with complex interactions between coordination and scopal operators. (Yatabe 2001; Beavers & Sag 2004; Park et al. 2019; Park 2019; Yatabe & Tam 2019). In a sense, HPSG retains a rigid phrase structure-based syntax (modulo the flexibility entertained with the use of the linearization-based architecture) and deals with the complex mapping to semantics via the use of underspecification languages in the semantic component (such as Minimal Recursion Semantics by Copestake et al. 2005 and Lexical Resources Semantics by Richter & Sailer 2004). CG, on the other hand, tends to adhere more closely to a tight mapping from syntax to semantics, but makes the syntactic component itself flexible. But it is important to keep in mind that, even within the CG research community, there is no clear consensus about how strictly one should adhere to the Montagovian notion of compositionality—a glimpse of the recent literature reveals that the issue is very much an open-ended one: many contemporary variants of CG make use of underspecification for certain purposes (see, for example, Steedman 2012, Bekki 2014, Bekki & Mineshima 2017 and Kubota et al. 2019), while at the same time Jacobson's (1999; 2000) program of Variable-Free Semantics is distinct in explicitly taking the classical notion of compositionality as a driving principle.

3.3 Morpho-syntax and word order

While there is relatively less detailed work on morphology and the morphosyntax interface in CG as compared to HPSG, there are several ideas originating in the CG literature that have either influenced some HPSG work or which are closely related to a certain line of work in HPSG. I review some of these in this section.[19]

[19] An important omission in the ensuing discussion is a comparison of recent work in HPSG on morphology by Olivier Bonami and Berthold Crysmann (see Crysmann 2020, Chapter 22 of this volume), which builds on and extends Greg Stump's *Paradigm Function Morphology* (PFM; Stump 2001), and early CG work on morphology (Hoeksema 1984; Moortgat 1984; Hoeksema & Janda 1988; Raffelsiefen 1992) which could be viewed as precursors of PFM.

3.3.1 Linearization-based HPSG and the phenogrammar/tectogrammar distinction in CG

The idea of separating surface word order and the underlying combinatorics, embodied in the so-called *linearization-based* version of HPSG (Reape 1994; Kathol 2000; cf. Müller 2020a, Chapter 10 of this volume), has its origin in the work by the logician Haskell Curry (1961), in which he proposed the distinction between *phenogrammar* (the component pertaining to surface word order) and *tectogrammar* (underlying combinatorics). This same idea has influenced a certain line of work in the CG literature too. Important early work was done by Dowty (1982a); Dowty (1996) in a variant of CG which is essentially an AB grammar with "syncategoremmatic" rules that directly manipulate string representations, of the sort utilized in Montague Grammar, for dealing with various sorts of discontinuous constituency.[20]

Dowty's early work has influenced two separate lines of work in the later development of CG. First, a more formally sophisticated implementation of an enriched theory of phenogrammatical component of the sort sketched in Dowty (1996) was developed in the literature of Multi-Modal Categorial Type Logics in the 90s, by exploiting the notion of "modal control" (as already noted, this technique was later incorporated into CCG by Baldridge 2002). Some empirical work in this line of research includes Moortgat & Oehrle (1994) (on Dutch cross-serial dependencies; see also Dowty 1997 for an accessible exposition of this analysis), Kraak (1998) (French clitic climbing), Whitman (2009) ("right-node wrapping" in English) and Kubota (2010; 2014) (complex predicates in Japanese). Second, the Curry/Dowty idea of pheno/tecto distinction has also been the core motivation for the underlying architecture of a family of approaches called *Linear Categorial Grammar* (LCG; Oehrle 1994; de Groote 2001; Muskens 2003; Mihaliček & Pollard 2012; Pollard 2013), in which, following the work of Oehrle (1994), the prosodic component is modelled as a lambda calculus (cf. Section 2.5.2) for dealing with complex operations pertaining to word order (the more standard approach in the TLCG tradition is to model the prosodic component as some sort of algebra of structured strings as in Moortgat 1997 and Morrill et al. 2011). In fact, among different variants of CG, LCG can be thought of as an extremist approach in relegating word order completely from the combinatorics, by doing away with the distinction between the Lambek forward and backward slashes.

One issue that arises for approaches that distinguish between the levels of

[20] See also Flickinger, Pollard & Wasow (2020), Chapter 2 of this volume, Section 1.2 for a discussion of the influence that early forms of CG (Bach 1979; 1980; Dowty 1982a; Dowty 1982b) had on Head Grammar (Pollard 1984), a precursor of HPSG.

phenogrammar and tectogrammar, across the HPSG/CG divide, is how closely these two components interact with one another. Kubota (2014) discusses some data in the morpho-syntax of complex predicates in Japanese which (according to him) would call for an architecture of grammar in which the pheno and tecto components interact with one another closely, and which would thus be problematic for the simpler LCG-type architecture. It would be interesting to see whether/to what extent this same criticism would carry over to linearization-based HPSG, which is similar (at least in its simplest form) to LCG in maintaining a clear separation of the pheno/tecto components.[21]

3.3.2 Syntactic features and feature neutralization

As compared to HPSG, the status of syntactic features in CG is somewhat unclear, despite the fact that such "features" are often used in linguistic analyses in the CG literature. One reason that a full-blown theory of syntactic features has not been developed in CG research to date seems to be that as compared to HPSG, syntactic features play a far less major role in linguistic analysis in CG. Another possible reason is that empirical work on complex linguistic phenomena (especially on languages other than English) are still very few in number in CG.

It is certainly conceivable to develop a theory of syntactic features and feature underspecification within CG by borrowing ideas from HPSG, for which there is already a rich tradition of foundational work on this issue. In fact, the work on Unification-based Categorial Grammar (Calder et al. 1988) explored at the end of the 80s seems to have had precisely such a goal. Unfortunately, this approach remains largely isolated from other developments in the literature (of either CG or other grammatical theories/formalisms). Another possibility would be to pursue a more logic-based approach. For some ideas, see Bayer & Johnson (1995), Bayer (1996) and Morrill (1994). Morrill (1994) in particular briefly explores the idea of implementing syntactic features via the notion of *dependent types*. There is some renewed interest in linguistic application of ideas from Dependent Type Theory (Martin-Löf 1984) in the recent literature of CG and formal semantics (see, for example, Chatzikyriakidis & Luo 2017), so pursuing this latter type of approach in connection with this new line of work may lead to some interesting developments.

[21]But note also in this connection that linearization-based HPSG is by no means monolithic; for example, Yatabe & Tam (2019) (discussed below in Section 4.2.1) propose a somewhat radical extension of the linearization-based approach in which semantic composition is done at the level of word order domains.

One issue that is worth noting in connection to syntactic features is the treatment of case syncretism and feature neutralization (cf. Przepiórkowski 2020, Chapter 7 of this volume, Section 2). The work by Morrill (1994), Bayer (1996) and Bayer & Johnson (1995) mentioned above proposed an approach to feature neutralization by positing meet and join connectives (which are like conjunction and disjunction in propositional logic) in CG. The key idea of this approach was recast in HPSG by means of inheritance hierarchies by Levy (2001), Levy & Pollard (2002) and Daniels (2002).[22] See Przepiórkowski (2020), Chapter 7 of this volume, Section 2 for an exposition of this HPSG work on feature neutralization.

4 Specific empirical phenomena

Part II of the present handbook contains an excellent introduction to recent developments of HPSG research on major linguistic phenomena. I will therefore presuppose familiarity with such recent analyses, and my discussion below aims to highlight the differences between HPSG and CG in the analyses of selected empirical phenomena. In order to make the ensuing discussion maximally informative, I focus on phenomena over which there is some ongoing major cross-theoretical debate, and those for which I believe one or the other theory would benefit from recent developments/rich research tradition in the other.

4.1 Long-distance dependencies

As noted in Section 2.4, CCG treats long-distance dependencies via a sequence of Function Composition, which is similar to the SLASH percolation analysis in HPSG. CCG offers a treatment of major aspects of long-distance dependencies, including island effects (Steedman 2000) and parasitic gaps (Steedman 1987). Earlier versions of CCG involved a somewhat ad-hoc stipulation on the use of crossed composition rules (Steedman 1997). This was overcome in the more recent, "multimodal" variant of CCG (Baldridge 2002), which controls the application of such non-order-preserving rules via a fine-grained system of "lexicalized modality". The modality specifications in this new version of CCG enable one to relocate language-specific idiosyncrasies to the lexicon, in line with the general spirit of lexicalist theories of grammar.

The situation is somewhat different in TLCG. TLCG typically makes use of a movement-like operation for the treatment of extraction phenomena (via hypo-

[22] As noted by Levy (2001), the type hierarchy-based rendering of "meet" and "join" was first introduced in HPSG by Levine et al. (2001).

thetical reasoning), but the specific implementations differ considerably in different variants of TLCG. Major alternatives include the approach in terms of "structural control" in Multi-Modal Categorial Type Logics (cf. Bernardi 2002; Moortgat 2011; see also Morrill 1994), and the one involving prosodic λ-binding in LCG and related approaches (see Section 2.5.2). In either approach, extraction phenomena are treated by means of some form of hypothetical reasoning, and this raises a major technical issue in the treatment of multiple gap phenomena. The underlying calculus of TLCG is a version of linear logic, and this means that the implication connective is resource sensitive. This is problematic in situations in which a single filler corresponds to multiple gaps, as in parasitic gaps and related phenomena. These cases of extraction require some sort of extension of the underlying logic or some special operator that is responsible for resource duplication. Currently, the most detailed treatment of extraction phenomena in the TLCG literature is Morrill (2017), which lays out in detail an analysis of long-distance dependencies capturing both major island constraints and parasitic gaps within the most recent version of Morrill's Displacement Calculus.

There are several complex issues that arise in relation to the linguistic analysis of extraction phenomena. One major open question is whether island constraints should be accounted for within narrow grammar. Both Steedman and Morrill follow the standard practice in generative grammar research in taking island effects to be syntactic, but this consensus has been challenged by a new body of research in the recent literature proposing various alternative explanations on different types of island constraints (some important work in this tradition includes Deane 1992, Kluender 1998, Hofmeister & Sag 2010 and Chaves & Putnam 2020; see Chaves 2020, Chapter 16 of this volume, Levine 2017 and Newmeyer 2016 for an overview of this line of work and pointers to the relevant literature). Recent syntactic analyses of long-distance dependencies in the HPSG literature explicitly avoid directly encoding major island constraints within the grammar (Sag 2010; Chaves 2012b). Unlike CCG and Displacement Calculus, Kubota & Levine's Hybrid TLCG opts for this latter type of view (that is, the one that is generally in line with recent HPSG work).

Another major empirical problem related to the analysis of long-distance dependencies is the so-called *extraction pathway marking* phenomenon (McCloskey 1979; Zaenen 1983). While this issue received considerable attention in the HPSG literature, through a series of work by Levine and Hukari (see Levine & Hukari 2006), there is currently no explicit treatment of this phenomenon in the CG literature. CCG can probably incorporate the HPSG analysis relatively easily, given the close similarity between the SLASH percolation mechanism and the step-by-step inheritance of the /NP specification in the Function Composition-based approach in CCG. Extraction pathway marking poses a much trickier challenge to TLCG, in which extraction is typically handled by a single-chain movement-like process by means of hypothetical reasoning (but see Kubota & Levine 2020: Chapter 7 for a sketch of a possible approach which mimics successive cyclic movement in the type-logical setup).

4.2 Coordination and ellipsis

Coordination and ellipsis are both major issues in contemporary syntactic theory. There are moreover some phenomena, such as Gapping and Stripping, which seem to lie at the boundary of the two empirical domains (see, for example, the recent overview by Johnson 2019). There are some important similarities and differences between analytic ideas entertained in the HPSG and CG literature for problems in these empirical domains.

4.2.1 Analyses of nonconstituent coordination

CG is perhaps best known in the linguistics literature for its analysis of nonconstituent coordination. Steedman's work on CCG (Steedman 1997; 2000; 2012) in particular has shown how this analysis of coordination interacts smoothly with analyses of other major linguistic phenomena (such as long-distance dependencies, control and raising and quantification) to achieve a surface-oriented grammar that has wide empirical coverage and at the same time has attractive computational properties. Kubota & Levine (2015; 2020) offer an up-to-date TLCG analysis of coordination, and compare it with major alternatives in both the CCG and HPSG literature.

As compared to long-distance dependencies, coordination (in particular NCC) has received considerably less attention in the (H)PSG literature initially (Sag et al. 1985 is an important exception in the early literature). Things started to change somewhat around 2000, with a series of related proposals appearing one after another, including Yatabe (2001), Beavers & Sag (2004), Chaves (2007) and Crysmann (2008) (see Abeillé & Chaves 2020, Chapter 17 of this volume and Nykiel & Kim 2020, Chapter 20 of this volume). Here, I take up Beavers & Sag (2004) and Yatabe (2001) (updated in Yatabe & Tam 2019) as two representative proposals in this line of work. The two proposals share some common assumptions and ideas, but they also differ in important respects.

Both Beavers & Sag (2004) and Yatabe (2001) adopt linearization-based HPSG, together with (a version of) Minimal Recursion Semantics for semantics. Of the two, Beavers & Sag's analysis is more in line with standard assumptions in HPSG. The basic idea of Beavers & Sag's analysis is indeed very simple: by exploiting the flexible mapping between the combinatoric component and the surface word order realization in linearization-based HPSG, they essentially propose a surface deletion-based analysis of NCC according to which NCC examples are analyzed as follows:

(41) [$_S$ Terry gave no man a book on Friday] or [$_S$ ~~Terry gave no man~~ a record on Saturday].

where the material in strike-out is underlyingly present but undergoes deletion in the prosodic representation.

In its simplest form, this analysis gets the scopal relation between the quantifier and coordination wrong in examples like (41) (a well-known problem for the conjunction reduction analysis from the 70s; cf. Partee 1970). Beavers & Sag address this issue by introducing a condition called *Optional Quantifier Merger*:

(42) *Optional Quantifier Merger*: For any elided phrase denoting a generalized quantifier in the domain of either conjunct, the semantics of that phrase may optionally be identified with the semantics of its non-elided counterpart.

As noted by Levine (2011) and Kubota & Levine (2015), this condition does not follow from any general principle and is merely stipulated in Beavers & Sag's account.

Yatabe (2001) and Yatabe & Tam (2019) (the latter of which contains a much more accessible exposition of essentially the same proposal as the former) propose a somewhat different analysis. Unlike Beavers & Sag, who assume that semantic composition is carried out on the basis of the meanings of *signs* on each node (which is the standard assumption about semantic composition in HPSG), Yatabe shifts the locus of semantic composition to the list of domain objects, that is, the component that directly gets affected by the deletion operation that yields the surface string.

This crucially changes the default meaning predicted for examples such as (41). Specifically, on Yatabe's analysis, the surface string for (41) is obtained by the "compaction" operation on word order domains that collapses two quantifiers originally contained in the two conjuncts into one. The semantics of the whole sentence is computed on the basis of this resultant word order domain representation, which contains only *one* instance of a domain object corresponding to the quantifier. The quantifier is then required to scope over the whole coordinate structure due to independently motivated principles of underspecification resolution. While this approach successfully yields the wide-scope reading for quantifiers, the distributive, narrow scope reading for quantifiers (which was trivial for Beavers & Sag) now becomes a challenge. Yatabe & Tam simply stipulate a complex disjunctive constraint on semantic interpretation tied to the "compaction" operation that takes place in coordination so as to generate the two scopal readings.

Kubota & Levine (2015) note that, in addition to the quantifier scope issue noted above, Beavers & Sag's approach suffers from similar problems in the interpretations of symmetrical predicates (*same*, *different*, etc.), summative predicates (*a total of X*, *X in total*, etc.) and the so-called "respective" readings of plural and conjoined expressions (see Chaves 2012a for a lucid discussion of the empirical parallels between the three phenomena and how the basic cases can receive a uniform analysis within HPSG). Yatabe & Tam (2019) offer a response to Kubota & Levine, working out explicit analyses of these more complex phenomena in linearization-based HPSG. A major point of disagreement between Kubota & Levine on the other hand and Yatabe & Tam on the other seems to be whether/to what extent an analysis of a linguistic phenomenon should aim to explain (as opposed to merely account for) linguistic generalizations. There is no easy answer to this question, and it is understandable that different theories put different degrees of emphasis on this goal (see also Borsley & Müller 2020, Chapter 30 of this volume for discussion on a related point). Whatever conclusion one draws from this recent HPSG/CG debate on the treatment of nonconstituent coordination, one point seems relatively uncontroversial: coordination continues to constitute a challenging empirical domain for any grammatical theory, consisting of both highly regular patterns such as systematic interactions with scopal operators (Kubota & Levine 2015; 2020) and puzzling idiosyncrasies, the latter of which includes the summative agreement facts (Postal 1998; Yatabe & Tam 2019) and extraposed relative clauses with split antecedents (Perlmutter & Ross 1970; Yatabe & Tam 2019).

4.2.2 Gapping and Stripping

Descriptively, Gapping is a type of ellipsis phenomenon that occurs in coordination and which deletes some material including the main verb:[23]

[23] There is some disagreement as to whether Gapping is restricted to coordination. Kubota & Levine (2016a), following authors such as Johnson (2009), take Gapping to be restricted to coordination. Park et al. (2019) and Park (2019) take a different view, and argue that Gapping should be viewed as a type of ellipsis phenomenon that is not restricted to coordination envi-

(43) a. Leslie *bought* a CD, and Robin ∅ a book.
b. Terry *can go* with me, and Pat ∅ with you.
c. John *wants to try to begin to write* a novel, and Mary ∅ a play.

Gapping has invoked some theoretical controversy in the recent HPSG/CG literature for the "scope anomaly" issue that it exhibits. The relevant data involving auxiliary verbs such as (44a) and (44b) have long been known in the literature since Oehrle (1971; 1987) and Siegel (1987). McCawley (1993) later pointed out similar examples involving downward-entailing quantifiers such as (44c).

(44) a. Mrs. J can't live in Boston and Mr. J ∅ in LA.
b. Kim didn't play bingo or Sandy ∅ sit at home all evening.
c. No dog eats Whiskas or ∅ cat ∅ Alpo.

The issue here is that (44a), for example, has a reading in which the modal *can't* scopes over conjunction ('it's not possible for Mrs. J to live in NY and Mr. J to live in LA at the same time'). This is puzzling, since such a reading wouldn't be predicted on the (initially plausible) assumption that Gapping sentences would be interpreted by simply supplying the meaning of the missing material in the right conjunct.

Kubota & Levine (2014; 2016a) note some difficulties for earlier accounts of Gapping in the (H)PSG literature (Sag et al. 1985; Abeillé et al. 2014) and argue for a constituent coordination analysis of Gapping in TLCG, building on earlier analyses of Gapping in CG (Steedman 1990; Hendriks 1995b; Morrill & Solias 1993). The key idea of Kubota & Levine's analysis involves taking Gapping as coordination of clauses missing a verb in the middle, which can be transparently represented as a function from strings to strings of category $\mathrm{S}{\upharpoonright}((\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP})$:

(45) $\lambda\varphi.\mathsf{leslie} \circ \varphi \circ \mathsf{a} \circ \mathsf{cd}; \lambda R.\exists x.\mathbf{cd}(x) \wedge R(x)(\mathbf{l}); \mathrm{S}{\upharpoonright}((\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP})$

A special type of conjunction entry (prosodically of type $(\mathbf{st}{\to}\mathbf{st}){\to}(\mathbf{st}{\to}\mathbf{st}){\to}(\mathbf{st}{\to}\mathbf{st})$) then conjoins two such expressions and returns a conjoined sentence missing the verb only in the first conjunct (on the prosodic representation). By feeding the verb to this resultant expression, a proper form-meaning pair is obtained for Gapping sentences like those in (43).

The apparently unexpected wide scope readings for auxiliaries and quantifiers in (44) turn out to be straightforward on this analysis. I refer the interested reader to Kubota & Levine (2016a) for details, but the key idea is that the apparently

ronments. See Kubota & Levine (2020: 46–47) for a response to Park et al. (2019).

anomalous scope in such examples isn't really anomalous on this approach, since the auxiliary (which prosodically lowers into the first conjunct) takes the whole conjoined gapped clause as its argument in the combinatoric component underlying semantic interpretation.[24] Thus, the existence of the wide scope reading is automatically predicted. Puthawala (2018) extends this approach to a similar "scope anomaly" data found in Stripping, in examples such as the following:

(46) John didn't sleep, or Mary (either).

Just like the Gapping examples in (44), this sentence has both wide scope ('neither John nor Mary slept') and narrow scope ('John was the one who didn't sleep, or maybe that was Mary') interpretations for negation.

The determiner gapping example in (44c) requires a somewhat more elaborate treatment. Kubota & Levine (2016a) analyze determiner gapping via higher-order functions. Morrill & Valentín (2017) criticize this approach for a certain type of overgeneration problem regarding word order and propose an alternative analysis in Displacement Calculus.

Park et al. (2019) and Park (2019) propose an analysis of Gapping in HPSG that overcomes the limitations of previous (H)PSG analyses of Gapping (Sag et al. 1985; Chaves 2005; Abeillé et al. 2014), couched in Lexical Resources Semantics. In Park et al.'s analysis, the lexical entries of the clause-level conjunction words *and* and *or* are underspecified as to the relative scope between the propositional operator contributed by the modal auxiliary in the first conjunct and the boolean conjunction or disjunction connective that is contributed by the conjunction word itself. Park et al. argue that this is sufficient for capturing the scope anomaly in the Oehrle/Siegel data such as (44a) and (44b). Extension to the determiner gapping case (44c) is left for future work.

Here again, instead of trying to settle the debate, I'd like to draw the reader's attention to the different perspectives on grammar that seem to be behind the HPSG and (Hybrid) TLCG approaches. Kubota & Levine's approach attains theoretical elegance at the cost of employing abstract higher-order operators (both in semantics and prosody). This makes the relationship between the competence grammar and the on-line human sentence processing model indirect, and relatedly, it is likely to make efficient computational implementation less straightforward (for a discussion on the relationship between competence grammar and a model of sentence processing, see Wasow 2020, Chapter 25 of this volume and Borsley & Müller 2020, Chapter 30 of this volume). Park et al.'s (2019) approach, on the other hand, is more in line with the usual practice (and the shared spirit)

[24] This is essentially a formalization of an idea that goes back to Siegel's (1987) work.

of HPSG research, where the main emphasis is on writing an explicit grammar fragment that is constraint-based and surface-oriented. This type of tension is perhaps not easy to overcome, but it seems useful (for researchers working in different grammatical theories) to at least recognize (and appreciate) the existence of these different theoretical orientations tied to different approaches.

4.2.3 Ellipsis

Analyses of major ellipsis phenomena in HPSG and CG share the same essential idea that ellipsis is a form of anaphora, without any invisible hierarchically structured representations corresponding to the "elided" expression. See Nykiel & Kim (2020), Chapter 20 of this volume and Ginzburg & Miller (2019) for an overview of approaches to ellipsis in HPSG.

Recent analyses of ellipsis in HPSG (Ginzburg & Sag 2000; Miller 2014) make heavy use of the notion of "construction" adopted from Construction Grammar (this idea is even borrowed into some of the CG analyses of ellipsis such as Jacobson 2016). Many ellipsis phenomena are known to exhibit some form of syntactic sensitivity (Kennedy 2003; Chung 2013; Yoshida et al. 2015), and this fact has long been taken to provide strong evidence for the "covert structure" analyses of ellipsis popular in mainstream generative grammar (Merchant 2019).

Some of the early works on ellipsis in CG include Hendriks (1995a) and Morrill & Merenciano (1996). Morrill & Merenciano (1996) in particular show how hypothetical reasoning in TLCG allows treatments of important properties of ellipsis phenomena such as strict/sloppy ambiguity and scope ambiguity of elided quantifiers in VP ellipsis. Jäger (2005) integrates these earlier works with a general theory of anaphora in TLCG, incorporating the key empirical analyses of pronominal anaphora by Jacobson (1999; 2000). Jacobson's (1998; 2008) analysis of Antecedent-Contained Ellipsis is also important. Antecedent-Contained Ellipsis is often taken to provide a strong piece of evidence for the representational analysis of ellipsis in mainstream generative syntax. Jacobson offers a counterproposal to this standard analysis that completely dispenses with covert structural representations. While the above works from the 90s have mostly focused on VP ellipsis, recent developments in the CG literature, including Barker (2013) on sluicing, Jacobson (2016) on fragment answers and Kubota & Levine (2017) on pseudogapping, considerably extended the empirical coverage of the same line of analysis.

The relationship between recent CG analyses of ellipsis and HPSG counterparts seems to be similar to the situation with competing analyses on coordination. Both Barker (2013) and Kubota & Levine (2017) exploit hypothetical reasoning to treat the antecedent of an elided material as a "constituent" with full-fledged semantic interpretation at an abstract combinatoric component of syntax. The anaphoric mechanism can then refer to both the syntactic and semantic information of the antecedent expression to capture syntactic sensitivity observed in ellipsis phenomena, without the need to posit hierarchical representations at the ellipsis site. Due to its surface-oriented nature, HPSG is not equipped with an analogous abstract combinatoric component that assigns "constituent" status to expressions that do not (in any obvious sense) correspond to constituents in the surface representation. In HPSG, the major work in restricting the possible form of ellipsis is instead taken over by constructional schemata, which can encode syntactic information of the antecedent to capture connectivity effects, as is done, for example, with the use of the SAL-UTT feature in Ginzburg & Sag's (2000) analysis of sluicing (cf. Nykiel & Kim 2020, Chapter 20 of this volume).

Kubota & Levine (2020: Chapter 8) extend Kubota & Levine's (2017) approach further to the treatment of interactions between VP ellipsis and extraction, which has often been invoked in the earlier literature (in particular, Kennedy 2003) as providing crucial evidence for covert structure analysis of ellipsis phenomena (see also Jacobson 2019 for a related proposal, cast in a variant of CCG). At least some of the counterproposals that Kubota & Levine formulate in their argument against the covert structure analysis seem to be directly compatible with the HPSG approach to ellipsis, but (so far as I am aware) no concrete analysis of extraction/ellipsis interaction currently exists in the HPSG literature.

4.2.4 Mismatches in right-node raising

While right-node raising (RNR) has mostly been discussed in connection to coordination in the literature, it is well-known that RNR is not necessarily restricted to coordination environments (see, for example, Wilder 2019 for a recent overview). Moreover, it has recently been pointed out by Abeillé et al. (2016); Shiraïshi et al. (2019) that RNR admits certain types of syntactic mismatch between the RNR'ed material and the selecting head in a non-adjacent conjunct. The current literature seems to agree that RNR is not a unitary phenomenon, and that at least some type of RNR should be treated via a mechanism of surface ellipsis, which could be modelled as deletion of syntactic (or prosodic) objects or via some sort of anaphoric mechanism (cf. Nykiel & Kim 2020, Chapter 20 of this volume, Chaves 2014, Kubota & Levine 2017, Shiraïshi et al. 2019).

One point that is worth emphasizing in this connection is that while the "NCC as constituent coordination" analysis of RNR in CG discussed in Section 2.4.1 (major evidence for which comes from the interactions between various sorts of scopal operators and RNR as noted in Section 4.2.1) is well-known, neither CCG nor TLCG is by any means committed to the idea that *all* instances of RNR should be analyzed this way. In fact, given the extensive evidence for the non-unitary nature of RNR reviewed in Chaves (2014) and the syntactic mismatch data from French offered by Abeillé et al. (2016); Shiraïshi et al. (2019), it seems that a comprehensive account of RNR in CG (or, for that matter, in any other theory) would need to recognize the non-unitary nature of the phenomenon, along lines similar to Chaves's (2014) recent proposal in HPSG. While there is currently no detailed comprehensive account of RNR along these lines in the CG literature, there does not seem to be any inherent obstacle for formulating such an account.

4.3 Binding

Empirical phenomena that have traditionally been analyzed by means of Binding Theory (both in the transformational and the non-transformational literature; cf. Müller & Branco 2020, Chapter 21 of this volume) potentially pose a major challenge to the "non-representational" view of the syntax-semantics interface common to most variants of CG. The HPSG Binding Theory in Pollard & Sag (1992; 1994) captures Principles A and B at the level of argument structure, while Principle C makes reference to the configurational structure (i.e. the feature-structure encoding of the constituent geometry). The status of Principle C itself is controversial to begin with, but if this condition needs to be stated in the syntax, it would possibly constitute one of the greatest challenges to CG-based theories of syntax, since, unlike phrase structure trees, the proof trees in CG are not objects that a principle of grammar can directly refer to.

While there seems to be no consensus in the current CG literature on how the standard facts about binding theory are to be accounted for, there are some important ideas and proposals in the wider literature of CG-based syntax (broadly construed to include work in the Montague Grammar tradition). First, as for Principle A, there is a recurrent suggestion in the literature that these effects can (and should) be captured simply via strictly lexical properties of reflexive pronouns (e.g. Szabolcsi 1992; see Büring 2005 for a concise summary). For example, for a reflexive in the direct object position of a transitive verb bound by the subject NP, the following type assignment (where the reflexive pronoun first takes a transitive verb and then the subject NP as arguments) suffices to capture its bound status:

(47) himself; $\lambda R \lambda x.R(x)(x)$; ((NP\S)/NP)\NP\S

This approach is attractively simple, but there are at least two things to keep in mind, in order to make it a complete analysis of Principle A in CG. First, while this lexical treatment of reflexive binding may at first sight appear to capture the locality of binding quite nicely, CG's flexible syntax potentially overgenerates unacceptable long-distance binding readings for (English) reflexives. Since RNR can take place across clause boundaries, it seems necessary to assume that hypothetical reasoning for the Lambek-slash (or a chain of Function Composition that has the same effect in CCG) can generally take place across clause boundaries. But then, expressions such as *thinks Bill hates* can be assigned the same syntactic type (i.e. (NP\S)/NP) as lexical transitive verbs, overgenerating non-local binding of a reflexive from a subject NP in the upstairs clause (* *John*$_i$ *thinks Bill hates himself*$_i$).

In order to prevent this situation while still retaining the lexical analysis of reflexivization sketched above, some kind of restriction needs to be imposed as to the way in which reflexives combine with other linguistic expressions. One possibility would be to distinguish between lexical transitive verbs and derived transitive verb-like expressions by positing different "modes of composition" in the two cases in a "multi-modal" version of CG.

The other issue is that the lexical entry in (47) needs to be generalized to cover all cases in which a reflexive is bound by an argument that is higher in the obliqueness hierarchy. This amounts to positing a polymorphic lexical entry for the reflexive. The use of polymorphism is not itself a problem, since it is needed in other places in the grammar (such as coordination) anyway. But this account would amount to capturing the Principle A effects purely in terms of the specific lexical encoding for reflexive pronouns (unlike the treatment in HPSG which explicitly refers to the obliqueness hierarchy).

While Principle A effects are in essence amenable to a relatively simple lexical treatment along lines sketched above, Principle B turns out to be considerably more challenging for CG. To see this point, note that the lexical analysis of reflexives sketched above crucially relies on the fact that the constraint associated with reflexives corresponds to a straightforward semantic effect of variable binding. Pronouns instead require *disjointness* of reference from less oblique coarguments, but such an effect cannot be captured by simply specifying some appropriate lambda term as the semantic translation for the pronoun.

To date, the most detailed treatment of Principle B effects in CG that explicitly addresses this difficulty is the proposal by Jacobson (2007), formulated in a version of CCG (Steedman 1997 proposes a different approach to binding, which will be briefly discussed at the end of this section). The key idea of Jacobson's account of Principle B effects is that NPs are divided by a binary-valued feature ±p, with pronouns marked NP[+p] and all other NPs NP[−p]. In all lexical entries of the form in (48), all NP (and PP) arguments in any realization of /$ are specified as [−p].[25]

(48) k; P; VP/$

The effect of this restriction is to rule out pronouns from argument positions of verbs with ordinary semantic denotations. On this approach, the only way a lexically specified functional category can take [+p] arguments is via the application of the following irreflexive operator:[26]

(49) $\lambda\varphi.\varphi$; $\lambda f \lambda u \lambda v.f(u)(v)$, $u \neq v$; (VP/NP[+p])↾(VP/NP[−p])

The greyed-in part $u \neq v$ separated from the truth conditional meaning by a comma is a presupposition introduced by the pronoun-seeking variant of the predicate. It says that the subject and object arguments are forced to pick out different objects in the model. For the semantics of pronouns themselves, one can assume, following the standard practice, that free (i.e. unbound) pronouns are simply translated as arbitrary variables (cf. Cooper 1979).

[25] Here, /$ is an abbreviation of a sequence of argument categories sought via /. Thus, VP/$ can be instantiated as VP/NP, VP/NP/NP, VP/PP/NP, etc.

[26] For expository purposes, I state the operator in (49) in its most restricted form, dealing with only the case where there is a single syntactic argument apart from the subject. A much broader coverage is of course necessary in order to handle cases like the following:

(i) a. * John$_i$ warned Mary about him$_i$.
b. * John talked to Mary$_i$ about her$_i$.
c. * John explained himself$_i$ to him$_i$.

What is needed in effect is a schematic type specification that applies to a pronoun in any or all argument positions, i.e., stated on an input of the form VP/$/XP[−p]/$ to yield an output of the form VP/$/XP[+p]/$. To ensure the correct implementation of this extension, some version of the "wrapping" analysis needs to be assumed (cf. Jacobson 2007), so that the order of the arguments in verbs' lexical entries is isomorphic to the obliqueness hierarchy (of the sort discussed by Pollard & Sag 1992).

Cases such as the following also call for an extension (also a relatively straightforward one):

(ii) * John$_i$ is proud of him$_i$.

By assuming (following Jacobson 2007) that the [±p] feature percolates from NPs to PPs and by generalizing the irreflexive operator still further so that it applies not just to VP/XP[−p] but to AP/XP[−p] as well, the ungrammaticality of (ii) follows straightforwardly.

Crucially, the operator in (49) is restricted in its domain of application to the set of signs which are specified in the lexicon. I notate this restriction by using the dashed line notion in what follows. Then (50) will be derived as in (51).

(50) John praises him.

(51)
$$\dfrac{\dfrac{\dfrac{\begin{matrix}\lambda\varphi.\varphi; \\ \lambda f\lambda u\lambda v.f(u)(v), u \neq v; \\ (\mathrm{VP}/\mathrm{NP}[+\mathrm{p}])\upharpoonright(\mathrm{VP}/\mathrm{NP}[-\mathrm{p}])\end{matrix} \quad \begin{matrix}\text{praises;} \\ \mathbf{praise}; \mathrm{VP}/\mathrm{NP}[-\mathrm{p}]\end{matrix}}{\text{praises}; \lambda u\lambda v.\mathbf{praise}(u)(v), u \neq v; \mathrm{VP}/\mathrm{NP}[+\mathrm{p}]} \quad \text{him}; z; \mathrm{NP}[+\mathrm{p}]}{\text{praises} \circ \text{him}; \lambda v.\mathbf{praise}(z)(v), z \neq v; \mathrm{VP}} \quad \text{john}; \mathbf{j}; \mathrm{NP}[-\mathrm{p}]}{\text{john} \circ \text{praises} \circ \text{him}; \mathbf{praise}(z)(\mathbf{j}), z \neq \mathbf{j}; \mathrm{S}}$$

The presupposition $z \neq \mathbf{j}$ ensures that the referent of the pronoun is different from John.

Thus, Jacobson's approach captures the relevant conditions on the interpretation of pronouns essentially as a type of lexical presupposition tied to the denotation of the pronoun-taking verb, and the syntactic feature [±p] mediates the distributional correlation between the pronoun and the verb that subcategorizes for it. The idea is essentially the same as in the HPSG Binding Theory, except that the relevant condition is directly encoded as a restriction on the denotation itself, since the standard CG syntax-semantics interface does not admit of syntactic indices of the sort assumed in HPSG.

Unlike Jacobson's proposal outlined above, Steedman's (1997) analysis of binding conditions in CCG recognizes the syntactic forms of the logical language that is used to write the denotations of linguistic expressions as the "level" at which binding conditions are stated. This approach can be thought of as a "compromise" which enables a straightforward encoding of the HPSG-style Binding Conditions by (slightly) deviating from the CG doctrine of not admitting any representational object at the syntax-semantics interface (see Dowty 1997 for a critique of the approach to binding by Steedman 1997 discussing this issue clearly).

Steedman's approach can be best illustrated by taking a look at the analysis of (52).[27]

(52) * Every student$_i$ praised him$_i$.

[27] At the same time that he formulates an essentially syntactic account of Principle B via the term **pro** in the translation language, Steedman (1997) briefly speculates on the (somewhat radical) possibility of relegating Principle B entirely to the pragmatic component of pronominal anaphora resolution. However, the relevant discussion is rather sketchy, and the details of such a pragmatic alternative are not entirely clear.

According to Steedman, pronouns receive translations of the form **pro**(x), where **pro** is effectively a term that marks the presence of (the translation of) a pronoun at some particular syntactic position in the logical formula that represents the meaning of the sentence.

With this assumption, the translation for (52) that needs to be ruled out (via Principle B) is as follows:

(53) $\forall x[\mathbf{student}(x) \to \mathbf{praise}(\mathbf{pro}(x))(x)]$

And this is where the CCG Binding Theory kicks in. The relevant part of the structure of the logical formula in (53) can be more perspicuously written as a tree as in Figure 1, which makes clear the hierarchical relation between subterms.

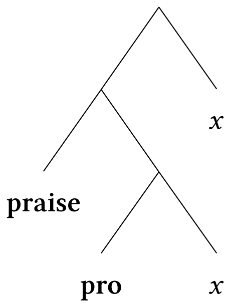

Figure 1: Logical formula as a tree

Principle B states that pronouns need to be locally free. (??) violates this condition since there is a locally c-commanding term x that binds **pro**(x) (where a term α binds term β when they are semantically bound by the same operator).

Principles A and C are formulated similarly by making crucial reference to the structures of the terms that represent the semantic translations of sentences.

What one can see from the comparison of different approaches to binding in CG and the treatment of binding in HPSG is that although HPSG and CG are both lexicalist theories of syntax, and there is a general consensus that binding conditions are to be formulated lexically rather than configurationally, there are important differences in the actual implementations of the conditions between approaches that stick to the classical Montagovian tradition (embodying the tenet of “direct compositionality” in Jacobson’s terms) and those that make use of (analogues of) representational devices more liberally.

Finally, some comments are in order regarding the status of Principle C, the part of Binding Theory that is supposed to rule out examples such as the following:

(54) a. * He$_i$ talked to John$_i$.

b. * He$_i$ talked to John's$_i$ brother.

The formulation of Principle C has always been a problem in lexicalist theories of syntax. While Principles A and B can be stated by just making reference to the local argument structure of a predicate in the lexicon, the global nature of Principle C seems to require looking at the whole configurational structure of the sentence in which the proper noun appears. In fact, Pollard & Sag (1992; 1994) opt for this solution, and their definition of the Principle C has a somewhat exceptional status within the whole theory (which otherwise adheres to strict locality conditions) in directly referring to the configurational structure.

Essentially the same problem arises in CG. Steedman's (1997) formulation of Principle C can be thought of as an analog of Pollard & Sag's (1992; 1994) proposal, where global reference to hierarchical structure is made not at the level of phrase structure, but instead at the level of "logical structure", that is, in the syntactic structure of the logical language used for writing the meanings of natural language expressions. As already noted above, if one takes the Montagovian, or "direct compositional", view of the syntax-semantics interface that is more traditional/standard in CG research, this option is unavailable.

Thus, Principle C has a somewhat cumbersome place within lexicalist theories in general. However, unlike Principles A and B, the status of Principle C in the grammar is still considerably unclear and controversial to begin with (see Büring 2005: 122–124 for some discussion on this point). In particular, it has been noted in the literature (Lasnik 1986) that there are languages such as Thai and Vietnamese that do not show Principle C effects. If, as suggested by some authors (cf., e.g., Levinson 1987; 1991), the effects of Principle C can be accounted for by pragmatic principles, that would remove one major sticking point in both HPSG and CG formulations of the Binding Theory.

5 A brief note on processing and implementation

The discussion above has mostly focused on linguistic analysis. In this final section, I will briefly comment on implications for psycholinguistics and computational linguistics research.

As should already be clear from the above discussion, different variants of both HPSG and CG make different assumptions about the relationship between the competence grammar and theories of performance. To make things even more complicated, such assumptions are often implicit. As a first approximation, it is probably fair to say that HPSG (at least the "bare-bones" version of it) and CCG are more similar to each other than they are to TLCG in being surface-oriented. TLCG makes heavy use of hypothetical reasoning in the analyses of certain linguistic phenomena, and, as should already be clear at this point, the role it plays in the grammar is much like the role of movement operations in mainstream generative grammar.

As repeatedly emphasized by practitioners of HPSG and CCG (see, for example, Sag & Wasow 2011, Steedman 2012 and Wasow 2020, Chapter 25 of this volume), all other things being equal, it is more preferable to make the relationship between the competence grammar and the model of performance as transparent as possible. It is unlikely that any reasonable researcher would deny such a claim, but it begs one big question: how exactly are we to understand the qualification "all other things being equal"? Practitioners of TLCG in general seem to have a somewhat more detached take on the relationship between competence and performance, and I believe the consensus there is more in line with (what seems to be) the spirit of mainstream generative grammar: the goal is to clarify the most fundamental principles of grammar and state them in the simplest form possible. TLCG subscribes to the thesis that (a certain variety of) logic is indeed the underlying principle of grammar of natural language. This is an attractive view, but at the same time language exhibits phenomena that suggest that pushing this perspective to the limit is unlikely to be the most fruitful research strategy. The right approach is probably one that combines the insights of both surface-oriented approaches (such as HPSG and CCG) and more abstract approaches (such as TLCG and mainstream generative grammar).

At a more specific level, one attractive feature of CCG (but not CG in general), when viewed as an integrated model of the competence grammar and human sentence processing, is that it enables surface-oriented, incremental analyses of strings from left to right. This aspect was emphasized in the early literature of CCG (Ades & Steedman 1982; Crain & Steedman 1985), but it does not seem to have had much impact on psycholinguistic research in general since then. A notable exception is the work by Pickering & Barry (1991; 1993) in early 90s. There is also some work on the relationship between processing and TLCG (see Morrill 2010: Chapters 9 and 10, and references therein). In any event, a serious investigation of the relationship between competence grammar and human sentence processing from a CG perspective (either CCG or TLCG) is a research topic that is waiting to be explored, much like the situation with HPSG (see Wasow 2020, Chapter 25 of this volume).

As for connections to computational linguistics (CL)/natural language processing (NLP) research, like HPSG (cf. Bender & Emerson 2020, Chapter 26 of this volume), large-scale computational implementation has been an important research agenda for CCG (see, for example, White & Baldridge 2003; Clark & Curran 2007). I refer the reader to Steedman (2012: Chapter 13) for an excellent summary on this subject (this chapter contains a discussion of human sentence processing as well). Together with work on linguistically informed parsing in HPSG, CCG parsers seem to be attracting some renewed interest in CL/NLP research recently, due to the new trend of combining the insights of statistical approaches and linguistically-informed approaches. In particular, the straightforward syntax-semantics interface of (C)CG is an attractive feature in building CL/NLP systems that have an explicit logical representation of meaning. See, for example, Lewis & Steedman (2013) and Mineshima et al. (2016) for this type of work. TLCG research has traditionally been less directly related to CL/NLP research. But there are recent attempts at constructing large-scale treebanks (Moot 2015) and combining TLCG frameworks with more mainstream approaches in NLP research such as distributional semantics (Moot 2018).

6 Conclusion

As should be clear from the above discussion, HPSG and CG share many important similarities, mainly due to the fact that they are both variants of lexicalist syntactic theories. This is particularly clear in the analyses of local dependencies in terms of lexically encoded argument structure information. Important differences emerge once one turns one's attention to less canonical types of phenomena, such as atypical types of coordination (nonconstituent coordination, Gapping) and the treatment of "constructional" patterns that are not easily lexicalizable. In general, HPSG has a richer and more comprehensive treatment of various empirical phenomena, whereas CG has a lot to offer to grammatical theory (perhaps somewhat paradoxically) due to the very fact that the potentials of the logic-based perspective it embodies has not yet been explored in full detail. It is more likely than not that the two will continue to develop as distinct theories of natural language syntax (and semantics). I hope that the discussion in the present chapter has made it clear that there are still many occasions for fruitful interactions between the two approaches both at the level of analytic ideas for specific empirical phenomena and at the more general, foundational level pertaining to the overall architecture of grammatical theory.

Acknowledgements

I'd like to thank Jean-Pierre Koenig, Bob Levine and Stefan Müller for comments.

References

Abeillé, Anne. 2020. Control and Raising. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 353–382. Berlin: Language Science Press. DOI:??

Abeillé, Anne, Gabriela Bîlbîie & Francois Mouret. 2014. A Romance perspective on Gapping constructions. In H. Boas & F. Gonzálvez García (eds.), *Romance in construction grammar*, 227–265. Amsterdam: John Benjamins.

Abeillé, Anne & Rui P. Chaves. 2020. Coordination. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 521–561. Berlin: Language Science Press. DOI:??

Abeillé, Anne, Berthold Crysmann & Aoi Shiraıshi. 2016. Syntactic mismatch in french peripheral ellipsis. In Christopher Piñón (ed.), *Empirical issues in syntax and semantics*, 1–30. Paris: CSSP.

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

Ajdukiewicz, Kazimierz. 1935. Die syntaktische Konnexität. *Studia Philosophica* 1. 1–27.

Bach, Emmon. 1979. Control in Montague Grammar. *Linguistic Inquiry* 4(10). 515–531.

Bach, Emmon. 1980. In defense of passive. *Linguistics and Philosophy* (3). 297–342.

Baldridge, Jason. 2002. *Lexically specified derivational control in Combinatory Categorial Grammar*. University of Edinburgh dissertation.

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

Barker, Chris. 2007. Parasitic scope. *Linguistics and Philosophy* 30(4). 407–444.

Barker, Chris. 2013. Scopability and sluicing. *Linguistics and Philosophy* 36(3). 187–223.

Barker, Chris & Chung-chieh Shan. 2015. *Continuations and natural language*. Oxford: Oxford University Press.

Bayer, Samuel. 1996. The coordination of unlike categories. *Language* 72(3). 579–616.

Bayer, Samuel & Mark Johnson. 1995. Features and agreement. In Hans Uszkoreit (ed.), *33rd Annual Meeting of the Association for Computational Linguistics. Proceedings of the conference*, 70–76. Cambridge, MA: Association for Computational Linguistics.

Beavers, John & Ivan A. Sag. 2004. Coordinate ellipsis and apparent non-constituent coordination. In Stefan Müller (ed.), *Proceedings of the 11th International Conference on Head-Driven Phrase Structure Grammar, Center for Computational Linguistics, Katholieke Universiteit Leuven*, 48–69. Stanford, CA: CSLI Publications. http://csli-publications.stanford.edu/HPSG/2004/, accessed 2018-2-25.

Bekki, Daisuke. 2014. Representing anaphora with dependent types. In Nicholas Asher & Sergei Soloviev (eds.), *Logical Aspects of Computational Linguistics 2014*, 14–29. Heidelberg: Springer.

Bekki, Daisuke & Koji Mineshima. 2017. Context-passing and underspecification in dependent type semantics. In Stergios Chatzikyriakidis & Zhaohui Luo (eds.), *Modern perspectives in type-theoretical semantics*, 11–41. Heidelberg: Springer.

Bender, Emily M. & Guy Emerson. 2020. Computational linguistics and grammar engineering. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 813–839. Berlin: Language Science Press. DOI:??

Bernardi, Raffaella. 2002. *Reasoning with polarity in Categorial Type Logic*. University of Utrecht dissertation.

Borsley, Robert D. & Kersti Börjars (eds.). 2011. *Non-transformational syntax: Formal and explicit models of grammar: A guide to current models*. Oxford, UK/Cambridge, MA: Blackwell Publishers Ltd.

Borsley, Robert D. & Berthold Crysmann. 2020. Unbounded Dependencies. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 383–429. Berlin: Language Science Press. DOI:??

Borsley, Robert D. & Stefan Müller. 2020. HPSG and Minimalism. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 913–965. Berlin: Language Science Press. DOI:??

Bresnan, Joan, Ash Asudeh, Ida Toivonen & Stephen Mark Wechsler. 2016. *Lexical-functional syntax*. 2nd edn. Oxford: Blackwell Publishers Ltd. DOI:10.1002/9781119105664

Bruening, Benjamin. 2018a. Brief response to Müller. *Language* 94(1). e67–e73. DOI:10.1353/lan.2018.0015

Bruening, Benjamin. 2018b. The lexicalist hypothesis: Both wrong and superfluous. *Language* 94(1). 1–42. DOI:10.1353/lan.2018.0000

Büring, Daniel. 2005. *Binding theory*. Cambridge, England: Cambridge University Press.

Calder, Jonathan, Ewan Klein & Henk Zeevat. 1988. Unification Categorial Grammar: A concise, extendable grammar for natural language processing. In *Proceedings of the 12th conference on computational linguistics*, vol. 1, 83–86. Stroudsburg, PA, USA: Association for Computational Linguistics.

Carpenter, Bob. 1998. *Type-logical semantics*. Cambridge, MA: MIT Press.

Chatzikyriakidis, Stergios & Zhaohui Luo (eds.). 2017. *Modern perspectives in type-theoretical semantics*. Heidelberg: Springer.

Chaves, Rui. 2012a. Conjunction, cumulation and respectively readings. *Journal of Linguistics* 48(2). 297–344.

Chaves, Rui. 2012b. On the grammar of extraction and coordination. *Natural Language and Linguistic Theory* 30(2). 465–512.

Chaves, Rui. 2020. Island phenomena and related matters. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 483–519. Berlin: Language Science Press. DOI:??

Chaves, Rui P. 2014. On the disunity of right-node raising phenomena: Extraposition, ellipsis, and deletion. *Language* 4. 834–886. DOI:10.1353/lan.2014.0081

Chaves, Rui P. & Michael T. Putnam. 2020. *Unbounded dependency constructions: Theoretical and experimental perspectives*. Oxford: Oxford University Press.

Chaves, Rui Pedro. 2005. A linearization-based approach to gapping. In G. Jäger, P. Monachesi, G. Penn & S. Wintner (eds.), *FG-MOL 2005: the 10th conference on Formal Grammar and the 9th meeting on mathematics of language*, 207–220. University of Edinburgh, Scotland.

Chaves, Rui Pedro. 2007. *Coordinate structures – constraint-based syntax-semantics processing*. Portugal: University of Lisbon dissertation. https://www.acsu.buffalo.edu/~rchaves/diss_rpc.pdf, accessed 2020-2-25.

Chung, Sandra. 2013. Syntactic identity in sluicing: how much, and why. *Linguistic Inquiry* 44(1). 1–44.

Clark, Stephen & James Curran. 2007. Wide-coverage efficient statistical parsing with CCG and log-linear models. *Computational Linguistics* 33(4). 493–552.

Cooper, Robin. 1979. The interpretation of pronouns. In Frank Heny & Helmut S. Schnelle (eds.), *Selections from the third groningen round table*, vol. 10 (Syntax and Semantics), 61–92. New York: Academic Press.

Copestake, Ann, Daniel P. Flickinger, Carl J. Pollard & Ivan A. Sag. 2005. Minimal Recursion Semantics: An introduction. *Research on Language and Computation* 3(2–3). 281–332. DOI:10.1007/s11168-006-6327-9

Crain, Stephen & Mark J. Steedman. 1985. On not being led up the garden path: The use of context by the psychological syntax processor. In David R. Dowty, Lauri Karttunen & Arnold M. Zwicky (eds.), *Natural language processing*, 320–358. Cambridge, UK: Cambridge University Press.

Crysmann, Berthold. 2008. An asymmetric theory of peripheral sharing in HPSG: Conjunction reduction and coordination of unlikes. In Gerhard Jäger, Paola Monachesi, Gerald Penn & Shuly Wintner (eds.), *Proceedings of Formal Grammar 2003, Vienna, Austria*, 47–62. Stanford, CA: CSLI Publications.

Crysmann, Berthold. 2020. Morphology. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 685–728. Berlin: Language Science Press. DOI:??

Curry, Haskell B. 1961. Some logical aspects of grammatical structure. In Roman Jakobson (ed.), *Structure of language and its mathematical aspects*, vol. 12 (Symposia on Applied Mathematics), 56–68. Providence: American Mathematical Society.

Daniels, Michael W. 2002. On a type-based analysis of feature neutrality and the coordination of unlikes. In Frank van Eynde, Lars Hellan & Dorothee Beermann (eds.), *Proceedings of the HPSG 2001 conference*, 137–147. Stanford, CA: CSLI Publications.

Davis, Anthony & Jean-Pierre Koenig. 2020. The nature and role of the lexicon in HPSG. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 65–106. Berlin: Language Science Press. DOI:??

de Groote, Philippe. 2001. Towards abstract Categorial Grammars. In *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter*, 148–155.

Deane, Paul D. 1992. *Grammar in mind and brain*. Berlin: Mouton de Gruyter.

Dowty, David. 1982a. Grammatical relations and Montague Grammar. In Pauline Jacobson & Geoffrey K. Pullum (eds.), *The nature of syntactic representation* (Synthese Language Library 15). Dordrecht: D. Reidel Publishing Company.

Dowty, David R. 1982b. More on the categorial analysis of grammatical relations. In Annie Zaenen (ed.), *Subjects and other subjects: Proceedings of the Harvard*

Conference on the Representation of Grammatical Relations, 115–153. Bloomington: Indiana University Linguistics Club.

Dowty, David R. 1988. Type raising, functional composition, and nonconstituent coordination. In Richard Oehrle, Emmon Bach & Deirdre Wheeler (eds.), *Categorial Grammars and natural language structures*, 153–198. Dordrecht: D. Reidel Publishing Company.

Dowty, David R. 1996. Toward a Minimalist theory of syntactic structure. In Harry Bunt & Arthur van Horck (eds.), *Discontinuous constituency* (Natural Language Processing 6), 11–62. Berlin: Mouton de Gruyter. Published version of a Ms. dated January 1990.

Dowty, David. 1997. Non-constituent coordination, wrapping, and Multimodal Categorial Grammars: Syntactic form as logical form. In Maria Luisa Dalla Chiara, Kees Doets, Daniele Mundici & Johan Van Benthem (eds.), *Structures and norms in science* (Synthese Library 260), 347–368. Springer Verlag. DOI:10.1007/978-94-017-0538-7_21

Fillmore, Charles J. 1999. Inversion and Constructional inheritance. In Gert Webelhuth, Jean-Pierre Koenig & Andreas Kathol (eds.), *Lexical and Constructional aspects of linguistic explanation* (Studies in Constraint-Based Lexicalism 1), 113–128. Stanford, CA: CSLI Publications.

Fillmore, Charles J., Paul Kay & Mary Catherine O'Connor. 1988. Regularity and idiomaticity in grammatical constructions: The case of *let alone*. *Language* 64(3). 501–538.

Flickinger, Dan, Carl Pollard & Tom Wasow. 2020. The evolution of HPSG. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 7–30. Berlin: Language Science Press. DOI:??

Geach, Peter Thomas. 1970. A program for syntax. *Synthese* 22. 3–17.

Ginzburg, Jonathan & Philip Miller. 2019. Ellipsis in head-driven phrase structure grammar. In Jeroen van Craenenbroeck & Tanja Temmerman (eds.), *The oxford handbook of ellipsis*, 75–121. Oxford: Oxford University Press.

Ginzburg, Jonathan & Ivan A. Sag. 2000. *Interrogative investigations: The form, meaning, and use of English interrogatives* (CSLI Lecture Notes 123). Stanford, CA: CSLI Publications.

Godard, Danièle & Pollet Samvelian. 2020. Complex predicates. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 297–351. Berlin: Language Science Press. DOI:??

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar approach to argument structure* (Cognitive Theory of Language and Culture). Chicago: The University of Chicago Press.

Gunji, Takao. 1999. On lexicalist treatments of Japanese causatives. In Robert D. Levine & Georgia Green (eds.), *Studies in contemporary phrase structure grammar*. Cambridge & New York: Cambridge University Press.

Hendriks, Petra. 1995a. *Comparatives and Categorial Grammar*. University of Groningen dissertation.

Hendriks, Petra. 1995b. Ellipsis and multimodal categorial type logic. In Glyn V. Morrill & Richard T. Oehrle (eds.), *Formal Grammar: Proceedings of the Conference of the European Summer School in Logic, Language and Information, Barcelona, 1995*, 107–122.

Hoeksema, Jack. 1984. *Categorial morphology*. Rijksuniversiteit Groningen dissertation.

Hoeksema, Jack & Richard D. Janda. 1988. Implications of process-morphology for Categorial Grammar. In Richard T. Oehrle, Emmon Bach & Deirdre Wheeler (eds.), *Categorial grammars and natural language structures*, 199–247. Dordrecht: D. Reidel Publishing Company.

Hofmeister, Philip & Ivan A Sag. 2010. Cognitive constraints and island effects. *Language* 86(2). 366–415. DOI:10.1353/lan.0.0223

Howard, William A. 1969. The formulae-as-types notion of construction. Published in Jonathan P. Seldin and J. Roger Hindley (eds.) 1980. *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*, pp. 479–490. Academic Press.

Humberstone, Lloyd. 2005. Geach's Categorial Grammar. *Linguistics and Philosophy* 28. 281–317.

Jacobson, Pauline. 1998. Antecedent contained deletion and pied-piping: evidence for a variable-free semantics. In Devon Strolovitch & Aaron Lawson (eds.), *Proceedings from Semantics and Linguistic Theory VIII*, 74–91. Ithaca, New York: Cornell University.

Jacobson, Pauline. 1999. Towards a variable-free semantics. *Linguistics and Philosophy* 22(2). 117–184.

Jacobson, Pauline. 2000. Paycheck pronouns, Bach-Peters sentences, and variable-free semantics. *Natural Language Semantics* 8(2). 77–155.

Jacobson, Pauline. 2007. Direct compositionality and variable-free semantics: The case of "Principle B" effects. In Chris Barker & Pauline Jacobson (eds.), *Direct compositionality*, 191–236. Oxford: Oxford University Press.

Jacobson, Pauline. 2008. Direct compositionality and variable-free semantics: the case of Antecedent Contained Deletion. In Kyle Johnson (ed.), *Topics in ellipsis*, 30–68. Cambridge, England: Cambridge University Press.

Jacobson, Pauline. 2016. The short answer: Implications for direct compositionality (and vice versa). *Language* 92(2). 331–375. DOI:10.1353/lan.2016.0038

Jacobson, Pauline. 2019. Ellipsis in Categorial Grammar. In Jeroen van Craenenbroeck & Tanja Temmerman (eds.), *The oxford handbook of ellipsis*, 122–141. Oxford: Oxford University Press.

Jäger, Gerhard. 2005. *Anaphora and Type-Logical Grammar*. Berlin: Springer.

Johnson, Kyle. 2009. Gapping is not (VP-) ellipsis. *Linguistic Inquiry* 40(2). 289–328.

Johnson, Kyle. 2019. Gapping and stripping. In Jeroen van Craenenbroeck & Tanja Temmerman (eds.), *The Oxford handbook of ellipsis*, 562–604. Oxford: Oxford University Press.

Kaplan, Ronald M. & Joan Bresnan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan (ed.), *The mental representation of grammatical relations* (MIT Press Series on Cognitive Theory and Mental Representation), 173–281. Cambridge, MA/London: MIT Press. Reprint in: **DKMZ95a-ed**.

Kathol, Andreas. 2000. *Linear syntax*. New York, NY: Oxford University Press.

Kennedy, Christopher. 2003. Ellipsis and syntactic representation. In Kerstin Schwabe & Susanne Winkler (eds.), *The interfaces: deriving and interpreting omitted structures*, 29–53. Amsterdam: John Benjamins.

Kluender, Robert. 1998. On the distinction between strong and weak islands: a processing perspective. In Peter Culicover & Louise McNally (eds.), *The limits of syntax*, vol. 29 (Syntax and Semantics), 241–279. San Diego: Academic Press.

Koenig, Jean-Pierre & Frank Richter. 2020. Semantics. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 729–762. Berlin: Language Science Press. DOI:??

Kraak, Esther. 1998. A deductive account of French object clitics. In Erhard W. Hinrichs, Andreas Kathol & Tsuneko Nakazawa (eds.), *Complex predicates in nonderivational syntax*, vol. 30 (Syntax and Semantics), 271–312. San Diego: Academic Press.

Kubota, Yusuke. 2010. *(In)flexibility of constituency in Japanese in Multi-Modal Categorial Grammar with Structured Phonology*. Ohio State University dissertation.

Kubota, Yusuke. 2014. The logic of complex predicates. English. *Natural Language and Linguistic Theory* 32(4). 1145–1204. DOI:10.1007/s11049-014-9246-8

Kubota, Yusuke. 2015. Nonconstituent coordination in Japanese as constituent coordination: an analysis in Hybrid Type-Logical Categorial Grammar. *Linguistic Inquiry* 46(1). 1–42.

Kubota, Yusuke & Robert Levine. 2014. Scope anomaly of Gapping. In *Proceedings of NELS 44*, 247–260. GLSA.

Kubota, Yusuke & Robert Levine. 2015. Against ellipsis: arguments for the direct licensing of 'non-canonical' coordinations. *Linguistics and Philosophy* 38(6). 521–576.

Kubota, Yusuke & Robert Levine. 2016a. Gapping as hypothetical reasoning. *Natural Language and Linguistic Theory* 34(1). 107–156.

Kubota, Yusuke & Robert Levine. 2016b. The syntax-semantics interface of 'respective' predication: a unified analysis in Hybrid Type-Logical Categorial Grammar. *Natural Language and Linguistic Theory* 34(3). 911–973.

Kubota, Yusuke & Robert Levine. 2017. Pseudogapping as pseudo-VP ellipsis. *Linguistic Inquiry* 48(2). 213–257.

Kubota, Yusuke & Robert Levine. 2020. *Type-logical syntax*. Cambridge, Mass.: MIT Press.

Kubota, Yusuke, Koji Mineshima, Robert Levine & Daisuke Bekki. 2019. Underspecification and interpretive parallelism in Dependent Type Semantics. In *Proceedings of the IWCS 2019 Workshop on Computing Semantics with Types, Frames and Related Structures*, 1–9. Gothenburg, Sweden. https://www.aclweb.org/anthology/W19-1001, accessed 2020-2-24.

Kuhlmann, Marco, Alexander Koller & Giorgio Satta. 2015. Lexicalization and generative power in CCG. *Computational Linguistics* 41(2). 187–219. https://www.aclweb.org/anthology/J15-2002. DOI:10.1162/COLI_a_00219

Lasnik, Howard. 1986. On the necessity of binding conditions. In Robert Freidin (ed.), *Principles and parameters in comparative grammar*, 7–28. Cambridge, Mass.: MIT Press.

Levine, Robert. 2011. Linearization and its discontents. In Stefan Müller (ed.), *The proceedings of the 18th International Conference on Head-Driven Phrase Structure Grammar*, 126–146. Stanford: CSLI Publications. http://cslipublications.stanford.edu/HPSG/2011/.

Levine, Robert D. 2017. *Syntactic analysis: an HPSG-based approach*. Cambridge, England: Cambridge University Press.

Levine, Robert D. & Thomas E. Hukari. 2006. *The unity of unbounded dependency constructions* (CSLI Lecture Notes 166). Stanford, CA: CSLI Publications.

Levine, Robert D., Thomas E. Hukari & Mike Calcagno. 2001. Parasitic gaps in English: Some overlooked cases and their theoretical consequences. In Peter W. Culicover & Paul M. Postal (eds.), *Parasitic gaps* (Current Studies in Linguistics 35), 181–222. London, England/Cambridge, MA: MIT Press.

Levinson, Stephen C. 1987. Pragmatics and the grammar of anaphora. *Journal of Linguistics* 23. 379–434.

Levinson, Stephen C. 1991. Pragmatic reduction of the Binding Conditions revisited. *Journal of Linguistics* 27. 107–161.

Levy, Roger. 2001. Feature indeterminacy and the coordination of unlikes in a totally well-typed HPSG. http://www.mit.edu/~rplevy/papers/feature-indet.pdf, accessed 2018-7-29. Unpublished manuscript.

Levy, Roger & Carl Pollard. 2002. Coordination and neutralization in HPSG. In Frank van Eynde, Lars Hellan & Dorothee Beermann (eds.), *Proceedings of the HPSG 2001 conference*, 221–234. Stanford, CA: CSLI Publications.

Lewis, Mike & Mark Steedman. 2013. Combined distributional and logical semantics. *Transactions of the Association for Computational Linguistics* 1. 179–192.

Manning, Christopher D., Ivan A. Sag & Masayo Iida. 1999. The lexical integrity of Japanese causatives. In Robert D. Levine & Georgia M. Green (eds.), *Studies in contemporary Phrase Structure Grammar*, 39–79. Cambridge, UK: Cambridge University Press.

Martin-Löf, Per. 1984. *Intuitionistic type theory*. Bibliopolis.

Martin, Scott. 2013. *The dynamics of sense and implicature*. Ohio State University dissertation.

McCawley, James D. 1993. Gapping with shared operators. In David A. Peterson (ed.), *Berkeley Linguistics Society*, 245–253. Berkeley, California.

McCloskey, James. 1979. *Transformational syntax and model-theoretic semantics*. Dordrecht: Reidel.

Merchant, Jason. 2019. Ellipsis: a survey of analytical approaches. In Jeroen van Craenenbroeck & Tanja Temmerman (eds.), *The Oxford handbook of ellipsis*, 19–45. Oxford: Oxford University Press.

Mihaliček, Vedrana & Carl Pollard. 2012. Distinguishing phenogrammar from tectogrammar simplifies the analysis of interrogatives. In Philippe de Groote & Mark-Jan Nederhof (eds.), *Formal Grammar 2010/2011*, 130–145. Heidelberg: Springer.

Miller, Philip. 2014. A corpus study of pseudogapping and its theoretical consequences. In Christopher Piñón (ed.), *Empirical issues in syntax and semantics*, vol. 10, 73–90.

Mineshima, Koji, Ribeka Tanaka, Pascual Martínez-Gómez, Yusuke Miyao & Daisuke Bekki. 2016. Building compositional semantics and higher-order inference system for a wide-coverage Japanese CCG parser. In *Proceedings of EMNLP 2016*, 2236–2242. Austin, Texas: Association for Computational Linguistics.

Moortgat, Michael. 1984. A Fregean restriction on metarules. In *Proceedings of NELS 14*, 306–325. GLSA.

Moortgat, Michael. 1997. Categorial Type Logics. In Johan van Benthem & Alice ter Meulen (eds.), *Handbook of logic and language*, 93–177. Amsterdam: Elsevier.

Moortgat, Michael. 2011. Categorial type logics. In Johan F. A. K. van Benthem & G. B. Alice ter Meulen (eds.), *Handbook of logic and language*, 2nd edn., 95–179. Amsterdam: Elsevier.

Moortgat, Michael & Richard T. Oehrle. 1994. Adjacency, dependence, and order. In Paul Dekker & Martin Stokhof (eds.), *Proceedings of the Ninth Amsterdam Colloquium*, 447–466. Universiteit van Amsterdam: Instituut voor Taal, Logica, en Informatica.

Moot, Richard. 2015. A type-logical treebank for French. *Journal of Language Modelling* 3(1). 229–264.

Moot, Richard. 2018. Combining logical and distributional methods in type-logical grammars. *Journal of Language Modelling* 6(2). 287–317.

Moot, Richard & Christian Retoré. 2012. *The logic of Categorial Grammars: A deductive account of natural language syntax and semantics.* Heidelberg: Springer Verlag.

Morrill, Glyn. 2010. *Categorial Grammar: Logical syntax, semantics, and processing*. Oxford: Oxford University Press.

Morrill, Glyn. 2017. Grammar logicised: relativisation. *Linguistics and Philosophy* 40(2). 119–163.

Morrill, Glyn V. 1994. *Type Logical Grammars: Categorial logic of signs.* Dordrecht: Kluwer Academic Publishers.

Morrill, Glyn & Josep-Maria Merenciano. 1996. Generalizing discontinuity. *Traitement Automatique des Langues* 27(2). 119–143.

Morrill, Glyn & Teresa Solias. 1993. Tuples, discontinuity, and gapping in Categorial Grammar. In *Proceedings of the Sixth Conference on European Chapter of the Association for Computational Linguistics*, 287–296. Morristown, NJ, USA: Association for Computational Linguistics.

Morrill, Glyn & Oriol Valentín. 2017. A reply to Kubota and Levine on gapping. *Natural Language and Linguistic Theory* 35(1). 257–270.

Morrill, Glyn, Oriol Valentín & Mario Fadda. 2011. The displacement calculus. *Journal of Logic, Language and Information* 20(1). 1–48.

Müller, Stefan. 2018. The end of lexicalism as we know it? *Language* 94(1). e54–e66. DOI:10.1353/lan.2018.0014

Müller, Stefan. 2019. *Grammatical theory: From Transformational Grammar to constraint-based approaches*. 3rd edn. (Textbooks in Language Sciences 1). Berlin: Language Science Press. DOI:10.5281/zenodo.3364215

Müller, Stefan. 2020a. Constituent order. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 259–293. Berlin: Language Science Press. DOI:??

Müller, Stefan. 2020b. HPSG and Construction Grammar. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 1087–1125. Berlin: Language Science Press. DOI:??

Müller, Stefan & António Branco. 2020. Anaphoric Binding. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 639–682. Berlin: Language Science Press. DOI:??

Müller, Stefan & Stephen Mark Wechsler. 2014. Lexical approaches to argument structure. *Theoretical Linguistics* 40(1–2). 1–76. DOI:10.1515/tl-2014-0001

Muskens, Reinhard. 2003. Language, lambdas, and logic. In Geert-Jan Kruijff & Richard Oehrle (eds.), *Resource sensitivity in binding and anaphora*, 23–54. Dordrecht: Kluwer.

Newmeyer, Frederick J. 2016. Nonsyntactic explanations of island constraints. *Annual Review of Linguistics* 2. 187–210.

Nykiel, Joanna & Jong-Bok Kim. 2020. Ellipsis. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 617–638. Berlin: Language Science Press. DOI:??

Oehrle, R. T. 1988. Multi-dimensional compositional functions as a basis for grammatical analysis. In R. T. Oehrle, E. Bach & D. Wheeler (eds.), *Categorial Grammars and natural language structures*, 349–389. Dordrecht: Reidel.

Oehrle, Richard T. 1971. On Gapping. MS., MIT.

Oehrle, Richard T. 1987. Boolean properties in the analysis of gapping. In Geoffrey J. Huck & Almerindo E. Ojeda (eds.), *Syntax and semantics: discontinuous constituency*, vol. 20, 203–240. Academic Press.

Oehrle, Richard T. 1994. Term-labeled categorial type systems. *Linguistics and Philosophy* 17(6). 633–678.

Oehrle, Richard T. 2011. Multi-Modal Type-Logical Grammar. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar: A guide to current models*, 225–267. Oxford, UK/Cambridge, MA: Blackwell Publishers Ltd.

Park, Sang-Hee. 2019. *Gapping: a constraint-based syntax-semantics interface*. State University of New York at Buffalo dissertation.

Park, Sang-Hee, Jean-Pierre Koenig & Rui P. Chaves. 2019. A semantic underspecification-based analysis of scope ambiguities in Gapping. In M. Teresa Espinal, Elena Castroviejo, Manuel Leonetti & Louise McNally (eds.), *Proceedings of Sinn und Bedeutung 23*, vol. 2, 237–251.

Partee, Barbara Hall. 1970. Negation, conjunction, and quantifiers: syntax vs. semantics. *Foundations of Language* 6. 153–165.

Partee, Barbara & Mats Rooth. 1983. Generalized conjunction and type ambiguity. In Rainer Bäuerle, Christoph Schwarze & Arnim von Stechow (eds.), *Meaning, use, and interpretation of language*, 361–383. Berlin: Walter de Gruyter.

Perlmutter, David M. & John Robert Ross. 1970. Relative clauses with split antecedents. *Linguistic Inquiry*. 350.

Pickering, Martin & Guy Barry. 1991. Sentence processing without empty categories. *Language and Cognitive Processes* 6. 229–264.

Pickering, Martin & Guy Barry. 1993. Dependency Categorial Grammar and coordination. *Linguistics* 31(5). 855–902.

Pogodalla, Sylvain & Florent Pompigne. 2012. Controlling extraction in Abstract Categorial Grammars. In Philippe de Groote & Mark-Jan Nederhof (eds.), *Formal Grammar 2010/2011*, 162–177. Heidelberg: Springer.

Pollard, Carl. 2013. Linear Categorial Grammar. Ms., Lecture Notes at ESSLLI 2013.

Pollard, Carl. 2014. What numerical determiners mean: a non-ambiguity analysis. Talk presented at the Workshop on Semantics of Cardinals, Ohio State University, March 6, 2014.

Pollard, Carl J. 1984. *Generalized Phrase Structure Grammars, Head Grammars, and natural language*. Stanford University dissertation.

Pollard, Carl J. & Ivan A. Sag. 1987. *Information-based syntax and semantics* (CSLI Lecture Notes 13). Stanford, CA: CSLI Publications.

Pollard, Carl J. & Ivan A. Sag. 1992. Anaphors in English and the scope of Binding Theory. *Linguistic Inquiry* 23(2). 261–303.

Pollard, Carl J. & Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar* (Studies in Contemporary Linguistics). Chicago: The University of Chicago Press.

Postal, Paul M. 1998. *Three investigations of extraction*. Cambridge, Mass.: MIT Press.

Przepiórkowski, Adam. 2020. Case. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 157–177. Berlin: Language Science Press. DOI:??

Puthawala, Daniel. 2018. Stripping isn't so mysterious, or anomalous scope, either. In Annie Foret, Greg Kobele & Sylvain Pogodalla (eds.), *Formal Grammar 2018*, 102–120.

Raffelsiefen, Renate. 1992. A nonconfigurational approach to morphology. In Mark Aronoff (ed.), *Morphology now*, 133–162. State University of New York Press.

Ranta, Aarne. 1994. *Type-theoretical grammar*. Oxford: Oxford University Press.

Reape, Mike. 1994. Domain union and word order variation in German. In John Nerbonne, Klaus Netter & Carl J. Pollard (eds.), *German in Head-Driven Phrase Structure Grammar* (CSLI Lecture Notes 46), 151–198. Stanford, CA: CSLI Publications.

Richter, Frank & Manfred Sailer. 2004. Basic concepts of lexical resource semantics. In Arnold Beckmann & Norbert Preining (eds.), *ESSLLI 2003 – Course material I* (Collegium Logicum 5), 87–143. Wien: Kurt Gödel Society.

Sag, Ivan A. 1997. English relative clause constructions. *Journal of Linguistics* 33(2). 431–484. DOI:10.1017/S002222679700652X

Sag, Ivan A. 2010. English filler-gap constructions. *Language* 86(3). 486–545.

Sag, Ivan A., Hans C. Boas & Paul Kay. 2012. Introducing Sign-Based Construction Grammar. In Hans C. Boas & Ivan A. Sag (eds.), *Sign-Based Construction Grammar* (CSLI Lecture Notes 193), 1–29. Stanford, CA: CSLI Publications.

Sag, Ivan A., Gerald Gazdar, Thomas Wasow & Steven Weisler. 1985. Coordination and how to distinguish categories. *Natural Language and Linguistic Theory* 3(2). 117–171.

Sag, Ivan A. & Thomas Wasow. 2011. Performance-compatible competence grammar. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar: A guide to current models*, 359–377. Oxford, UK/Cambridge, MA: Blackwell Publishers Ltd.

Shiraïshi, Aoi, Anne Abeillé, Barbara Hemforth & Philip Miller. 2019. Verbal mismatch in right-node raising. *Glossa: A journal of general linguistics* 4(1). 114.1–26. DOI:http://doi.org/10.5334/gjgl.843

Siegel, Muffy A. 1987. Compositionality, case, and the scope of auxiliaries. *Linguistics and Philosophy* 10(1). 53–75.

Steedman, Mark. 1985. Dependency and coördination in the grammar of Dutch and English. *Language* 61(3). 523–568. DOI:10.2307/414385

Steedman, Mark. 1987. Combinatory grammars and parasitic gaps. *Natural Language and Linguistic Theory* 5(3). 403–439. DOI:10.1007/BF00134555

Steedman, Mark J. 1990. Gapping as constituent coordination. *Linguistics and Philosophy* 13(2). 207–265. DOI:10.1007/BF00630734

Steedman, Mark. 1997. *Surface structure and interpretation* (Linguistic Inquiry Monographs 30). Cambridge, MA: MIT Press.

Steedman, Mark. 2000. *The syntactic process* (Language, Speech, and Communication). Cambridge, MA: MIT Press.

Steedman, Mark. 2012. *Taking scope*. Cambridge, MA: MIT Press.

Steedman, Mark & Jason Baldridge. 2011. Combinatory Categorial Grammar. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar: A guide to current models*, 181–224. Oxford, UK/Cambridge, MA: Blackwell Publishers Ltd.

Stump, Gregory T. 2001. *Inflectional morphology: A theory of paradigm structure* (Cambridge Studies in Linguistics 93). Cambridge University Press.

Szabolcsi, Anna. 1992. Combinatory grammar and projection from the lexicon. In Ivan A. Sag & Anna Szabolcsi (eds.), *Lexical matters* (CSLI Lecture Notes 24), 241–268. Stanford: CSLI Publications.

van Benthem, Johan. 1988. The semantics of variety in Categorial Grammar. In Wojciech Buszkowski, Witold Marciszewski & Johan van Benthem (eds.), *Categorial grammar*, 37–55. Amsterdam: John Benjamins Publishing Co.

Wasow, Tom. 2020. Processing. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 797–812. Berlin: Language Science Press. DOI:??

Wechsler, Stephen, Jean-Pierre Koenig & Anthony Davis. 2020. Argument structure and linking. In Stefan Müller, Anne Abeillé, Robert D. Borsley & Jean-Pierre Koenig (eds.), *Head-Driven Phrase Structure Grammar: The handbook*, 215–258. Berlin: Language Science Press. DOI:??

White, Michael & Jason Baldridge. 2003. Adapting chart realization to CCG. In *Proceedings of the 9th European workshop on natural language generation*, 119–126.

Whitman, Neal. 2009. Right-node wrapping: Multimodal Categorial Grammar and the 'friends in low places' coordination. In Erhard Hinrichs & John Nerbonne (eds.), *Theory and evidence in semantics*, 235–256. Stanford, CA: CSLI Publications.

Wilder, Chris. 2019. Conjunction reduction and right-node raising. In Jeroen van Craenenbroeck & Tanja Temmerman (eds.), *The Oxford handbook of ellipsis*, 681–718. Oxford: Oxford University Press.

Yatabe, Shûichi. 2001. The syntax and semantics of Left-Node Raising in Japanese. In Daniel P. Flickinger & Andreas Kathol (eds.), *Proceedings of the HPSG-2000 Conference, University of California, Berkeley*, 325–344. CSLI Publications. http://csli-publications.stanford.edu/HPSG/1/, accessed 2018-2-25.

Yatabe, Shûichi & Wai Lok Tam. 2019. In defense of an HPSG-based theory of non-constituent coordination: a reply to Kubota and Levine. *Linguistics and Philosophy* 0. 00–00.

Yoshida, Masaya, Tim Hunter & Michael Frazier. 2015. Parasitic gaps licensed by elided syntactic structure. *Natural Language and Linguistic Theory* 33(4). 1439–1471.

Zaenen, Annie. 1983. On syntactic binding. *Linguistic Inquiry* 14(3). 469–504.