12 Comparison with Other Variants of Categorial Grammar

In this chapter, we provide a brief comparison of our framework with other variants of CG. The presentation in chapter 2 has deliberately emphasized the similarities between Hybrid TLCG and mainstream generative syntax, but being a variant of CG, Hybrid TLCG also owes a lot to the so-called lexicalist tradition in syntactic research to which most variants of CG (especially CCG) are normally assimilated. In what follows, we try to clarify what we take to be the key similarities and differences between our approach and other major variants of CG in the current literature.

12.1 Other Variants of TLCG

Variants of TLCG can all be seen as extensions of the Lambek calculus to cope with its shortcomings in dealing with phenomena such as medial extraction and quantification. They are thus closely related to each other (for example, theorems in the Lambek calculus are also theorems in all these systems). Among these different variants, Morrill's Displacement Calculus and Barker and Shan's continuation-based calculus NL_{λ} are most closely related to Hybrid TLCG.

12.1.1 Multi-Modal TLCG

There is a line of work in the TLCG literature, most actively investigated back in the 1990s by researchers such as Michael Moortgat, Dick Oehrle, and Rafaella Bernardi (Moortgat and Oehrle 1994; Moortgat 1997; Bernardi 2002), which sees the underlying combinatoric component of natural language syntax as a kind of substructural logic (Restall 2000). As discussed in chapter 11, the multi-modal prosodic component of (an extended version of) Hybrid TLCG is inspired by some of the technical innovations in this line of work, but the conceptual and technical underpinnings are rather different in the two approaches. One crucial difference between Hybrid TLCG and this earlier version of TLCG is that, instead of recognizing a separate level of prosodic representation, this version of TLCG deals with various (somewhat heterogeneous) phenomena ranging from morpho-syntactic properties of verb clusters in Dutch cross-serial dependencies (Moortgat and Oehrle 1994) to technical difficulties with the Lambek calculus in view of medial extraction (Moortgat 1997) via the abstract notion of "structural control," building on the technique of mixing different kinds of logic within a single deductive system originally developed in the literature of substructural logic (Restall 2000).

To see this point, it is instructive to take a look at the analysis of medial extraction in Bernardi (2002):

(614)

$$
\dfrac{
\begin{array}{c}(\mathrm{N}\backslash\mathrm{N})/(\mathrm{S}/_a\mathrm{NP}) \vdash \\ (\mathrm{N}\backslash\mathrm{N})/(\mathrm{S}/_a\mathrm{NP})\end{array}
\quad
\dfrac{
\dfrac{
\dfrac{
\dfrac{
\mathrm{NP} \vdash \mathrm{NP}
\quad
\dfrac{
\dfrac{
\begin{array}{c}(\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \vdash \\ (\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP}\end{array}
\quad [\,\mathrm{NP} \vdash \mathrm{NP}\,]^1
}{
(\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ_a \mathrm{NP} \vdash \mathrm{NP}\backslash\mathrm{S}
}{/_a\mathrm{E}}
\quad
\begin{array}{c}(\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S}) \vdash \\ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S})\end{array}
}{
((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ_a \mathrm{NP}) \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S}) \vdash \mathrm{NP}\backslash\mathrm{S}
}{\backslash\mathrm{E}}
}{
\mathrm{NP} \circ (((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ_a \mathrm{NP}) \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S})) \vdash \mathrm{S}
}{\backslash\mathrm{E}}
}{
\mathrm{NP} \circ (((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S})) \circ_a \mathrm{NP}) \vdash \mathrm{S}
}{\mathrm{Diss}}
}{
(\mathrm{NP} \circ ((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S}))) \circ_a \mathrm{NP} \vdash \mathrm{S}
}{\mathrm{MixAssoc}}
}{
\mathrm{NP} \circ ((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S})) \vdash \mathrm{S}/_a\mathrm{NP}
}{/_a\mathrm{I}^1}
}{
(\mathrm{N}\backslash\mathrm{N})/(\mathrm{S}/_a\mathrm{NP}) \circ (\mathrm{NP} \circ ((\mathrm{NP}\backslash\mathrm{S})/_a\mathrm{NP} \circ (\mathrm{NP}\backslash\mathrm{S})\backslash(\mathrm{NP}\backslash\mathrm{S}))) \vdash \mathrm{N}\backslash\mathrm{N}
}{/\mathrm{E}}
$$

Here, the derivation is written in the sequent-style natural deduction, in which a sequent is a construct of the form $\Gamma \vdash A$ where the consequent A is a syntactic type and the antecedent Γ is a structured object consisting of possibly multiple syntactic types that are combined with one another via binary connectives (of several sorts). On the leaves of the trees, we find axioms of the form $A \vdash A$, and the sequent on the final line of the proof tree is obtained by applying the rules of logic (such as the Elimination and Introduction rules similar to the ones we have posited in previous chapters) and "structural rules," which manipulate the forms of the antecedents of the sequents that are derived.

Intuitively, what is going on in the above derivation is that an object NP that is hypothesized on the right of a transitive verb is "moved" to the right edge via the structural rules of Diss and MixAssoc in the structured antecedent on the third line from the bottom. This licenses the $/_a\mathrm{I}$ rule, withdrawing the hypothesis to create the $\mathrm{S}/_a\mathrm{NP}$ category that is suitable as an argument to the relative pronoun. After lexical substitution, the derivation in (614) embodies, for example, a formal proof of the fact that the string *which Sara wrote __ there* is a well-formed relative clause in English containing a medial gap.

There are a couple of important points of difference between this version of TLCG and other variants of categorial grammar that should already be clear from this rather cursory exposition. First, the analysis of extraction via a series of structure rewriting operations in the abstract substructural component of the type logic is very different from the prosodic λ-abstraction analysis of extraction we have adopted in our own

This is a portion of the eBook at doi:10.7551/mitpress/11866.001.0001

approach, which is due originally to Muskens (2003) and is incorporated in many contemporary variants of TLCG and related approaches (such as Hybrid TLCG and Linear Categorial Grammar—see below). Second, the exact ontological status of this "multimodal" substructural component is somewhat unclear. As noted above, the structure-changing operations have been put to use not just for extraction in English but also for dealing with complex morpho-syntactic properties of verb clustering in Dutch cross-serial dependencies (Moortgat and Oehrle 1994) (in Hybrid TLCG, the latter type of phenomenon would be dealt with in the multi-modal prosodic component, as discussed in chapter 11). All of these are done inside a complex logic of syntactic types, where these syntactic types are taken as primitives that enter into binary composition operations of various sorts having different combinatorial possibilities. While there is nothing technically wrong with this approach, it considerably obscures the conceptual underpinnings of the system as a theory of natural language syntax. But the different ontological setup in different variants of TLCG may reflect different research goals and research practices. When the emphasis is on linguistic application (as in our approach), a clear separation of ontologically distinct components would be an important point of consideration, but when the emphasis is on studying the metalogical properties of the formal calculus, building directly on the rich literature of substructural logic and formalizing the type logic for natural language syntax literally *as* a substructural logic is certainly an attractive research strategy.

12.1.2 Displacement Calculus and NL_λ

Displacement Calculus (Morrill et al. 2011; Morrill 2010) and NL_λ (Barker and Shan 2015) are perhaps the versions of categorial grammar that are closest to Hybrid TLCG in terms of both the general architecture, analytic toolkits available in the respective approaches, and the linguistic analyses of specific phenomena formulated in each. There are, however, several nontrivial differences. Roughly speaking, Hybrid TLCG's vertical slash $\upharpoonright$ plays more or less the same role as the discontinuity connectives $\uparrow$ and $\downarrow$ in Displacement Calculus and the "continuation" slashes $/\!\!/$ and $\backslash\!\!\backslash$ and in NL_λ. Many empirical analyses of linguistic phenomena formulated in one of these variants of TLCG translate to the other two more or less straightforwardly (for example, the analyses of Gapping and symmetrical predicates in Kubota and Levine [2016a] and Kubota and Levine [2016b], whose key ideas are briefly sketched above, build on Morrill's and Barker's analyses of the respective phenomena).

One major difference between the Displacement Calculus and Hybrid TLCG on the one hand and NL_λ on the other is that the latter takes **NL**, namely, the Non-associative Lambek calculus, as the underlying calculus for the directional slashes $/$ and $\backslash$. Barker and Shan (2015) briefly comment on this property of their system, alluding to the possibility of controlling flexibility of constituency via the notion of "structural control" in Multi-Modal Type-Logical Categorial Grammar (see above). This certainly is a viable view, but no explicit extension of NL_λ along these lines currently exists. The other major difference pertains to the broader architectural design: NL_λ is technically a version of Multi-Modal TLCG, and the behaviors of the two continuation slashes $/\!\!/$ and $\backslash\!\!\backslash$ are controlled by structural rules of the sort described above.

Morrill's approach differs from ours in certain important ways in the treatment of specific linguistic phenomena. The most substantial disagreement pertains to the treatment of island constraints. Morrill consistently holds the view that major island constraints should be treated within the narrow syntax (Morrill 1994, 2010, 2017), which contrasts sharply with our own view explicated in chapter 10. See also a brief discussion about determiner gapping in chapter 3 for another point of disagreement.

12.2 Combinatory Categorial Grammar

CCG (Steedman 1996, 2000) is essentially an extension of the AB Grammar that proposes to simulate movement with a limited set of additional rules. Though there are several different variants and extensions, CCG typically consists of rules of *type raising* and *function composition*, together with function application.[1] Thus, the following represents a reasonable rule set for a simple CCG fragment (to facilitate comparison, we have written these rules in the tripartite sign format and have adopted the Lambek slash notation):[2]

(615) a. Forward Function Application

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

b. Backward Function Application

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

(616) a. Forward Function Composition

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

b. Backward Function Composition

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

(617) a. Forward Type Raising

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

b. Backward Type Raising

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

1. In some variants of CCG (such as Steedman [2012]), type raising is not recognized as a syntactic rule, but the effects of type raising are all lexicalized. This has certain advantages in terms of computational considerations.

2. But the fragment here is an impoverished version of CCG for an expository purpose only. A linguistically more adequate version typically involves rules of "crossed" function composition (for extraction from nonperipheral positions; cf. Steedman 1996) and the so-called substitution rules for the treatment of parasitic gaps (Steedman 1987).

As noted by Steedman (1985), with type raising and function composition, we can analyze a string of words such as *John loves* as a constituent of type S/NP:

(618)
$$\dfrac{\dfrac{\mathsf{john};\,\mathbf{j};\,\mathrm{NP}}{\mathsf{john};\,\lambda f.f(\mathbf{j});\,\mathrm{S}/(\mathrm{NP}\backslash\mathrm{S})}\mathrm{TR} \quad \mathsf{loves};\,\mathbf{love};\,(\mathrm{NP}\backslash\mathrm{S})/\mathrm{NP}}{\mathsf{john}\circ\mathsf{loves};\,\lambda x.\mathbf{love}(\mathbf{j})(x);\,\mathrm{S}/\mathrm{NP}}\mathrm{FC}$$

Thus, in CCG, essentially the same analysis of RNR is possible as in Hybrid TLCG. The same is true for DCC, as first noted by Dowty (1988). For details, we refer the reader to Dowty (1988) and Steedman (2000). Note, however, that in CCG the derived constituents in RNR and DCC are obtained by the rules in (615)–(617) rather than via hypothetical reasoning. In fact, CCG is notable for its strong thesis of "surface compositionality," according to which the semantic interpretation of a sentence is directly obtained on the basis of the surface constituent structure assigned on the basis of a limited set of "combinatory" rules like those in (615)–(617). In particular, there is no direct analogue of vertical slash in CCG.

As far as the coverage of the basic syntactic patterns of NCC is concerned, CCG and Hybrid TLCG are more or less equivalent. But things are not so simple when we consider a wider range of data. First of all, CCG does not have a fully general analysis of the interactions between coordination and quantifier scope. An analysis of wide-scope readings of quantifiers in coordination (including RNR and DCC) in examples like (619) is straightforward in CCG (as demonstrated in (620)).

(619) (Either) the department owns, or the library has an interlibrary license to, every single book in the SLAP series.

(620)
$$\dfrac{\begin{array}{l}\vdots\\ \mathsf{the}\circ\mathsf{department}\circ\mathsf{owns}\circ\mathsf{or}\circ\mathsf{the}\circ\mathsf{library}\circ\mathsf{has}\circ\mathsf{a}\circ\mathsf{license}\circ\mathsf{to};\\ \lambda x.\mathbf{own}(x)(\mathbf{d})\vee\mathbf{has\text{-}license}(x)(\mathbf{l});\mathrm{S}/\mathrm{NP}\end{array} \quad \begin{array}{l}\mathsf{every}\circ\mathsf{book};\\ \mathbf{V}_{\mathbf{book}};(\mathrm{S}/\mathrm{NP})\backslash\mathrm{S}\end{array}}{\begin{array}{l}\mathsf{the}\circ\mathsf{department}\circ\mathsf{owns}\circ\mathsf{or}\circ\mathsf{the}\circ\mathsf{library}\circ\mathsf{has}\circ\mathsf{a}\circ\mathsf{license}\circ\mathsf{to}\circ\mathsf{every}\circ\mathsf{book};\\ \mathbf{V}_{\mathbf{book}}(\lambda x.\mathbf{own}(x)(\mathbf{d})\vee\mathbf{has\text{-}license}(x)(\mathbf{l}));\mathrm{S}\end{array}}\mathrm{FA}$$

However, it is unclear how the distributive readings of quantifiers, exemplified by data such as (621) (in its $\vee > \forall$ reading), are obtained in CCG.

(621) (The department used to have a heavy requirement for candidacy. For example, I no longer remember which it was, but) back in those days, every student had to pass at least two language exams or had to write QPs in both syntax and phonology.

In CCG, coordinated VPs are unambiguously of type NP\S and the subject quantifier has category S/(NP\S). Thus, only the weaker $\forall > \vee$ reading is predicted for this sentence. The discussion of the interaction between coordination and scope in Steedman (2012, section 2.3 and chapter 10) suggests that Steedman takes the distributive readings of quantifiers in coordination to be unavailable.[3] However, we have argued in chapter 4 that distributive readings for quantifiers are in fact available for both constituent coordination and NCC by setting up the appropriate pragmatic context and pronouncing the sentence in the right prosody.

It is of course conceivable to extend the rule set of CCG so that the $\vee > \forall$ reading is obtained for (621). In particular, with the rule of *argument raising* (Hendriks 1993), one can derive the type $(S/(NP\backslash S))\backslash S$ from the lower type $NP\backslash S$ for the VP. But argument raising is usually not recognized as an admissible rule in standard CCG.[4]

This reveals one important difference between TLCG and CCG: rules such as type raising, function composition, and argument raising are all theorems that can be derived from the more basic rules of inference in the more general, logic-based setup of TLCG, including ours. Thus, unless some additional constraints are imposed, the availability of both scopal relations are predicted for sentences such as (621) in TLCG, assuming that the generalized conjunction entry for *or* can be instantiated to any arbitrarily complex category. Such a prediction does not automatically follow in CCG, since only a subset of theorems in the Lambek calculus are recognized as legal syntactic rules in CCG. Whether one takes this "limited" flexibility of CCG to be desirable seems to largely depend on one's take on the status of empirical data such as (621).

Other areas of potential difficulty for CCG include the analysis of symmetrical predicates and related expressions (chapter 5) and Gapping (chapter 3). In both of these cases, we have made crucial use of the vertical slash and proposed movement-like analyses of the scope-taking properties of the relevant expressions. Since CCG does not have a direct analog of the vertical slash, the analyses we have proposed for these phenomena do not seem to translate straightforwardly to CCG. It is of course conceivable that alternative analyses of these empirical phenomena in CCG are possible, but so far as we are aware, no concrete analysis that has comparable empirical coverage to ours has been proposed in CCG to date.[5]

3. See also Gärtner (2012) for some discussion about quantification and scope in CCG. The treatment of distributive readings for sentences like (621) is similarly unclear in Gärtner's (2012) approach.

4. Also, the category metavariables *A* and *B* in the type-raising schema in (617) are usually taken to be restricted in a certain way in CCG, ruling out the possibility of obtaining the same effect via type raising.

5. As far as anomalous scope in Gapping is concerned, the recent proposal in HPSG by Park et al. (2019) comes closest to a starting point for such a counter-analysis in CCG. But Park et al.'s analysis relies heavily on underspecification, and not all formal details are worked out explicitly. Thus, whether the key ideas of their analysis can be implemented in CCG is still considerably unclear.

12.3 Linear Categorial Grammar

The problem that quantification (or scope-taking more generally) poses for CCG essentially stems from the fact that the forward and backward slashes that encode directionality are not the optimal tool for characterizing the mismatch between the surface position of the quantifier and its semantic scope. In fact, as we have already noted in connection to our discussion of the vertical slash, most variants of CG, including the *Lambek calculus* (Lambek 1958), have essentially the same problem. The family of CGs that we call LCG here, which include the term-labeled calculus of Oehrle (1994), *Abstract Categorial Grammar* (de Groote 2001), *Lambda Grammar* (Muskens 2003), and *Linear Categorial Grammar* (Mihalicek and Pollard 2012), have been proposed partly in response to this empirical shortcoming of "directional" variants of CG, including the Lambek calculus.

LCG is essentially Hybrid TLCG with only $\upharpoonright$ as the syntactic connective and only $\upharpoonright$I and $\upharpoonright$E as the syntactic rules. All the lexical entries involving / and \ are rewritten using $\upharpoonright$ and specifying word order directly in the prosodic form of the lexical entries via λ-terms, as in the following lexical entry for the transitive verb *saw*:[6]

(622) $\lambda\varphi_2\lambda\varphi_1.\varphi_2 \circ \mathsf{saw} \circ \varphi_1; \mathbf{saw}; S\upharpoonright NP\upharpoonright NP$

To facilitate comparison, we present in (623) the derivation for inverse scope in LCG.

(623)

$$\frac{\lambda\varphi_1\lambda\varphi_2.\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1;\ \mathbf{talked\text{-}to};\ S\upharpoonright NP\upharpoonright NP \qquad \begin{bmatrix}\varphi_1;\\ x_1;\\ NP\end{bmatrix}^1}{\lambda\varphi_2.\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1;\ \mathbf{talked\text{-}to}(x_1);\ S\upharpoonright NP}\upharpoonright\mathrm{E}$$

$$\frac{\begin{bmatrix}\varphi_2;\\ x_2;\\ NP\end{bmatrix}^2 \qquad \lambda\varphi_2.\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1;\ \mathbf{talked\text{-}to}(x_1);\ S\upharpoonright NP}{\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1;\ \mathbf{talked\text{-}to}(x_1)(x_2);\ S}\upharpoonright\mathrm{E}$$

$$\frac{\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1;\ \mathbf{talked\text{-}to}(x_1)(x_2);\ S \qquad \lambda\varphi.\varphi\circ\mathsf{yesterday};\ \mathbf{yest};\ S\upharpoonright S}{\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2));\ S}\upharpoonright\mathrm{E}$$

$$\frac{\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2));\ S}{\lambda\varphi_2.\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2));\ S\upharpoonright NP}\upharpoonright\mathrm{I}^2$$

$$\frac{\lambda\sigma.\sigma(\mathsf{someone});\ \mathbf{\exists_{person}};\ S\upharpoonright(S\upharpoonright NP) \qquad \lambda\varphi_2.\varphi_2\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2));\ S\upharpoonright NP}{\mathsf{someone}\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \mathbf{\exists_{person}}(\lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2)));\ S}\upharpoonright\mathrm{E}$$

$$\frac{\mathsf{someone}\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \mathbf{\exists_{person}}(\lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2)));\ S}{\lambda\varphi_1.\mathsf{someone}\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \lambda x_1.\mathbf{\exists_{person}}(\lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2)));\ S\upharpoonright NP}\upharpoonright\mathrm{I}^1$$

$$\frac{\lambda\sigma.\sigma(\mathsf{everyone});\ \mathbf{\forall_{person}};\ S\upharpoonright(S\upharpoonright NP) \qquad \lambda\varphi_1.\mathsf{someone}\circ\mathsf{talked}\circ\mathsf{to}\circ\varphi_1\circ\mathsf{yesterday};\ \lambda x_1.\mathbf{\exists_{person}}(\lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2)));\ S\upharpoonright NP}{\mathsf{someone}\circ\mathsf{talked}\circ\mathsf{to}\circ\mathsf{everyone}\circ\mathsf{yesterday};\ \mathbf{\forall_{person}}(\lambda x_1.\mathbf{\exists_{person}}(\lambda x_2.\mathbf{yest}(\mathbf{talked\text{-}to}(x_1)(x_2))));\ S}\upharpoonright\mathrm{E}$$

6. Here, for notational consistency with other parts of the book, we adopt our notation of slashes. Since $\upharpoonright$, the only syntactic connective in LCG, is really just linear implication, $A\upharpoonright B$ in our notation is more standardly written as $B \multimap A$ in the LCG literature.

Notwithstanding the elegant analysis of scope-taking phenomena available with the use of λ-binding in the prosodic component, however, LCG has its own, quite serious empirical shortcoming: unlike CCG and Lambek calculus–based variants of CG, in which there is a very simple analysis of coordination extending straightforwardly to NCC, in LCG, coordination becomes an almost intractable problem. Since this is an important empirical issue that has not received sufficient attention in the literature (but see Muskens [2001] for a cursory remark noting but ultimately dismissing the problem), we discuss it in some detail here. A more thorough discussion addressing various partial fixes one might make in LCG, such as adding information about grammatical case (none of which generalizes properly), is found in Moot (2014) (see also Kubota 2010, section 3.2.1).

The problem can be succinctly illustrated by RNR examples such as the following:

(624) Terry hates, and Leslie likes, Robin.

Suppose we attempt to derive this example in LCG. The derivation in (625) goes through straightforwardly.

(625)
$$\dfrac{\dfrac{\dfrac{\lambda\varphi_1\lambda\varphi_2.\varphi_2 \circ \mathsf{hates} \circ \varphi_1; \mathbf{hate}; S{\upharpoonright}NP{\upharpoonright}NP \quad [\varphi_3; x; NP]^3}{\lambda\varphi_2.\varphi_2 \circ \mathsf{hates} \circ \varphi_3; \mathbf{hate}(x); S{\upharpoonright}NP}{\upharpoonright}E \quad \mathsf{terry}; \mathbf{t}; NP}{\mathsf{terry} \circ \mathsf{hates} \circ \varphi_3; \mathbf{hate}(x)(\mathbf{t}); S}{\upharpoonright}E}{\lambda\varphi_3.\mathsf{terry} \circ \mathsf{hates} \circ \varphi_3; \lambda x.\mathbf{hate}(x)(\mathbf{t}); S{\upharpoonright}NP}{\upharpoonright}I^3$$

A parallel derivation yields the category $S{\upharpoonright}NP$ for *Leslie likes*. But a complication arises at this point, since, unlike in CCG (or in the Lambek calculus), in LCG, the conjuncts do not correspond to strings; they are functional terms of type $\mathbf{st}{\to}\mathbf{st}$. Thus, they cannot be directly concatenated to form the coordinated string. One might think that assigning the following type of lexical entry for the conjunction word *and* (of type $(\mathbf{st}{\to}\mathbf{st}){\to}(\mathbf{st}{\to}\mathbf{st}){\to}(\mathbf{st}{\to}\mathbf{st})$) would work:

(626) $\lambda\sigma_1\lambda\sigma_2\lambda\varphi.\sigma_2(\boldsymbol{\varepsilon}) \circ \mathsf{and} \circ \sigma_1(\varphi); \sqcap; (S{\upharpoonright}NP){\upharpoonright}(S{\upharpoonright}NP){\upharpoonright}(S{\upharpoonright}NP)$

Applying this functor to the two conjuncts indeed yields the following sign for the whole coordinate structure,

(627) $\lambda\varphi.\mathsf{terry} \circ \mathsf{hates} \circ \boldsymbol{\varepsilon} \circ \mathsf{and} \circ \mathsf{leslie} \circ \mathsf{likes} \circ \varphi; \lambda x.\mathbf{hate}(x)(\mathbf{t}) \wedge \mathbf{like}(x)(\mathbf{l}); S{\upharpoonright}NP$

to which the object NP *Robin* can be given as an argument to complete the derivation.

This analysis, however, overgenerates in a quite serious way. To see this, note that the following can also be derived as a well-formed sign with category $S{\upharpoonright}NP$ in LCG via hypothetical reasoning:

(628) $\lambda\varphi_1.\varphi_1 \circ \mathsf{likes} \circ \mathsf{robin}; \mathbf{like}(\mathbf{r})(x); S{\upharpoonright}NP$

Conjoining this with the same first conjunct *Terry hates* above yields the following expression:

(629) $\lambda\varphi.\mathsf{terry} \circ \mathsf{hates} \circ \boldsymbol{\varepsilon} \circ \mathsf{and} \circ \varphi \circ \mathsf{likes} \circ \mathsf{robin}; \lambda x.\mathbf{hate}(x)(\mathbf{t}) \wedge \mathbf{like}(\mathbf{r})(x); \mathsf{S}\upharpoonright\mathsf{NP}$

By giving *Leslie* as an argument to this functor, we have an analysis for the sentence *Terry hates and Leslie likes Robin*, to which the meaning "Terry hates Leslie and Leslie likes Robin" is assigned, which is obviously wrong. The problem, of course, is that what in a directional CG would be two distinct types S/NP and NP\S are conflated as both instances of $\mathrm{S}\upharpoonright\mathrm{NP}$ in LCG, and therefore there is no apparent way to avoid conjoining *Terry hates* with *likes Robin*.[7]

A problem of essentially the same nature arises with even more basic cases of constituent coordination such as the following, with an even more embarrassing effect:

(630) John caught and ate the fish.

Note that, in LCG, the following two signs are interderivable (hypothesizing object and subject NPs and withdrawing them in the opposite order derives (631b) from (631a)):

(631) a. $\lambda\varphi_1\lambda\varphi_2.\varphi_2 \circ \mathsf{ate} \circ \varphi_1; \mathbf{ate}; \mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP}$
b. $\lambda\varphi_2\lambda\varphi_1.\varphi_2 \circ \mathsf{ate} \circ \varphi_1; \lambda x\lambda y.\mathbf{ate}(y)(x); \mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP}$

Then, a conjunction entry in (632) for coordination of transitive verbs of type $\mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP}$ can license (630), at the expense of predicting that the sentence is ambiguous in all of the four readings in (633).

(632) $\lambda\sigma_1\lambda\sigma_2\lambda\varphi_1\lambda\varphi_2.\varphi_2 \circ \sigma_2(\boldsymbol{\varepsilon})(\boldsymbol{\varepsilon}) \circ \mathsf{and} \circ \sigma_1(\boldsymbol{\varepsilon})(\boldsymbol{\varepsilon}) \circ \varphi_1;$
$\sqcap; (\mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP})\upharpoonright(\mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP})\upharpoonright(\mathrm{S}\upharpoonright\mathrm{NP}\upharpoonright\mathrm{NP})$

(633) a. $\mathbf{caught}(\mathbf{j})(\mathbf{the\text{-}fish}) \wedge \mathbf{ate}(\mathbf{j})(\mathbf{the\text{-}fish})$
b. $\mathbf{caught}(\mathbf{j})(\mathbf{the\text{-}fish}) \wedge \mathbf{ate}(\mathbf{the\text{-}fish})(\mathbf{j})$
c. $\mathbf{caught}(\mathbf{the\text{-}fish})(\mathbf{j}) \wedge \mathbf{ate}(\mathbf{j})(\mathbf{the\text{-}fish})$
d. $\mathbf{caught}(\mathbf{the\text{-}fish})(\mathbf{j}) \wedge \mathbf{ate}(\mathbf{the\text{-}fish})(\mathbf{j})$

This issue of overgeneration in coordination has received some attention in the recent LCG literature (see, e.g., Kanazawa 2015; Pollard and Worth 2015; Worth 2016). In particular, Pollard and Worth (2015) and Worth (2016) propose to overcome this problem by encoding order sensitivity for linguistic expressions that have functional

7. Even though Hybrid TLCG incorporates LCG as its subsystem, the type of overgeneration in LCG in coordination discussed here does not arise in Hybrid TLCG. The problem with LCG in a nutshell is that since the system allows for only the vertical slash $\upharpoonright$, which does not encode linear order, one is forced to specify the conjunction category for functional expressions (such as verbs) in terms of $\upharpoonright$, losing control over linear order. In Hybrid TLCG (as in CCG), the conjunction category is specified in terms of directional slashes (except for the case of Gapping, which receives a different treatment due to the fact that the "gap" is medial), and conjunction entries such as (626) and (632) discussed here are not posited.

prosodies via operators called “phenominators” that make reference to fine-grained subtyping in the prosodic type system encoding directionality information. This enables simulating most of the empirical analyses of coordination developed in directional variants of CG within LCG. However, based on the results reported in Pollard and Worth (2015) and Worth (2016), it is as yet not totally clear whether a completely general characterization of the prosodic subtyping system involving phenominators can be worked out. Until such a characterization is provided, we take it that coordination poses a major empirical challenge for LCG.

Type-Logical Syntax

By: Yusuke Kubota, Robert D. Levine

Citation:

Type-Logical Syntax
By: Yusuke Kubota, Robert D. Levine
DOI: 10.7551/mitpress/11866.001.0001
ISBN (electronic): 9780262360807
Publisher: The MIT Press
Published: 2020

The open access edition of this book was made possible by generous funding and support from Arcadia – a charitable fund of Lisbet Rausing and Peter Baldwin

The MIT Press

The open access edition of this book was made possible by generous funding from Arcadia—a charitable fund of Lisbet Rausing and Peter Baldwin.

ARCADIA
A charitable fund of Lisbet Rausing and Peter Baldwin

This book was set in Syntax and Times Roman by the authors.

Library of Congress Cataloging-in-Publication Data

Names: Kubota, Yusuke, author. | Levine, Robert, 1947- author.
Title: Type-logical syntax / Yusuke Kubota, Robert D. Levine.
Description: Cambridge, Massachusetts : The MIT Press, [2016] | Includes bibliographical references and index.
Identifiers: LCCN 2020000483 | ISBN 9780262539746 (paperback)
Subjects: LCSH: Categorial grammar. | Grammar, Comparative and general–Syntax.
Classification: LCC P161 .K83 2016 | DDC 415–dc23
LC record available at https://lccn.loc.gov/2020000483

ISBN: 978-0-262-53974-6