Combinatory Grammar and Projection from the Lexicon

ANNA SZABOLCSI

Objective

This paper addresses the relation between the lexicon and syntax, and sug-
gests that projection from the lexicon should be the constitutive principle
of grammar. The argument will consist of a general part in Section 1 and
a specific part in Section 2.

In Section 1, I will ask to what extent the assumptions concerning the
nature of lexical items determine the form of syntactic representations in
various theories. As an illustration, I will briefly examine the treatments of
unbounded dependencies in Government and Binding theory, Head-driven
Phrase Structure Grammar, and Combinatory Categorial Grammar. I will
argue that the concept of lexical items as functions underlies all three of
them. However, only CCG recognizes this within its theoretical frames, and
therefore only CCG can use it as an explanatory principle.

In Section 2, I present a case study on anaphora as a specific exam-
ple of the connection between the lexicon and syntax within this theory.
I will show that a simple lexical distinction between duplicators and free
variables interacts with the independently motivated machinery of CCG to
derive what is commonly stipulated in Binding Theory in syntax. Section
2.1 will be concerned with reflexives, and Section 2.2 with pronouns. After
showing that the combinatory approach inescapably leads to a theory à la
Reinhart, in Section 2.3 I will extend the proposal to VP-ellipsis, and elimi-

The research reported in this paper was supported by a fellowship from IREX at the
1987 Linguistic Institute at Stanford University, and one from the British Council in
the Centre for Cognitive Science in Edinburgh in 1988. I wish to thank the members of
those communities, as well as of MIT, for insightful objections and suggestions. I am fur-
thermore indebted to Jeff Goldberg, László Kálmán, Katalin E. Kiss, Edith Moravcsik,
Stanley Peters, and Mark Steedman for recent discussions.

Lexical Matters. Ivan A. Sag and Anna Szabolcsi, eds.
Copyright © 1992, Stanford University.
inate an apparent problem caused by strict readings with non-referential antecedents.

1 The Projection Principle: One Filter or the Whole Story?

In current theories of grammar the relation of the lexicon to syntax is regulated by some form of the Projection Principle:

(1) Syntactic representations are projected from the lexicon in that they observe the pertinent properties of lexical items (where “pertinent” and “observe” are defined theory-internally)

This principle is used to impose a necessary, but not sufficient, condition on well-formed syntactic representations; they must not contradict the properties of the lexical items they contain (subcategorization properties in the first place). That granted, however, they may be subject to any kind of further conditions (pertaining to empty elements and binding, for instance). Given that nothing constrains the nature of such conditions with reference to the nature of lexical items, the extent to which the lexicon eventually determines the form of syntactic representations need not be overwhelming.

This conclusion, while perfectly compatible with the technical status of the Projection Principle, is in conflict with the constraining power many linguists attribute to it. Intuitively, what we want may be stated as make as much of the lexicon as possible, or, more strongly, take lexical items to be axioms and derive sentences as theorems. This intuition would require that projection from the lexicon be the constitutive principle of grammar.

One obvious part of this program is to develop a sophisticated empirical theory of the lexicon that explains, for instance, why certain argument structures are attested and others are not. All “interesting” issues about grammar should belong here, and I will not be concerned with it in this paper. The other part of the program, on which I will focus, is to show how “uninteresting” our beloved syntax actually is. This involves making specific assumptions about the general nature of lexical items that, so to say, “predict syntax.” I believe that the crucial reason why syntactic representations tend to be underdetermined by the lexicon is that most theories are extremely neutral as to what kind of objects lexical items are. Given this neutrality, the choice of syntactic tools is extremely free, so the “laws” of syntax turn out to be rather independent of “lexical substance”.

1.1 Government-Binding Theory and Head-Driven Phrase Structure Grammar

To illustrate this independence, let us consider two examples, one from GB as in Kayne 1983\(^1\) and another from HPSG as in Pollard 1984, 1988

\(^1\)The insights of Kayne 1983 are also featured in versions of GB, based on Chomsky 1986a. See Koster 1987 as well.
and Pollard and Sag 1987. GB is a theory in which there is no prevalent specific mathematical commitment as to what kind of entities linguistic objects, lexical items among them, are. HPSG is a theory that makes a very general but mathematically precise commitment: it takes linguistic objects, including lexical items, to be feature structures in a strict technical sense. In this section I will first briefly review their treatments of “extraction” and “parasitic gaps”. I will then point out that in neither theory does the nature of the generalizations concerning the well-formedness of syntactic structures follow naturally from the view of lexical items that the theory adopts.

Consider the following examples:

(2) (a gangster) who Mary said relatives of ____ thought Kim eliminated ____.
(3) (a gangster) who Mary said the police thought Kim eliminated ____.
(4) *(a gangster) who Mary said relatives of ____ thought Kim died.

For (2) to be well-formed in Kayne’s theory, it must be possible to assign a structure to it that conforms to the X-bar theory of phrase structure and the theta theory of subcategorization in the first place. I assume that the reader can imagine how this would work. Secondly, given that gaps are filled by an empty category, (2) must satisfy the Empty Category Principle (ECP) that Kayne defines as follows:

(5) **Empty Category Principle:**
An empty category $e$ must have an antecedent $a$ such that
a. $a$ governs $e$, or
   a c-commands $e$ and there exists a lexical category $X$ such that $X$ governs $e$, and $a$ is contained in some $g$ (overnment)-projection of $X$, and
b. the union of the $g$-projection sets of all the empty categories $e_1, \ldots, e_n$ bound by the same antecedent $a$ must constitute a subtree.

(6) $Y$ is a $g$-projection of $X$ iff
a. $Y$ is an X-bar projection of $X$ or of a $g$-projection of $X$, or
b. $X$ is a structural governor, and $Y$ immediately dominates $W$ and $Z$, where $Z$ is a governed (Longobardi’s amendment) maximal projection of a $g$-projection of $X$, and $W$ and $Z$ are in a canonical government configuration. (English being a VO language, the canonical government configuration is $WZ$.)

(7) The $g$-projection set of $e$ contains
a. every $p$ that is a $g$-projection of $X$, the governor of $e$, and
b. $e$ itself.

Now consider (2). Note that Kayne takes $S$ and $S'$ to be projections
of V. In diagram (8) below only branching nodes are given. Nodes are labelled with the numbers of the relevant clauses of the definition.

(8) \[ \begin{align*}
\text{COMP} & \quad \text{V''(6a)} \\
\text{who} & \quad \text{NP} \\
\text{Mary} & \quad \text{V} \\
\text{said} & \quad \text{NP(6b)} \\
\text{N} & \quad \text{PP(6a)} \\
\text{relatives} & \quad \text{P} \\
\text{of} & \quad e_2 \\
\text{Kim} & \quad \text{V} \\
\text{eliminated} & \quad e_1 
\end{align*} \]

Let us first see whether \( e_1 \) is legitimate in view of the ECP (5). \( e_1 \)'s antecedent \( \text{who} \) c-commands \( e_1 \) but it is too far up in the tree to govern it. On the other hand, there exists a lexical category, \( \text{eliminate} \), that governs \( e_1 \). So the question is whether \( \text{who} \) is contained in a \( g \)-projection of \( \text{eliminate} \). We check this against definition (6), proceeding from bottom to top. \( \text{V'} \) and \( \text{V''} \) are \( X \)-bar projections, and hence \( g \)-projections, of \( \text{eliminate} \). \( \text{V''} \) is in a canonical government configuration with its sister \( \text{thought} \), hence the next \( \text{V'} \) is a \( ^{\text{c}} \)-projection of \( \text{eliminate} \). With similar steps we get as far as the topmost \( \text{V'''} \) (= S-bar). It indeed contains the c-commanding binder \( \text{who} \). The relation between \( \text{who} \) and the "real gap" \( e_1 \) is therefore legitimate, as is also attested by the grammaticality of (3).

Now take the relation between \( \text{who} \) and \( e_2 \). Of governs \( e_2 \). \( \text{Of} \) is an \( X \)-bar projection, hence a \( g \)-projection, of \( \text{of} \). \( \text{PP} \) is in a canonical government configuration with \( \text{relatives} \); hence \( \text{NP} \) is a \( g \)-projection of \( \text{of} \). \( \text{V''} \) is not an \( X \)-bar projection of this \( \text{NP} \), nor is \( \text{NP} \) in a canonical government configuration with its sister. Hence \( \text{V''} \) is not a \( g \)-projection of \( \text{of} \). Therefore, if \( e_2 \) were the only gap, it could not be connected to \( \text{who} \), as is also attested by the ungrammaticality of (4). However, the \( g \)-projection of \( \text{of} \) that reaches \( \text{NP} \) forms a subtree with that of \( \text{eliminate} \). Hence, in view of the second clause of ECP, \( e_2 \) is saved; it can be "parasitic" on \( e_1 \).

Consider now the treatment of the same example in Pollard 1984 and 1988. For (2) to be well-formed in HPSG, it must be obtainable by unifying the feature structures corresponding to its lexical items. \( \text{Eliminate} \)
COMBINATORY GRAMMAR AND PROJECTION FROM THE LEXICON / 245

has, for instance, alongside with features specifying its part of speech and morphological properties, a feature SUBCAT, whose value is (NP, NP), and an empty category e has a feature SLASH with value (NP). It is an important property of the grammar that it contains, in addition to the general principles of unification, principles specifically pertaining to such particular features. The informal summary of the principles that govern the behavior of SUBCAT and SLASH is modeled after that of Pollard (1988):

(9) Remove symbols from the front of the SUBCAT list one by one, observing the Subcategorization Principle (SCP):
    Match the symbols with a complement.

(10) Binding Inheritance Principle (BIP):
    a. For each binding feature B, the value of B on the mother is obtained by starting with the value on the head daughter and appending (host dependency) or merging (parasitic dependency) into it the values on the other daughters, in the order more-oblique to less-oblique.
    b. Binding features are not passed up if they are specifically discharged in the construction.

(11) Head Feature Principle (HFP):
    For other features, the value on mother is obtained from the head daughter.

(12) SLASH and REL are binding features.

Caution: only the relevant portion of feature structures is given in (13).

We start from e₁ again. Unification of eliminated with the trace e₁ yields the feature structure V[SUBCAT (NP), SLASH (NP)]. Unification of this with Kim yields V with [SUBCAT ( )] by the SCP and [SLASH (NP)] by the BIP. [SLASH (NP)] will propagate up by the BIP. At the node dominating relatives of e₂ thought Kim eliminated e₁ we have merger since another SLASH value turns up, coming from e₂. Note, incidentally, that nothing in (10) would prevent this latter SLASH from propagating even in the absence of e₁, in which case (4) needs to be excluded by an extra constraint on the BIP (Pollard 1984, p. 175). At the top we find who with another binding feature, REL, of value NP, and Mary said relatives of thought Kim eliminated with [SUBCAT ( ), SLASH (NP)]. Finally, a topicalization rule allows these to unify, emptying SLASH and retaining NP in the REL-stack.

Let us return to Kayne’s proposal now. The first thing to notice is that the presence of who needs to be guaranteed by the ECP, for the principles pertaining to subcategorization are already satisfied by the empty categories e₁ and e₂. Second, the g-projection mechanism seems entirely stipulative in the sense that there is no independent reason in the grammar to make us expect that “pathbuilding” between the governor of the gap and its antecedent will play a role in the grammaticality of sentences. Viewed
(13) 

```
(13) V[REL (NP)]
    /   
NP[REL (NP)] ... V[SLASH (NP)]
     |      
   who    

V[SLASH (NP)]
    /   
NP[SLASH (NP)] V[SUBCAT (NP, SLASH (NP))]
     |      
 N[SUBCAT (PP)] PP[SLASH (NP)]
       |      
  relatives thought

V[SUBCAT (NP, S)]
    /   
V[SLASH (NP)]
    /   
NP[SLASH (NP)]
     |      
   e2    

V[SUBCAT (NP)]
    /   
NP[SUBCAT (NP)]
     |      
 NP[SLASH (NP)]
     |      
   Kim    

V[SUBCAT (NP, SLASH (NP))]
    /   
NP[SLASH (NP)]
     |      
   e1    
```

from another angle, Kayne’s proposal is not arbitrary, however. Suppose that heads are not merely said to “subcategorize for” specifiers and complements but are admitted as functions of these. Then, using the notation $f : a \rightarrow b$, viz., function: domain $\rightarrow$ co-domain, (6a,b) can be said to be about the following constellation (assume association to the right): $X$ is a function from $e$ (and possibly other arguments) to $X^{\text{max}}$, and $W$ is a function from $X^{\text{max}}$ (and possibly other arguments) to $W^{\text{max}}$:

(14) a. $X : \ldots \rightarrow e \rightarrow \ldots \rightarrow X^{\text{max}}$
    b. $W : \ldots \rightarrow X^{\text{max}} \rightarrow \ldots \rightarrow W^{\text{max}}$

To say now that a path is definable between $e$ and $W^{\text{max}}$ is to notice that $X$ and $W$ as given above are composible functions. The path-forming operation can be schematized as a general version of function composition:

(15) $\lambda e[W \ldots (X \ldots (e)\ldots)\ldots]$

Furthermore, the subtree condition suggests an extension of this operation which, in addition to composing $W$ and $X$, identifies one argument of $W$ with an argument of $X$. Let us say that $W$ and $X$ are connected in this case:

(16) $\lambda e[W \ldots (e)\ldots (X \ldots (e)\ldots)\ldots]$

Now we can say that Kayne’s definition of $g$-projection is nothing other than an algorithm that specifies how the items intervening between the empty categori(es) and the antecedent can be assembled into one big function by composition and connection (plus application to pick up other arguments on the way). This is an interesting result because we may now feel we understand the rationale behind this rather complex mechanism. First, we understand what a $g$-projection path is: the result of application, composition and connection. Second, we understand why $g$-projection, as opposed to any arbitrary graph-theoretic notion, may play a distinguished role in grammar: because heads are function-like and it is in the nature of functions to apply, compose and connect. Third, we even understand why $g$-projection is specifically relevant for structures with gaps: because composition and connection allow a function to extend itself and thus combine with a distant argument. The presence of who on top will now be directly related to the subcategorization of the governor(s) at the root(s) of the path. Note, though, that while these considerations provide glorious justification for Kayne’s ECP, they remain external to his theory since GB does not explicitly identify lexical items with subcategorization frames as functions.

The situation is even more striking in the case of HPSG. Pollard (1988) demonstrates, for example, with exemplary clarity, that the Subcategorization Principle could be paraphrased as “Combine signs as if heads were functions applying to complements as arguments”, and the use of SLASH
plus Binding Inheritance can be paraphrased as "Propagate information as if you were composing (connecting) functions". It remains a fact, though, that HPSG's lexical items are not even uncommitted entities but are entities committed specifically not to be functions. SUBCAT and SLASH are just features on a par with, say, suppletion features. Therefore every bit of their functional behavior needs to be imposed by brute force (by means of the SCP and the BIP). That is to say, the recognition of their functional behavior is not merely external but rather orthogonal to the theory they are part of.

1.2 Combinatory Categorial Grammar

We have seen that the "function metaphor" appears insightful in connection with important properties of syntactic representations. It may thus be interesting to consider a grammar that elevates this metaphor to the status of a theoretical claim and thereby lets the kind of operations found in syntax be a natural consequence of the kind of entities it takes lexical items to be.

Categorial grammar is known for adopting the view that lexical items with subcategorization frames are to be conceived of as functions/functors. One respect in which versions of categorial grammar differ quite significantly has to do with what abilities of functions they make use of in syntax. For extensive discussion, see the contributions in Oehrle et al. 1988. The version that I will explore below began to take shape in Ades and Steedman 1982 and Szabolcsi 1983, and has come to bear the name "combinatory categorial grammar" (CCG), for reasons that will become clear shortly. Its main strategy can be summarized as follows.

(17) a. No appeal is made to a phonetically empty placeholder or some equivalent instance of hidden application to satisfy the subcategorization requirements of the head when it does not combine immediately with its intended argument.

b. The observations concerning functional behavior that are built into GB/HPSG in the form of constraints are turned into operations that construct the well-formed representations, and only those, directly.

For an expression to be well-formed in this theory, it must be possible to assemble its lexical items in a strictly monotonic fashion by rules that are applicable whenever they meet categories they are defined for, and whose output is not subjected to further constraints. Rules (18)–(21) constitute a representative portion of the grammar. Their input ought to be more restricted, but I will adhere to these overly general formulations to let the reader see what can be achieved without fixes.

The rules in (18)–(21) are given in the same format as derivations will be. For example, (18a) can be spelled out as follows: Let a be an expression
of category \( Y/X \), the category of functors that look for an \( X \) on their right and return a \( Y \), and let \( b \) be an expression of category \( X \). They can be concatenated in this order to yield \( ab \) of category \( Y \). In (18b) we have \( a \) of \( Y\backslash X \), a leftward-looking functor from \( X \) to \( Y \), hence \( b \) of \( X \) must precede \( a \). (Notice that it is the orientation of the slash that indicates the directionality of the functor; the domain category is always to the right of the co-domain category.) In both cases the result of concatenation is interpreted by applying \( a' \), the function interpreting \( a \), to \( b' \), the argument interpreting \( b \). Underlines bear an index that has no status in the theory but merely indicates, for the sake of transparency, what semantic operation is performed, and which version of concatenation is used.\(^2\)

The output of each semantic operation is spelled out in lambda terms for convenience. Notice that we use exactly application, composition and connection—the operations noted in the preceding section. (The fourth rule, lifting, is a kind of auxiliary device that is necessary but is of no particular theoretical interest in the present context.)

(18) Concatenation interpreted as application: \( a'(b') \)

\[
\begin{array}{ll}
\text{i.} & a & \underline{b} \\
Y/X & X & A_i \\
\text{ii.} & b & a \\
X & Y\backslash X & A_{ii}
\end{array}
\]

(19) Concatenation interpreted as composition: \( \lambda x[a'(b'(x)))] \)

\[
\begin{array}{ll}
\text{i.} & a & \underline{b} \\
Z/Y & Y/X & B_i \\
\text{ii.} & b & a \\
Y/X & Z\backslash Y & B_{ii}
\end{array}
\]

(20) Concatenation interpreted as connection: \(^3\) \( \lambda x[a'(x)(b'(x))] \)

\[
\begin{array}{ll}
\text{i.} & a & \underline{b} \\
(Z/Y)/X & Y/X & S_i \\
\text{ii.} & b & a \\
Y/X & (Z\backslash Y)/X & S_{ii}
\end{array}
\]

(21) Category lifting interpreted as: \( \lambda f[f(a')] \)

\[
\begin{array}{ll}
\text{i.} & a \\
Z & T_i \\
\text{ii.} & a \\
Z & T_{ii}
\end{array}
\]

Y/(Y\backslash Z) \\
Y\backslash (Y/Z)

The derivation of (2) will proceed as follows:

\(^2\)For the choice of index letters, see (23) below.

\(^3\)The connection operation, introduced in Szabolcsi 1983, is renamed as substitution in Steedman 1987, 1988.
The (S\NP)/NP category of eliminated specifies that it is a functor with two NP arguments, of which the first (object) is expected from the right and the second (subject) is expected from the left. In this sentence there is nothing in the position where eliminated expects its object to be, and it is a remarkable feature of the grammar that it does not insert a phonetically empty element, either. So eliminated combines directly with Kim. This cannot be by application, though, since the subject NP is not the first argument of the transitive verb (and there is nothing in the grammar to change argument order). The subject and the transitive verb can combine via composition, which is possible iff the subject also bears an appropriate functor category, cf. the input conditions of (19i). (Whether a noun phrase in the nominative comes with the functor category S/(S\NP) from the lexicon, or (21i) lifts NP to S/(S\NP), is immaterial for our present concerns.) The composed functor Kim eliminated is exactly like eliminated in that it expects an NP from its right. The same holds for thought Kim eliminated that we compose in the next step.

Now comes relatives of. Its internal structure can be ignored; the important point is that, once more, it has nothing in the position of the prepositional object and is therefore a functor. Relatives of and thought Kim eliminated match the input conditions in (20i), so they can be connected into a big “forked” functor that is looking for one NP (who) which, when found, will satisfy both prongs. (This grammar, just like that of Pollard (1988), would require an extra restriction to rule (4) out. As things stand now, S\NP can be lifted to S\(S/(S\NP))\), whence thought that Kim died can compose with relatives of of (S/(S\NP))/NP.) The trivial composition steps that take care of said and Mary are omitted to save space. Finally, accusative who must bring its S/(S/NP) category from the lexicon. The reason is partly semantic (who does not denote an entity, hence cannot be just NP), and partly syntactic (lifting as in (21) preserves the order of combination and hence cannot be responsible for the “left-extracting” property of who). Who combines with (Mary said) relatives of thought Kim eliminated by application.

Now, do the lack of traces and the use of complex operations, as in
(17ab), represent two arbitrarily juxtaposed grammatical strategies, or are they logically coherent? It seems they are, since as Steedman (1987, 1988) observes, there exists a system of logic that has essentially these features as its defining properties. This is combinatory logic, initiated by Schönfinkel (1924), developed, among others, by Curry and Feys (1958), and expanded in a most enjoyable fashion by Smullyan (1985). Combinatory logic is a system with the same potential expressive power as the lambda calculus. But while the lambda calculus uses abstraction and bound variables, combinatory logic appeals solely to functional operations (combinators), to achieve the same results. That is, it differs from the lambda calculus in precisely the same way as the grammar just sketched differs from GB and HPSG.

For a fast-and-easy introduction to combinators, note the following. In (15)–(16) and (19)–(21) we were dealing with “composed functions”, “connected functions”, and “lifted objects”. The combinators “compositor” ($B$), “connector” ($S$) and “lifter” ($T$) are the operations themselves that we get by abstracting from those functions/objects. In (23) I define them both in usual lambda terms and in standard combinatory notation. The latter should be read as follows: $B$ is that operation which, when applied to $f$, $g$, and $x$ in this order, returns $f(gx)$, etc. Left-associativity is assumed, i.e., $(ab)c = abc$.

$$f, g, x$$

(23) a. Compositor: $B = \lambda f \lambda g \lambda x[f(gx)]$

$$Bf gx = f(gx)$$

b. Connector: $S = \lambda h \lambda g \lambda x[h(x)(gx)]$

$$Sh gx = hx(gx)$$

c. Lifter: $T = \lambda x \lambda g [g(x)]$

$$Tx g = gx$$

Application is not a combinator; it is the interpretation of the concatenation of any two combinatory terms. However, it is possible to define a combinator $A$ that does exactly this, $\lambda g \lambda x [g(x)]$.

The fact that combinatory logic has the same potential expressive power as the lambda calculus entails that variable binding can be dispensed with. To illustrate what sameness of expressive power means in more complex cases, consider two examples in (24), which will also be useful below:

$$f, g, x$$

(24) a. $ST$ expresses the same as $\lambda f \lambda x [f(x)(x)]$ because

$$STfx = Tx(fx) = fxx$$

b. $BB(ST)$ expresses the same as $\lambda f \lambda g \lambda x [f(gx)(gx)]$ because

$$BB(ST)fgx = B(STf)gx = STf(gx) = T(gx)(f(gx)) = f(gx)(gx)$$

It may be interesting to point out that the operations we found useful in devising a grammar of English are not merely definable in combinatory terms but actually all correspond to rather fundamental combinators. $B$, $S$ and $T$ represent those independent operation types which, were they supplemented with an identifier or a cancellator and allowed to apply recursively, would actually yield the power of the full lambda calculus.
Since that power is neither necessary nor desirable in our syntax, I will continue to restrict my attention to combinators with specific linguistic motivation. Another important assumption that I make is that combinators are typed (cf. Hindley et al. 1972 and Morrill and Carpenter 1987. For instance, in (23a,b,c), \( f \) is to have a type of the form \( P \rightarrow R \), \( g \) of \( Q \rightarrow P \), \( x \) of \( Q \), and \( h \) of \( Q \rightarrow (P \rightarrow R) \). Typing will be assumed but not notated throughout the discussion.

In sum, the pursuit of the “function metaphor” appears to have led us to the “combinator metaphor”. Again, the question is whether we should leave it as a metaphor, or we should assign it the status of a theoretical claim. The latter possibility would mean that we try to devise a restrictive theory of grammar by letting the spirit and techniques of combinatory logic delimit our analytic options.

In what follows I will make one move in this direction. I will present a case study of anaphors and pronouns, and argue that, in view of both its positive and negative results, combinatory logic can serve as a guideline for their study.

Caveat: In the rest of the paper I will continue to use lambda terms alongside with, and sometimes even instead of, combinatory terms. I wish to point out that this is only because combinators are unfamiliar to many readers. Lambda terms will merely serve the purpose of exposition; they have the same status as a paraphrase in ordinary English would.

2 Anaphora—Lexical Semantics and Syntax

The suggestion that natural language syntax implements combinators, rather than explicit variable binding, was motivated above by a specific kind of example in which the variable would be phonetically empty. In those cases it is easy to argue that we only need to account for the placement and interpretation of overt words, and thus the use of variables is a mere artifact of some theories. Consider the phenomenon of anaphora, however. The items himself and him below can be looked upon as bound variables, but both they and their binders are in every respect normal arguments of the verb, so they cannot be dispensed with:

\[
\begin{align*}
(25) & \quad \text{a. Everyone loves himself.} \\
       & \quad \text{b. Everyone thinks that Mary loves him.}
\end{align*}
\]

Modulo technical details, standard theories of anaphors and pronouns have two components. The lexical component is rather meager; these items are assigned a so-to-say minimal interpretation, namely, that of a free variable. The syntactic component is rather rich: it consists of a mechanism for binding those variables plus a set of constraints on what they can, or must, be bound to. See Chomsky 1981, 1986b; Chierchia 1988; Pollard 1984; among others.
The crucial binding component of these theories is not reproducible in our grammar. The reason is that combinatory logic does not merely allow us to handle specifically extraction and parasitic gap structures without bound variables: it has no variable binding at all.\footnote{That is, we have combinatory terms with the same meaning as \( f(x) \) and combinatory terms with the same meaning as \( \lambda x[f(x)] \), but the latter is not obtained from the former.}

Alongside with the above theories of binding, a number of proposals have been put forth to the effect that (sentences containing) a reflexive can be interpreted as (containing) a kind of argument reducer, in lambda terms, \( \lambda f \lambda x[f(x)] \). Applied to a two-place function, say, \( \lambda z \lambda y[\text{SEE}(z)(y)] \) it returns a one-place function obtained by identifying \( \text{SEE}'s \) arguments, viz., \( \lambda x[\text{SEE}(x)(x)] \). See, for example, Quine 1960, Geach 1972, von Stechow 1979, Keenan 1987, and Kanski 1987.

Given that combinatory logic has the same expressive power as the lambda calculus, this proposal does have a straightforward equivalent in our grammar. The requisite combinator is known as \( W \), the duplicator in Curry and Feys 1958.

\begin{equation}
\text{Duplicator: } \ W = \lambda f \lambda x[f(x)] \\
Wf = fxx
\end{equation}

where \( f \) is of \( P \rightarrow (P \rightarrow Q) \), \( x \) of \( P \)

Notice that \( W \) is the combinator that we defined as \( ST \) in (24a).

The argument reducer proposals in literature have been essentially semantic in nature. They were not intended to account for the syntactic constraints that binding theories are preoccupied with. Keenan 1987 even argues that semantics is all that there should be to it: matters of interpretation are independent of matters of form like constituency.

The question that is interesting to us is this: What happens if we incorporate the duplicator account of anaphors into the grammar outlined in 1.2? In the rest of the paper I will focus on three aspects of this question. In 2.1, I will examine how the duplicator account of reflexives interacts with the syntax of CCG as in (18)–(21) to make predictions concerning “Principle A” properties. In 2.2, I will examine what the theory entails for pronouns, free and bound, and point out that it practically derives Reinhart’s (1983) results. In 2.3, I will extend the proposal to VP-ellipsis to resolve a problem facing any Reinhart-like theory of anaphora.

### 2.1 Reflexives

Standard binding theories attribute three primitive properties to anaphors: the necessity for there to be a binder, the prominence condition on the binder, and the locality condition on the binder. I suggest that the simplest possible account of the first of these is to interpret reflexives as duplicators in the lexicon. Given that reflexives are lexical items in need of some
meaning in any case, any other treatment would involve that we assign the "wrong" lexical meaning to them, and go on to put it "right" in syntax.

Note a problem now: \textit{W} is an operation over functions, whereas \textit{himself} is an argument of the verb. But notice that turning a two-place function into a one-place function is essentially the same as providing one of its arguments. The conceptual gap is bridged by function-argument structure reversal, viz., lifting. This tells us what kind of a noun phrase a reflexive is: nothing but a lifted kind.

Lifting has ample syntactic motivation in our grammar (see especially Steedman 1986 and Dowty 1988), so reflexives are by no means exceptional in having lifted kind of categories. The only peculiarity is that while an entity-denoting item like \textit{Mary} may come from the lexicon with the category NP and get lifted in syntax, the same categories are assigned to reflexives directly in the lexicon, matching their meaning. In this regard the treatment of reflexives is like that of quantifiers. See (27), in which items in the first row are annotated with a category and an interpretation that suit the subject position, and those in the second, the object position. I continue to use lambda terms for the reader's convenience:

\begin{center}
\begin{tabular}{ll}
(27) & \textit{everyone} & \textit{Mary} & \textit{*sheself} \\
& S/(S\backslash NP) & S/(S\backslash NP) & S/(S\backslash NP) \\
& \lambda f \forall y [fy] & \lambda f [fm] & \\
\text{everyone} & \text{Mary} & \text{herself} \\
& (S\backslash NP)/(((S\backslash NP)/NP)) & (S\backslash NP)/(((S\backslash NP)/NP)) & (S\backslash NP)/(((S\backslash NP)/NP)) \\
& \lambda g \lambda y \forall x [gxy] & \lambda g \lambda y [gmy] & \lambda g \lambda x [gxx] \\
\end{tabular}
\end{center}

A simple example is then derived as below. \textit{Herself} is a leftward-looking functor with the duplicator interpretation \(\lambda g \lambda x [gxx]\). It applies to the transitive verb \textit{sees} to yield a verb phrase \(S\backslash NP\) with the interpretation \(\lambda x [SEE(x)(x)]\). The subject quantifier \textit{everyone} is a rightward-looking functor with interpretation \(\lambda f \forall y [fy]\). It applies to the verb phrase and yields a sentence with the desired interpretation \(\forall y[SEE(y)(y)]\).

\begin{center}
\begin{tabular}{ll}
(28) & \textit{Everyone} & \textit{sees} & \textit{herself} \\
& S/(S\backslash NP) & (S\backslash NP)/NP & (S\backslash NP)/(((S\backslash NP)/NP)) \\
& & & \underbrace{A_{ii}} \\
& S\backslash NP & & \underbrace{A_i} \\
& S & & \\
\end{tabular}
\end{center}

Now consider how this proposal accounts for the ungrammaticality of (29a,b):\footnote{(29a,b) cannot be ruled out with reference to a morphological gap. They are equally ungrammatical in Hungarian, a language that has nominative anaphors in the subject position of NP. (On the clause-like structure of the Hungarian noun phrase, see Szabolcsi 1984.)}
As for (29a), the non-existence of sheself of category S/(S\NP) is due to the fact that the duplicator is by definition a two-place function, which cannot be the interpretation of a one-place functor. No such simple semantic explanation can be given for (29b), however. To wit, sheself might also be expected to bear the lexical category (S/NP)/(S\NP)/NP, which may well be interpreted as a duplicator and bind the subject to the object. This is essentially what Keenan points out when he invokes his Nominative Reference Condition to eliminate (29b).

The situation in our grammar is slightly different, however. Notice that the dangerous category above cannot be obtained by lifting NP but only by applying the compositor B to S/(S\NP) unarily. To make this more transparent, in (30) let us use the ad hoc labels subj and obj, and ignore directionality:

\[
(30) \quad a. \frac{S/(S\NP)}{((S\NP)/NP)} = \frac{(S\obj)||(S\subj)|obj}{(S\subj)} \\
\quad b. \ T of subj may be \frac{(S\obj)||(S\obj)|subj}{(S\obj)|subj} \\
\quad c. \ B of S|(S\obj) may be \frac{(S\obj)||(S\subj)|obj}{(S\subj)|obj}
\]

Now, in distinction to Lambek calculi, for instance, the syntax in (18)-(21) does not include unary B (division). We only used B to interpret concatenation. This is not accidental. There are various kinds of disasters, some but not all mentioned in (the revised version of) Szabolcsi 1987, that unary B may cause in syntax. Hence the problematic category is not a standard category for noun phrases in English: even Mary will never acquire it. So all we need to assume in order to exclude (29b) is that reflexives may not have lexical categories that are not available in syntax to normal noun phrases. Which, in fact, is the null hypothesis.

It may be concluded that by lexically interpreting reflexives as duplicators, plus assigning them to categories in conformity to independently motivated syntactic assumptions, we predict that they "are bound by a more prominent argument". But no binding mechanism and c-command condition need to be stipulated. Further details of this proposal, including the treatment of two-complement verb cases, pied piping, and interaction with extraction and coordination are developed in Szabolcsi 1987.

That said about the requirement of a more prominent binder, let us turn to the locality condition on anaphors. It is to be observed that locality cannot appear as a natural condition in a system that includes composition. Consider, for instance:

\[
(31) \quad a. *\text{Mary believes that John loves herself.} \\
\quad b. \text{Who does Mary believe that John loves?}
\]

The extraction structure (31b) shows that believe that John loves of cate-
gory (S\NP)/NP can be obtained by composition. If *herself* in (31a) applies to such a composed transitive verb, it will “get bound” by the subject *Mary*. So, in case the treatment of anaphors is to be part of this system, we must resort to brute force to capture the locality condition. The brute force method is, basically, to require that the duplicator only apply to functors that are lexical in some sense. (Cf. Chomsky’s 1986b minimal complete functional complex.)

Without going into details regarding the precise definition of lexicality, let us ask how sad one should be about this. Is the locality condition part and parcel of the notion anaphor, where by “anaphor” we mean an item that must be bound by a c-commanding argument, i.e., which is interpreted as a duplicator?

The existence of long-distance anaphors has been widely recognized for quite some time. Most of them reside within NPs and are exclusively subject oriented, that is, they appear to have rather peculiar restrictions. A very interesting case from Modern Greek is reported by Iatridou 1986, however. Greek has two anaphors in our sense, of which *ton eafton tou* is bound within, and *ton idhio* outside, its governing category. The data Iatridou presents also indicate that this latter, long-distance anaphor has no funny restrictions:

(32) O Yanis, ipe ston Costa gaghapa ton idhio, said to that loves himself

A plethora of further relevant data can be found in Keenan 1988.

This indicates that locality can in general be divorced from the core notion of anaphor, contrary to what binding theories suggest. The locality condition (lexicality requirement) may really be a brute force device employed by natural language to facilitate processing. Clearly, it is very useful for the hearer if binding ambiguities are reduced by having different forms for the duplicator; but there may be nothing more to it. While my proposal (as it stands, at least) is unrevealing with respect to what locality conditions different languages may impose on their anaphors, it may be taken to be revealing in the sense that it predicts locality to be a more or less ad hoc matter.

2.2 Pronouns

There is another phenomenon that the present theory makes predictions about, namely, bound versus free pronouns.

If the local binding condition on reflexives is a more or less ad hoc matter, there is in principle no obstacle to extending the class of anaphors—i.e., lexical duplicators—to include items which only differ from reflexives in that they are subject to no, or different, locality conditions. This is good news because, given that our combinatory logic has no variable binding (assignment switching) mechanism, having a duplicator kind of meaning is
the only chance for an item to get interpreted as a bound variable in the usual sense. Besides this, combinatory logic only offers free variables. But those are like any name: they start out free and remain free. They may only co-refer with other referential expressions on independent grounds.

These considerations imply that pronouns must be multiply ambiguous.

(33) a. $He$[bound] is a member of the class of anaphors. As a first approximation, it may be assigned the same lexical meaning(s) as reflexives, whence it is subject to the same prominence/constituency requirements, and to possibly different locality conditions.

b. $He$[free] is basically deictic. It represents arbitrarily many different free variables, each having its value fixed once and for all.

Consider now the anti-locality condition on pronouns ("free in its governing category"). The main point to note is this. The relation between $he$[bound] and its antecedent is recognizable within the combinatory theory, but the relation between $he$[free] and the item it happens to co-refer with is not. Hence we may hope to be able to impose an anti-locality condition only on the former, but not on the latter.

Anti-locality for $he$[bound] can be captured by something like the opposite of the lexicality requirement for reflexives or, more interestingly, by utilizing a combinator that is independently necessary for standard cases of pied piping. The combinatory equivalent of having a feature inherited by mother from daughter is discussed in Szabolcsi 1987, following Steedman (p.c.).

(34) If a noun phrase is interpreted as $a$, its pied piper version is interpreted as $C(B(Ba)B)$ or $B(Ba)B$, and has a matching category. For example: $X(X/NP) -$its pied piper version: $(X/(X/Y))(Y/NP)$. For instance, if $a$ is $W$ and $Y$ is PP, the use of the pied piper category will allow to himself to inherit anaphorhood from himself. But notice now that the "feature" that "percolates" from NP to $Y$ is necessarily "inert" within $Y$. Thus, by making pronouns obligatory S-pied pipers, we let $S$ inherit anaphorhood from $he$ and $him$ and, at the same time, we guarantee that $him$ has no antecedent within its minimal $S$. So let us assume that $he$[bound] has no simple $W$ interpretation but is at least $C(B(BW)B)$ or $B(BW)B$, viz., $\lambda g \lambda f \lambda x[f(gx)(x)]$.

(35) a. Everyone thinks he[bound] likes John

<table>
<thead>
<tr>
<th>$S/VP$</th>
<th>$VP/S$</th>
<th>$S/\text{NP}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$VP/(VP/S)/(S/\text{NP})$</td>
<td>$\lambda f \lambda x [f(\text{likes}(j)(x))(x)]$</td>
<td></td>
</tr>
</tbody>
</table>

$^6$C is the permutator, $B(T(BBT))(BBT)$ or $\lambda f \lambda x \lambda y [f(yx)]$.

$^7$Pace Pollard's "Evidently, there is no principled analysis of pied piping in an extended categorial framework like Steedman's without the addition of a feature-passing mechanism for unbounded dependencies" (1988, 421).
b. Everyone thinks John likes him[bound].

\[
\begin{align*}
S/VP & \quad VP/S & \quad S/VP & \quad VP/NP & \quad (VP\backslash (VP/S))\backslash (S/NP) \\
VP/\backslash (VP/S) & \quad (S/VP) & \quad S/NP & \quad \text{B}_i & \quad A_{ii} \\
\end{align*}
\]

He/him[bound], the S-pied piper, can only combine with S/NP or S\NP, whence it cannot be bound to anything within its own clause. As was pointed out above, anti-locality for he/him[free] cannot be accommodated in this way and must therefore be attributed to independent mechanisms.

The picture emerging here is not only technically but also conceptually different from that of standard binding theory, which specifies only that pronouns are free in their governing category. However, it squares rather well with an alternative theory proposed in Reinhart 1983. Reinhart argues that the conflation of coreference and bound variable interpretation leads to enormous complications because the two are empirically different. A pronoun may be coreferential with another referential expression even if the latter does not c-command it; moreover, their (non-)coreference is affected by pragmatic factors. On the other hand, a pronoun can only be bound by a quantifier that c-commands it from a distance at s-structure: i.e., bound variable interpretation is as strictly syntactic as reflexivization. In conclusion, she proposes to distinguish the syntactic theory of bound anaphora from the pragmatic theory of coreference. See also Partee 1978.

It is extremely interesting to note here that acquisition studies by Wexler and Chien from 1988 provide evidence in support of Reinhart’s distinctions. In previous experiments they had found that English-speaking children’s performance on locality properties of reflexives increases steadily from age 2;6 to almost perfect performance at 5;6. On the other hand, children in the age range 5;6 to 6;6 still do not show that they have the knowledge that a pronoun may not have a local c-commanding antecedent. However, in the study reported in this paper they tested the hypothesis that, in line with Reinhart’s proposal, principle B really applies to bound, and not coreferential, pronouns. They found that children at age 6 violate principle B approximately 50% of the time with coreference, but less than 15% of the time in the case of a bound variable pronoun.⁸

⁸The significance of Chien and Wexler’s earlier results has been challenged by Grimshaw and Rosen 1988: they argue that children know, although they may not obey, principle B. The 1988 study involving bound variables is not subjected to criticism in Grimshaw and
These results, if correct, are especially important for combinatory grammar. As was pointed out, the above treatment of reflexives and bound versus free pronouns is the only kind of treatment the logical foundations make available. Therefore it makes a much stronger case than Reinhart’s theory, which formulates empirical observations within a grammar explicitly incorporating the lambda-calculus.

Despite its acknowledged appeal, Reinhart’s theory has been criticized in the literature from various angles. In this paper I will not be concerned with the problem raised by the possibility of a pronoun being bound by a quantifier that does not c-command it at s-structure. (See a suggestion in the revised version of Szabolcsi 1987.) Rather, I will turn directly to a third crucial binding phenomenon in Reinhart’s theory, viz., VP-ellipsis.

2.3 VP-Ellipsis

It is well-known that elliptical VPs may be ambiguous between the “strict” and the “sloppy” readings in the following kind of context:

(36)  
Felix hates his neighbors and so does Max.

   a. sloppy reading: “and Max hates Max’s neighbors”
   b. strict reading: “and Max hates Felix’s neighbors”

Reinhart (1983, 150–51) points out that the sloppy reading is obtained if Felix binds his in the antecedent clause, and the strict reading is obtained if his is merely coreferential with Felix. Reinhart is primarily interested in the sloppy case, and her pertinent claim is generally accepted. On the other hand, her claim concerning the strict reading faces a serious problem. If the strict reading is contingent on accidental coreference, then it is predicted to be available only if the antecedent itself is referential. But this prediction is refuted by quantificational antecedents, which also support the strict reading in a slightly different context, viz., when we are not dealing with coordination:

(37)  
Every man mentioned his merits before Mary did.

   a. sloppy: “before Mary mentioned her merits”
   b. strict: “before Mary mentioned his merits”

Gawron and Peters (1990) discuss this point extensively and take it to be one important piece of evidence against the correctness of Reinhart-like theories of pronominal anaphora. They propose a system whose innovations include the postulation of three, rather than two, kinds of anaphoric relations: co-variation (cf. co-reference), role-linking (cf. binding) and, crucially, co-parametrization for cases like (37b). The reader is referred to their book for details.

Rosen’s paper, however, and it appears to me that Grimshaw and Rosen’s findings are perfectly compatible with the results concerning the differential acquisition of conditions on binding and coreference.
Let us see what the combinatory theory of anaphora has to say about example (37). At first glance, it appears to be impossible to give an account of (37b). The reasoning, along the lines of Gawron and Peters, goes as follows. We have binding in the first clause, so its VP must be interpreted as $\lambda x[\text{mentioned}(x\text{'s merits})(x)]$. If the ellipted VP echoes this, we get the sloppy reading; fine. In order to get the strict reading, we should start out with an antecedent VP meaning containing a free variable, echo this, and then bind both occurrences of that variable. But we cannot get free variables bound. Hence (37b) seems disastrous. In a sense, it is even more disastrous for this theory than for Reinhart's since, as I noted above, what is stipulated in her theory is a matter of logical necessity here.

This reasoning has an important presupposition, namely, that strict readings with non-referential antecedents need to be accounted for specifically by the machinery for pronominal anaphora and, therefore, if the pronominal anaphora machinery has nothing to say about it, the availability of this reading is evidence against the machinery.

In what follows I will suggest that this presupposition is false, and hence the reasoning is wrong. First, I will observe that strict readings with non-referential antecedents arise even in contexts without any overt anaphora. Therefore the existence of (37b) does not directly reflect on the viability of proposals concerning pronouns. Second, I will ask whether the treatment of the natural class of phenomena to which (37b) belongs really requires that we go beyond the tools of combinatory grammar and introduce a novel device like co-parametrization. I will argue that it does not. I will outline an account in CCG, and point out that it requires only a minimal extension of what we already have, because good old composition plus duplication will do "co-parametrization" for us. In this paper I will not investigate how other theories that do not have these would derive the critical examples.

The first step is entirely theory-independent. The consideration of two examples will suffice to show that the problem and, consequently, the treatment of (37b) is independent of the specific abilities of pronouns. One is (38):

(38)  *Every man* mentioned *himself* before Mary did.
   a. sloppy: “before Mary mentioned herself"
   b. strict: “before Mary mentioned him”

Given that (38) contains a reflexive, it shows that we must be able to produce the strict reading with a non-referential antecedent even under the canonically syntactic conditions reflexive binding is subject to. Second, consider (39):

(39)  Who did you mention before Mary did?
    "Which $x$, you mentioned $x$ before Mary mentioned $x$"
As the interpretation makes clear, (39) presents precisely the same problem as (37b) and (38b). But it is a simple extraction structure, so the problem must be independent of the treatment of overt anaphoric elements on the whole.

With these observations in mind we may now proceed and see how the phenomenon can be handled in CCG. It is to be emphasized that in this paper I will not develop a detailed account of VP-ellipsis but concentrate only on the points relevant for the argument as sketched above. The first task is to accommodate the simplest case of VP-ellipsis, with which we start:

(40) John left before Mary did.

Observe that the interpretation of (40) involves duplication, namely: before(left(Mary))(left). One way to capture this would be to say that did is a duplicator, i.e., it has the same meaning as items like himself, but its typing is different: the function whose arguments it identifies has VP, rather than NP, arguments. This proposal encounters two kinds of difficulty. One is the fact that many of the pertinent examples work also in the absence of did (John left before Mary, etc.). The second is that this proposal would extend easily to (39) but, for rather technical reasons, not to (38b). Therefore I will largely ignore the contribution of did and use the same W as the interpretation of a syntactic operation. Technically, W can be implemented as a new interpretation for concatenation, see (41a), or as a unary (type-change) rule that applies to the material in the second clause, as in (41b). Given that here the use of W is followed by application, the choice makes no difference. I sketch both options because binary W may be more intuitive, but subsequent steps will have to utilize the unary one.

(41) a. left before Mary did

\[ \begin{align*}
\text{S\NP} & \quad ((\text{S\NP})\,(\text{S\NP}))\,/\,\text{S} \quad \text{S/(S\NP)} \\
& \quad ((\text{S\NP})\,(\text{S\NP}))\,/\,(\text{S\NP}) \\
& \quad \text{B}_{i} \\
& \quad \text{W}_{ii} \\
\text{S\NP} & \quad ((\text{S\NP})\,(\text{S\NP}))\,/\,(\text{S\NP}) \\
& \quad \text{A}_{ii} \\
\end{align*} \]

For the reader’s convenience, in what follows I will spell out the interpretation of each line in lambda terms. Intermediate steps will be omitted if given in a previous example:
(41') a. \( B(\text{before})(\text{Mary did}) \)
\[ = \lambda a b \lambda c [a(b)c](\lambda p \lambda g \lambda x [\text{before}(p)(g)(x)](\lambda f [f(m)]) \]
\[ = \lambda b \lambda c [\lambda p \lambda g \lambda x [\text{before}(p)(g)(x)](\lambda f [f(m)]) \]
\[ = \lambda b \lambda c \lambda g \lambda x [\text{before}(b c)(g)(x)](\lambda f [f(m)]) \]
\[ = \lambda c \lambda g \lambda x [\text{before}(\lambda f[f(m)](c))(g)(x)] \]
\[ = \lambda c \lambda g \lambda x [\text{before}((c m)(g))(x)] \]

\( W(B(\text{before})(\text{Mary-did}))(\text{left}) \)
\[ = \lambda a b [a b b](\lambda c \lambda g \lambda x [\text{before}(c m)(g)(x)](\text{left})) \]
\[ = \lambda b [\lambda c \lambda g \lambda x [\text{before}(c m)(g)(x)](b)(b)](\text{left}) \]
\[ = \lambda b \lambda x [\text{before}(b m)(b)(x)](\text{left}) \]
\[ = \lambda x [\text{before}(\text{left}(m))(\text{left})(x)] \]

b. \( B(\text{before})(\text{Mary-did}) = \lambda c \lambda g \lambda x [\text{before}(c m)(g)(x)] \)

\( W(B(\text{before})(\text{Mary-did})) \)
\[ = \lambda a b [a b b](\lambda c \lambda g \lambda x [\text{before}(c m)(g)(x)](\text{left})) \]
\[ = \lambda b [\lambda c \lambda g \lambda x [\text{before}(c m)(g)(x)](b)(b)](\text{left}) \]
\[ = \lambda b \lambda x [\text{before}(b m)(b)(x)](\text{left}) \]
\[ = \lambda x [\text{before}(\text{left}(m))(\text{left})(x)] \]

This accommodates the simplest case of VP-ellipsis. Now consider (39):

(39) Who did you mention before Mary did?

Sentence (39) presents a problem on its own right. \( W \) echoes a VP-meaning but, the direct object being extracted, there is no VP in the antecedent clause. Recall that our grammar has no traces.

The problem of (39) is in fact easy to solve with the tools our grammar has had all along. In (41b) the segment \( W(B(\text{before Mary})) \) is combined with \( \text{left} \) by application. But it can equally well combine with \text{mention} by composition—and that is all we need. A straightforward execution of this idea is given in (42a). Once more, (42b) presents an alternative that makes no difference here but will underlie the next derivation. In (42b) a unary combinator \( BBW \) is used, followed by application. \( BBW \) is obtained by composing the above motivated steps \( W \) and \( B \), and serves no other end than lumping these two together in the said order. For its expression in lambda terms, recall (24b).

(42) a. \text{mention} before Mary did
\[
\begin{array}{c}
(S\backslash NP)\backslash NP \\
((S\backslash NP)(S\backslash NP))/S \\
S/(S\backslash NP) \\
\end{array}
\]
\[
\begin{array}{c}
(B_i) \\
((S\backslash NP)(S\backslash NP))/(S\backslash NP) \\
W \\
(S\backslash NP)(S\backslash NP) \\
\end{array}
\]
b. mention before Mary did
(S\NP)/NP \rightarrow ((S\NP)\(S\NP))/S S/(S\NP)

\[
\begin{array}{c}
((S\NP)\(S\NP))/S/S/(S\NP) \\
\hline
((S\NP)/NP)/(S\NP)/NP
\end{array}
\]

Note the replacement of the two occurrences of \(b\) in (42'\(a\)) with \(cd\): this has the effect of “co-parametrization”.

(42') a. \(W(B(\text{before})(\text{Mary-did})) = \lambda b\lambda x[\text{before}(bm)(b)(x)]\)

\[
B((W(\text{before})(\text{Mary-did}))(\text{mention})
= \lambda a\lambda c\lambda d[a(cd)](\lambda b\lambda x[\text{before}(bm)(b)(x)])(\text{mention})
= \lambda c\lambda d[\lambda b\lambda x[\text{before}(bm)(b)(x)]][cd](\text{mention})
= \lambda c\lambda d\lambda x[\text{before}(cdm)(cd)(x)](\text{mention})
= \lambda d\lambda x[\text{before}(\text{mention}(d)(m))(\text{mention}(d))(x)]
\]

b. \(B(\text{before})(\text{Mary-did}) = \lambda c\lambda g\lambda x[\text{before}(cm)(g)(x)]\)

\[
BBW = \lambda f\lambda h\lambda y[f(hy)(hy)]
\]

\[
BBW(B(\text{before})(\text{Mary-did}))
= \lambda f\lambda h\lambda y(f(hy)(hy))[(\lambda c\lambda g\lambda x[\text{before}(cm)(g)(x)])]
\]

\[
BBW(B(\text{before})(\text{Mary-did}))(\text{mention})
= \lambda y\lambda x[\text{before}(\text{mention}(y)(m))(\text{mention}(y))(x)]
\]

In view of the informal suggestion made at the outset, with these we must have everything ready to cater to the ambiguity with anaphors. To avoid the treatment of possessive pronouns, I will only consider (38) here.

The sloppy reading is derivable by mimicking (41b). For compactness, \(S\NP\) will be abbreviated as \(VP\) whenever this does not affect intelligibility.

(43) mention himself before Mary did
\(\text{VP}/\NP\) \(\text{VP}/(\text{VP}/\NP)\) \(\text{VP}/(\text{VP}/\NP)\)
\[A_{ii}\]
\[B_{i}\]

\(\text{S}/\NP\)

(43') \(W(B(\text{before})(\text{Mary-did})) = \lambda b\lambda x[\text{before}(bm)(b)(x)]\)

\[
himself = W = \lambda f\lambda y[ffy]
\]

\[
W(\text{mention}) = \lambda f\lambda y[ffy](\text{mention}) = \lambda y[\text{mention}(y)(y)]
\]

\[
W(B(\text{before})(\text{Mary-did}))(W(\text{mention}))
= \lambda b\lambda x[\text{before}(bm)(b)(x)][\lambda y[\text{mention}(y)(y)])
= \lambda x[\text{before}(\text{mention}(m)(m))(\text{mention}(x)(x)])
\]
Consider now the strict reading. The correct semantic result obtains if we apply *himself*, the duplicator to the interpretation of *mention before Mary did* derived in (42):

\[
(44) \quad W(BBW(B(before)(Mary-did))(mention))
\]

\[
= \lambda f \lambda z[fzz](\lambda y \lambda x[before(mention(y)(m))(mention(y))(x)])
\]

\[
= \lambda z[\lambda y \lambda x[before(mention(y)(m))(mention(y))(x)](z)(z)]
\]

\[
= \lambda z[before(mention(z)(m))(mention(z))(z)]
\]

We could indeed do this and thus derive the strict reading by simply mimicking (42) if we had a wrap operation in syntax:

\[
(45) \quad \text{mention himself before Mary did}
\]

As a matter of fact, the grammar I am working with does not have wrap. This is not the only case for which wrap would be needed, however. An account of VP-internal anaphora presupposes that the neutral linear order of complements in English is the reverse of their semantic order. Therefore in Szabolcsi 1987, a simulation of wrap is developed, using forward mixing composition (i.e., disharmonic composition with the direct object acting as principal functor). Since that is an operation we otherwise do not want to set free in syntax, it is pushed back into the lexicon in the form of unary B with idiosyncratic slashing. Using this lexically derived category, wrap can be broken into two concatenation steps.\(^9\)

With these technicalities in mind, consider (46). Notice that it utilizes, and thus provides the ultimate motivation for, the composite combinator BBW developed in (42b):

\[
(46) \quad B_{tex}(\text{himself})
\]

\[
(VP\backslash(VP/NP))\backslash((VP/NP)\backslash(VP/NP))
\]

\[
\text{everyone mentioned before Mary-did}
\]

\[
S/(S\backslash\NP) \quad VP/\NP
\]

\[
\text{B} \quad \text{BBW}
\]

\[
(VP/\NP)\backslash(VP/\NP)
\]

\[
\text{VP} \quad \text{A}_i
\]

\[
(VP/\NP) \quad \text{A}_{ii}
\]

\[
S
\]

\(^9\)This is not the only, or even the best, imaginable simulation of wrap in CCG. It is possible that a nicer solution will also allow us to dispense with BBW and, hence, W as the interpretation of a syntactic rule in general.
Finally, note that this treatment does not carry over to VP-ellipsis in coordination or in discourse. But that makes just the right prediction. In contradistinction to (36) with referential Felix, (47a) and (47b) have no strict readings:

(47) a. Every man corrected himself, and so did Mary.

b. Every man corrected himself. So did Mary.

Let me summarize what we have done as follows. We need some new tool to account for the simplest case of VP-ellipsis, viz., *John left before Mary did*. We choose this tool to be the rule W. Once we add that, the tools we have always had in the grammar, composition (B) in particular, will allow us to derive *Who did you mention before Mary did?*. This latter, in conjunction with the apparatus necessary for pure reflexivization, will derive the sloppy as well as the strict readings of *Everyone mentioned himself before Mary did*. In this way, the existence of the strict reading of VP-ellipsis with non-referential antecedents ceases to constitute an argument against Reinhart-like theories of pronominal anaphora in general and against the combinatory theory in particular.

3 Conclusion

In Section 1 a specific problem was raised concerning the implementation of the Projection Principle. It was pointed out that in various theories of grammar the nature of syntactic representations is underdetermined by the nature of lexical items. An examination of accounts of extraction and parasitic gaps suggested that taking lexical items to be functions allows us to establish a significant link between the two. The “function
metaphor" was then shown to lead to the "combinator metaphor" in a natural way.

In Section 2 the combinatory proposal was confronted with a new phenomenon, namely, anaphora. A selection of specific problems concerning reflexives, pronouns, and VP-ellipsis was examined. At the technical level it was argued that combinatory logic offers a simple lexical-based treatment of anaphora which, in conjunction with the grammar independently motivated in Section 1, factors out and derives a set of coherent properties of anaphora without any specific stipulation. At the intuitive level I hope to have shown that the use of combinators in grammar has its own heuristic value and offers genuine insights.

References


