17

Direct Compositionality and the Architecture of LFG

Ash Asudeh

17.1 Introduction

The principle of compositionality is arguably the foundational principle of formal semantics. It is quoted in (1) from Janssen (1997:419), but can be found in innumerable sources in much the same formulation.

(1) *The Principle of Compositionality*
The meaning of a complex expression is a function of the meanings of its parts.

This principle is often called 'Frege's Principle', although it is doubtful whether Frege himself formulated it (Janssen 1997, Hodges 1998, 2001).[1] Hodges (2001:7) suggests that compositionality in the modern sense is more readily attributable to Tarski (1983 [1935]), the foundational work in truth-conditional semantics. Heim and Kratzer (1998:1–3) point out that Tarskian truth-conditional schemas can only be informative in light of compositionality.

Despite its generally acknowledged importance to modern semantic theory, compositionality is in danger of becoming a shibboleth, because it is typically formulated sufficiently broadly that just about any semantic theory would satisfy it in some sense or other.[2] In this light, recent

[1] Nevertheless, the principle is clearly in the spirit of Frege's later works, hence the common attribution (Janssen 1997:421).

[2] Zadrozny (1994) notes that 'the standard definition of compositionality is formally vacuous', given his theorem that any semantics can be made compositional.

Intelligent Linguistic Architectures: Variations on themes by Ronald M. Kaplan.
Miriam Butt, Mary Dalrymple, and Tracy Holloway King (eds.).
Copyright © 2006, CSLI Publications.

work by Jacobson (1999, 2002, 2004, 2005) on 'the hypothesis of Direct Compositionality' is important, because it features an in-depth defense of one well-articulated substantive conception of compositionality that simultaneously suggests why certain other modern approaches do not satisfy the conception. The non-directly compositional approach that forms Jacobson's main target is semantics based on Logical Form (LF) in Principles and Parameters Theory (P&P; Chomsky 1981, 1995). May (1977) is an early and influential precursor of the LF approach, but its principal modern articulation is Heim and Kratzer (1998). Lexical Functional Grammar (LFG; Kaplan and Bresnan 1982) superficially appears to similarly fall into the non-directly compositional class of formalisms, due to its postulation of a grammatical level of *semantic structure* (s-structure) in its parallel projection architecture (Kaplan 1987, Halvorsen and Kaplan 1988, Kaplan 1989, Dalrymple 1993).

In this paper, I argue that this superficial impression is incorrect, expanding on some remarks in Asudeh (2005:433–439). Both that initial treatment of the problem and this expanded treatment are based on Ron Kaplan's foundational work on LFG's grammatical architecture. Rather than entering into a direct comparison of LF in P&P with s-structure in LFG, I will show that strings in LFGs can be assigned a directly compositional interpretation. Therefore, despite whatever similarities between LF and semantic structure suggest themselves without delving deeper, LFG grammars are *not* outside the class of directly compositional grammars in Jacobson's sense. Nevertheless, I will argue, based on Kaplan's insights, that a grammatical architecture like LFG's — an architecture that posits many intermediate structures between form and meaning, but which crucially treats the structures as eliminable — is preferable to an architecture that allows *only* a very direct mapping between form and meaning.

The paper is structured as follows. In section 17.2, I present Kaplan's notion of a parallel projection architecture and a synthesis of subsequent LFG-theoretic architectural proposals in the literature (based on Asudeh 2004:32–35, with some modifications). In section 17.3, I present Jacobson's work on direct compositionality. Then, in section 17.4, I present the apparent problem that Jacobson's work poses for LFG and show how the problem can be resolved. Lastly, in section 17.5, I take the opposing tack and consider the hypothesis of direct compositionality in light of LFG's grammatical architecture.

17.2 The Parallel Projection Architecture

The original architecture of LFG (Kaplan and Bresnan 1982) consisted of two syntactic levels: constituent structure (c-structure) and functional structure (f-structure). C-structures are represented as trees, which are described in the usual manner (with a set of nodes, a labeling on the set, and functions for dominance and precedence). The level of c-structure represents syntactic information about precedence, dominance, and constituency. F-structures are represented as feature structures (attribute-value matrices), described by a set of recursive functional equations on a set of symbols. The level of f-structure is another aspect of syntactic representation — it is not a semantic representation. However, f-structure represents more abstract aspects of syntax, such as grammatical functions, predication, subcategorization, and local and non-local dependencies. C-structure and f-structure are projected from lexical items, which specify their c-structure category and f-structure feature contributions. Variables in lexical items are instantiated by the c-structure parse. The two syntactic representations are present simultaneously, in parallel. They are related by the ϕ projection function, also known as a correspondence function. The ϕ function maps c-structure nodes (i.e., tree nodes) to f-structure nodes (i.e., feature structures). The original grammatical architecture of LFG is shown schematically in (2).

(2) *The original LFG architecture*:

$$\text{constituent structure} \xrightarrow{\phi} \text{functional structure}$$

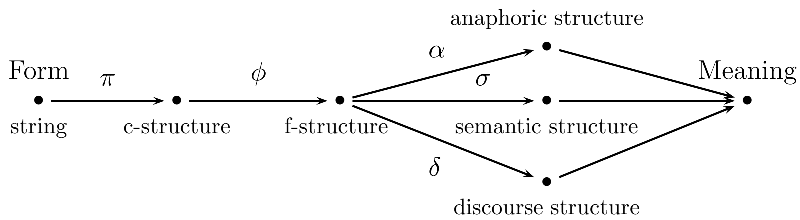
An LFG representation of an expression on this view is a triple consisting of a c-structure, an f-structure and a ϕ projection function that maps the c-structure to the f-structure: $\langle c,\ f,\ \phi\rangle$.

C-structures and f-structures are constructed by simultaneous constraint satisfaction. LFG is a declarative, non-transformational theory. The fact that c-structure and f-structure are represented using distinct data structures (trees and feature structures) distinguishes LFG from both transformational theories such as P&P, which represents all syntactic information in a tree, and non-transformational theories such as Head-Driven Phrase Structure Grammar (Pollard and Sag 1987, 1994), which represents all grammatical information, whether syntactic or not, in a directed acyclic graph. LFG uses mixed data structures related by structural correspondences, rather than a single monolithic data structure.

The LFG architecture was subsequently further generalized to a parallel projection architecture (Kaplan 1987, Halvorsen and Kaplan 1988, Kaplan 1989). According to this architecture, there are various levels of linguistic representation (not just syntactic ones) called *projections* that are present in *parallel* and are related by structural correspondences (i.e., projection functions) which map elements of one projection onto elements of another. C-structure and f-structure are still the best-understood projections, but they are now two among several levels of representation and the projection function ϕ is now one of many. For example, f-structures are mapped onto s(emantic)-structures by the σ-function (Halvorsen 1983, Dalrymple 1993, Dalrymple et al. 1999b, Dalrymple 2001).

Kaplan (1987, 1989) gives (3) as a hypothetical example of the projection architecture, representing the decomposition of a single mapping, Γ, from form to meaning.

(3) *Kaplan's hypothetical parallel projection architecture*:



Two of the projections proposed in (3) — anaphoric structure and discourse structure — never received much further attention in the LFG literature, at least not in the way that Kaplan originally suggested. Anaphors have been handled at semantic structure (Dalrymple 1993, 2001), and discourse structure has been pursued instead as information structure (i-structure; Butt and King 2000), which encodes notions like discourse topic and focus and old and new information.

Importantly, the correspondence functions between levels can be composed (see below for details), since the domain of each successive function is the range of the previous one. This is summarized in the following passage from Kaplan (1987:363):

> Although the structures related by multiple correspondences might be descriptively or linguistically motivated levels of representation, justified by sound theoretical argumentation, they are formally and mathematically, and also computationally, eliminable . . . Obviously there is a structural correspondence that goes from the word string to the f-structure, namely the composition of π with ϕ. . . . So as a kind of formal, mathematical trick, you can say 'Those intermediate levels of representation are not real, they are just linguistic fictions, useful for stating the necessary constraints'.

There are two key points in this passage. First, intermediate levels are eliminable through composition of correspondence functions. Second, although such elimination is possible, it may nevertheless be desirable to have separate levels. I will pick up on both of these points in sections 17.4 and 17.5 below.

Kaplan observes that we can compose π and ϕ to go directly from strings to f-structures. We can further compose $\pi \circ \phi$ with σ, moving directly from the string to semantic structure. The nature of these mapping functions is important to consider. The postulation of a projection function is tantamount to the claim that there is a function from a structure of type A to a structure of type B. The range of the function may, however, be the empty set. Or there may be more than one such function. Each projection function therefore represents a family of functions. For example, consider the mapping π from strings to c-structures. For each string there is a π function mapping the string to c-structure. An unparseable string — one that has no structural analysis — will not be mapped to anything by π. A parseable but unambiguous string will be in the domain of exactly one π function. An ambiguous string will be in the domain of more than one π function. Similarly, a string may have only one c-structure, but there may be multiple instances of the ϕ mapping if the c-structure is f-structurally ambiguous. The same comments apply to the σ function from f-structure to s-structure and all the other projection functions.

The various levels of grammatical representation in the projection architecture are simultaneously present, but each level is governed by its own rules and representations. This separation of levels allows one to make simple theoretical statements about just the aspects of grammar that the level in question models. It is also possible to split up correspondences in novel ways. Since the projection functions are functions in the mathematical sense, we can always regain the original function through composition of the new functions. This is exemplified by the Butt et al. (1997) proposal for argument structure, discussed below, which separates the original ϕ function into α and λ functions. Another important feature of this architecture is that there can be systematic mismatches between grammatical levels. For example, null pronoun subjects in pro-drop languages are not present at c-structure, because they are unmotivated by the aspects of syntax that are represented at that level. Rather, null pronouns are present at f-structure, where they can participate in agreement, binding, and other syntactic processes modeled at that level.

Although the exact specification of the projection architecture is not the main point of this paper, it is useful from a general LFG-theoretic perspective to stop and take stock of certain subsequent augmentations that have been proposed in the LFG literature. Information structure, the alternative to discourse structure mentioned above, is just one of several subsequent proposals for new projections. Three other proposals are argument structure (a-structure; Butt et al. 1997), morphological structure (m-structure; Butt et al. 1996, 1999, Frank and Zaenen 2002) and phonological structure (p-structure; Butt and King 1998), the latter of which should perhaps be called prosodic structure, since it is concerned with phrasal phonology and prosody. Butt et al. (1997) propose that argument structure should be interpolated between c-structure and f-structure, with the ϕ projection function broken up into the α function from c-structure to a-structure and the λ function from a-structure to f-structure. The original ϕ function would then be the composition of these two new functions: $\phi = \alpha \circ \lambda$ (this will be slightly revised below, in light of m-structure). Information structure and phonological structure have both been proposed as projections from c-structure. There has been some debate over the proper location for morphological structure in the architecture. Butt et al. (1996, 1999) treat it as a projection from c-structure. Frank and Zaenen (2002) argue that although this is adequate for the phenomena for which Butt et al. (1996, 1999) use morphological structure (auxiliaries), there are reasons to prefer morphological structure as a projection from f-structure. I assume that morphological information should feed both argument structure and functional structure; I therefore place m-structure between c-structure and a-structure. This also means that Butt et al. (1997)'s α projection function now maps from m-structure to a-structure, rather than from c-structure to a-structure (their original α is the composition of μ and my α). The original ϕ function of Kaplan and Bresnan (1982) is thus the composition of μ, α and λ (that is, $\mu \circ \alpha \circ \lambda$).

Figure 1 shows an architecture resulting from the addition of these proposals to Kaplan's hypothetical architecture in (3) (note that anaphoric structure and discourse structure have been removed). The architecture in figure 1 is considerably more complex than the original LFG architecture in (2), or even the initial parallel architecture in (3). However, it rests on Kaplan's simple, but powerful, fundamental idea: there is a series of functions, the domain of each subsequent one being the range of the previous one, that map from linguistic form to linguistic meaning.

Let me spell out the mapping to semantics in figure 1 in a little more detail. First, I will define a function that captures the mapping from c-structure to s-structure that is represented by the smaller functions. This function has three components, representing the three paths

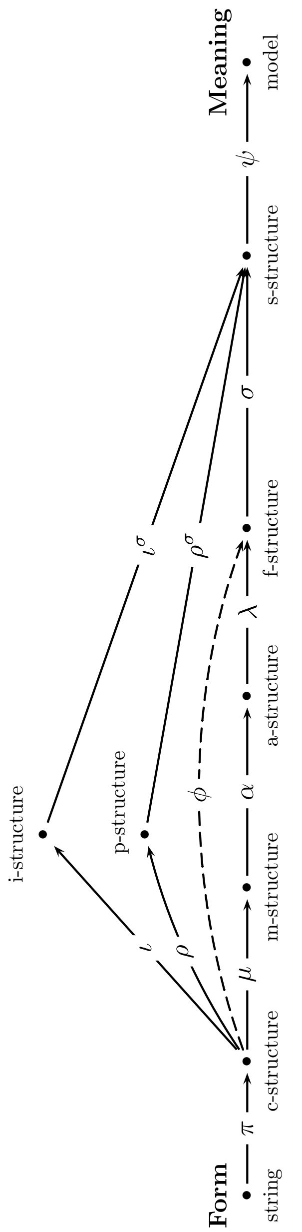


FIGURE 1 The parallel projection architecture (incorporating certain recent proposals)

of information flow from c-structure to s-structure: via morphological structure, argument structure and functional structure, via information structure, and via phonological structure. The function is thus a mapping from c-structure nodes to a triple of s-structure information. Let us call the function Σ ('Big Sigma') in homage to the original σ function. Big Sigma can be characterized as follows:

(4) $\Sigma = \lambda y. \langle (\phi \circ \sigma)(y),\ (\iota \circ \iota^{\sigma})(y),\ (\rho \circ \rho^{\sigma})(y) \rangle$
where y is a c-structure node

Note that the ϕ function in the body of Big Sigma is the new ϕ (i.e., $\mu \circ \alpha \circ \lambda$).

As in other theories of grammar, most work on LFG semantics has focused on the mapping from syntax to semantics, leaving aside the semantic contributions of information structure and phonological/prosodic structure. Thus, semantics in LFG has focused on the first member of the Big Sigma triple, which we can access via projection on the triple:

(5) $\mathsf{first}(\Sigma) = \lambda y.(\phi \circ \sigma)(y)$

In Glue Semantics for LFG (Dalrymple 1999, 2001), s-structure nodes and lexically-defined logical operations on s-structure nodes form the input to a linear logic (Girard 1987) proof of an expression's semantics. Linear logic provides the 'glue language' that specifies how meanings are put together; that is, linear logic is the logic of semantic composition. The linear logic proof is directly related to a model-theoretic semantics via the Curry-Howard Isomorphism between formulas and types (Curry and Feys 1958, Howard 1980). This will be illustrated with respect to a specific example in section 17.4.

The crucial point, though, is that semantic structure forms the input to semantic composition. It thus seems that semantic structure is an indispensable pre-semantic level of representation, on a par with LF in Principles and Parameters Theory. In the next section, I review Jacobson's criticism of LF semantics based on the hypothesis of direct compositionality. Then, in section 17.4, I show that LFG semantic structure is not analogous to Logical Form in the relevant sense through a demonstration that LFG semantics can satisfy 'Strong Direct Compositionality' (Jacobson 2002).

17.3 Direct Compositionality

The hypothesis of Direct Compositionality (DC) has been discussed in some detail in recent work by Jacobson (1999, 2002, 2004, 2005). Jacobson (1999) characterizes the hypothesis as follows:

> [S]urface structures directly receive a model-theoretic interpretation without being mapped into another level (i.e., LF). (Jacobson 1999:117)

In later work, Jacobson (2002, 2004, 2005) characterizes DC slightly differently:

> [T]here is a set of syntactic rules which prove the well-formedness of the set of sentences (or other expressions) in the language . . . Coupled with each syntactic rule is a semantic rule specifying how the meaning of the larger expression is derived from the meaning of the smaller expressions. (Jacobson 2002:603)

The latter form of DC is part of Jacobson's characterization of *Strong Direct Compositionality*, which is one of three successively weaker notions of DC, the other two being *Weak(er) Direct Compositionality* and *Deep Compositionality* (Jacobson 2002). However, it is clear from Jacobson's latest work (2004, 2005), that the notion of compositionality laid out in the second quote is meant as a characterization of the general form of DC, as evidenced by the following introductory passage from Jacobson (2004):

> The hypothesis of Direct Compositionality. . . is that the syntax and semantics work "in tandem". The syntax is a system of rules . . . which prove the well-formedness of linguistic expressions while the semantics works simultaneously to provide a model-theoretic interpretation for each expression as it is proved well-formed in the syntax.

There are thus two characterizations of the DC hypothesis (the one in the first quote and the one in the second two quotes); these can be summed up as follows:

(6) Hypothesis of Direct Compositionality
  a. Surface structure is directly model-theoretically interpreted without mapping to an intervening level.
  b. Model-theoretic interpretation is a function of syntactic well-formedness.

Jacobson tends to treat these two characterizations of DC equivalently, but they are logically distinct. LFG semantics upholds (6b), but it seems similar to LF semantics in contravening (6a).

The second characterization of DC, construed broadly, seems to be a corollary of the principle of compositionality. As Janssen notes, a 'more precise version' of the principle (Janssen 1997:462) than the version quoted in (1) involves reference to syntax, as in the following formulation from Partee et al. (1993:316):

(7) *The Principle of Compositionality (version 2)*
The meaning of a complex expression is a function of the meanings of its parts and of the syntactic rules by which they are combined.

It is clear, though, that one can have a syntax–semantics architecture that respects (7), and therefore clause (6b) of DC, without respecting clause (6a). In fact, the very system that Jacobson (1999:117) cites as a crucial early exemplar of the direct compositionality approach — Montague's semantics in *The proper treatment of quantification in ordinary English* (PTQ; Montague 1973) — is an instance of the separation of (6a) and (6b). In PTQ, strings of English are first translated into expressions of intensional logic (IL), and it is these IL expressions that are subsequently model-theoretically interpreted. PTQ therefore postulates a level, intensional logic, between surface forms and their interpretations, thus contravening (6a). However, it clearly respects (6b) in its foundational presentation of rule-by-rule translation. Thus, (6b) does not logically depend on (6a).

Similarly, one can imagine systems that respect (6a) by not positing any intervening level between surface structure and semantics, but which contravene (6b) through appeal to non-compositionality. Indeed, such proposals have been made; relevant examples and discussion can be found in Partee (1984) and Janssen (1997:437–441). It is not clear how interesting the proposals are, in light of theorems about the syntactic and semantic side of compositionality (Janssen 1986, Zadrozny 1994), which together show that 'without constraints on syntax and semantics, there are no counterexamples to compositionality' (Janssen 1997:456–457). In other words, these proposals are arguably more relevant to the question of proper constraints on syntax and semantics than to the question of compositionality. However, the proposals do show that (6a) does not depend on (6b) (although, according to such proposals, surface syntax is not the sole determinant of interpretation). Therefore, the two parts of DC are independent.

Jacobson's conflation of (6a) and (6b) is understandable, though, when seen in light of the broader context of her work on direct compositionality. This work is part of a research program, set out in detail in Jacobson (1999), that seeks to argue for a directly compositional (in both senses of (6)), variable-free semantics in opposition to semantics in the tradition of Logical Form, for which Heim and Kratzer (1998) is a key touchstone, both for Jacobson and in the semantics literature more generally. As it happens, Heim and Kratzer (1998) deny both parts of (6) in the sense that Jacobson has in mind, although they do not deny

(6b) under a different construal (i.e., their semantics *is* compositional). Their denial of (6a) is obvious, since their semantics interprets Logical Forms, which are not surface structures. Their denial of (6b) is much more subtle, though.

Heim and Kratzer (1998:49) propose the following principle for semantic interpretation:

(8) *Principle of Interpretability*
All nodes in a phrase structure tree must be in the domain of the interpretation function $[\![\ ]\!]$.

Given that the phrase structure tree in question must be well-formed according to some syntax, it initially seems that Heim and Kratzer's system does support (6b). However, it is clear that what Jacobson means by 'syntax' is whatever proves well-formedness of the surface strings of the language. But this is not the kind of syntax that yields the phrase structure trees of interest in (8): those are LF trees and LF is not surface structure (the yields of LF trees are not the surface strings of the language, since movement operations can occur at LF).

Here is another way to think about the difference between the position of Jacobson and that of Heim and Kratzer. Jacobson assumes that semantic interpretability and syntactic well-formedness are mutually entailing: if a string is syntactically well-formed, it receives an interpretation, and if a string receives an interpretation, it is syntactically well-formed. Heim and Kratzer explicitly deny this:

> In sum, we are adopting a view of the grammar as a whole in which syntax and semantics are independent modules. Each imposes its own constraints on the grammatical structures of the language, and we expect there to be structures that are interpretable though syntactically illegitimate, as well as structures that are syntactically correct but uninterpretable. (Heim and Kratzer 1998:49)

Thus, Jacobson assumes an extremely tight relationship between syntax and semantics, and Heim and Kratzer assume a looser relationship, although for them interpretation still depends compositionally on the level of Logical Form.

The question now is how the architecture of Lexical Functional Grammar fits into this picture, assuming that Glue Semantics is providing the semantic theory. I will call this combination 'LFG-Glue'. With respect to the question of the relationship between model-theoretic interpretation and syntactic well-formedness, LFG-Glue provides a potentially interesting intermediate position between the Jacobson and Heim & Kratzer positions. The linear logic proofs in Glue Semantics depend on the syntax to instantiate semantic structure nodes in the premises for the proofs. If the syntactic input to the proofs is ill-formed, the proofs will consequently fail due to improper instantiation of premises. Thus, successful semantic interpretation depends on syntactic well-formedness. Under certain circumstances, syntactically ill-formed structures may have informative *partial* interpretations (Asudeh 2004:321–334). However, these structures are not *fully* interpretable, contra the picture sketched in the Heim and Kratzer quote above, which specifically countenances 'structures that are interpretable though syntactically illegitimate'. In sum, (6b) is upheld in Glue Semantics: complete interpretation is a function of syntactic well-formedness, where the syntax in question is the syntax of the surface strings in LFG-Glue.

Jacobson's position further entails that if a structure is syntactically well-formed, it is interpretable. In general, this is also true in Glue Semantics, because lexical items specify their semantic types and there is, as in most theories, a strong correlation between semantic type and syntactic category. The upshot is that as long as the lexical items are assigned motivated meanings, syntactically well-formed structures will be interpretable. However, the syntax of composition in Glue Semantics is divorced from the syntax of string formation, unlike in Categorial Grammar, the framework that Jacobson assumes (see Jacobson 1999). Therefore, if one assigns a type to a lexical item in LFG-Glue that is not reflected by the item's syntactic information, there could be a syntactically well-formed structure that is not fully saturated (i.e., not of type t). Such cases do not happen in practice, but are possible in principle.

For example, suppose we had a syntax with the following annotated phrase structure rule for c-structure:

(9) $$\mathrm{S} \longrightarrow \underset{(\uparrow \textsc{subj}) = \downarrow}{\mathrm{N}} \quad \underset{\uparrow = \downarrow}{\mathrm{V}}$$

Now suppose we have a verb in our lexicon, e.g. *floobles*, which has the semantically transitive type $\langle e, \langle e, t\rangle\rangle$, but which does not syntactically select for an object. The verb would have the usual syntactic category of V. Assuming appropriate category N lexical entries, our syntax would then derive sentences like *John floobles*, which is *syntactically* well-formed, but which is uninterpretable, according to the type specification of *floobles*.

LFG-Glue therefore upholds (6b) in Jacobson's strict sense, because successful interpretation does entail well-formedness of the string-yielding syntax. As far as interpretability entailing well-formedness, LFG-Glue and Categorial Grammar are thus in agreement, contra LF semantics. However, LFG-Glue is like the latter, contra Categorial Grammar, in denying that syntactic well-formedness *in principle* entails interpretability (although it does in practice). This difference between LFG-Glue and Categorial Grammar is directly traceable to the logic of composition. Categorial Grammar posits that the syntax of string analysis (parsing/generation) and the syntax of composition are essentially the same. The non-commutativity of syntax is thus passed on to semantics. However, it is questionable whether the logic of semantic composition should itself be non-commutative, because the fundamental operation in compositional semantics, functional application, is commutative (Asudeh 2004:76–77). Glue Semantics, through its use of the commutative linear logic for semantic composition, separates the non-commutativity of string analysis (provided by an LFG syntax in LFG-Glue) from the commutativity of semantic composition. This is, at heart, what gives LFG-Glue a theoretical position with respect to the modularity of syntax and semantics that is intermediate between syntactic well-formedness and semantic interpretability being mutually entailing (Jacobson's position) or fully independent (Heim and Kratzer's position).

The LFG architecture thus respects (6b), the second part of Direct Compositionality, even on a strict interpretation. However, it seems that the LFG architecture does not support (6a), because there is a level of semantic structure immediately before interpretation. In the next section, I show that semantic structure is dispensable and that LFG therefore upholds (6a), despite initial appearances.

17.4 Directly Compositional LFG

Jacobson (2002:603ff.) uses quantifier scope ambiguity to illuminate the hypothesis of direct compositionality. In this section, I will first introduce an analysis of scope ambiguity in LFG-Glue; this will help put the subsequent discussion of direct compositionality in LFG-Glue on an even footing with Jacobson's presentation, which considers other frameworks.

The example that Jacobson (2002:603) uses is:

(10) Some man read every book.

Appropriate simplified lexical entries for this sentence in LFG-Glue are shown in (11).

(11) *some* D $(\uparrow \text{PRED}) = \text{`some'}$

$$\begin{array}{l} some' : \\ [((\text{SPEC}\uparrow)_\sigma\ \text{VAR}) \multimap ((\text{SPEC}\uparrow)_\sigma\ \text{RESTR})] \multimap \\ \quad [((\text{SPEC}\uparrow)_\sigma \multimap X) \multimap X] \end{array}$$

man	N	$(\uparrow \textsc{pred}) = \text{'man'}$
		$\mathit{man}' : (\uparrow_\sigma \textsc{var}) \multimap (\uparrow_\sigma \textsc{restr})$
read	V	$(\uparrow \textsc{pred}) = \text{'read}\langle(\uparrow \textsc{subj}), (\uparrow \textsc{obj})\rangle\text{'}$
		$(\uparrow \textsc{tense}) = \textsc{past}$
		$\mathit{read}' : (\uparrow \textsc{obj})_\sigma \multimap (\uparrow \textsc{subj})_\sigma \multimap \uparrow_\sigma$
every	D	$(\uparrow \textsc{pred}) = \text{'every'}$
		$\mathit{every}' :$ $[((\textsc{spec}\uparrow)_\sigma\ \textsc{var}) \multimap ((\textsc{spec}\uparrow)_\sigma\ \textsc{restr})] \multimap$ $[((\textsc{spec}\uparrow)_\sigma \multimap Y) \multimap Y]$
book	N	$(\uparrow \textsc{pred}) = \text{'book'}$
		$\mathit{book}' : (\uparrow_\sigma \textsc{var}) \multimap (\uparrow_\sigma \textsc{restr})$

The last part of each of these lexical entries is a Glue Semantics *meaning constructor*. Each meaning constructor has the following form:

(12) $\mathcal{M} : G$

$\mathcal{M}$ is a term from the meaning language — the language that is model-theoretically interpreted; this is a fragment of a logic that supports the lambda calculus. G is a term from the glue language, a fragment of linear logic. The meaning constructors for an expression form the premise set for a linear logic proof of the expression's semantics. Terms in $\mathcal{M}$ and G are systematically related in the proof by the Curry-Howard Isomorphism (CHI) between formulas and types (Curry and Feys 1958, Howard 1980). The only aspects of the CHI that will be relevant in what follows are the correspondences between functional application in the meaning language and implication elimination ($\multimap_{\mathcal{E}}$; *modus ponens*) in the linear logic glue language and between abstraction and implication introduction ($\multimap_{\mathcal{I}}$).[3]

These lexical entries form the terminals in a constituent structure parse of the string in (10). The constituent structure rules are annotated phrase structure rules (Kaplan and Bresnan 1982). The annotations are typically functional structure equations that are defined in terms of two variables over c-structure nodes — $*$ for the current node and $\widehat{*}$ for the current node's mother[4] — and the ϕ projection function from c-structures to f-structures. The annotation $\phi(*)$ therefore means the f-structure correspondent of the current node and the annotation $\phi(\widehat{*})$

[3] For further details on the CHI and Glue Semantics, see Dalrymple et al. (1999a), Crouch and van Genabith (2000), Dalrymple (2001), and Asudeh (2004, 2005).

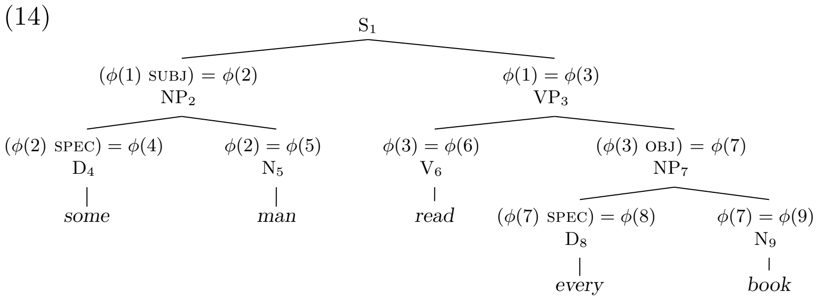
[4] These notions are ultimately defined in terms of N, the set of c-structure nodes, and the mother function on nodes, M: N → N; see Kaplan (1987) for a concise overview.

means the f-structure correspondent of the current node's mother. $\phi(*)$ and $\phi(\widehat{*})$ are conventionally written as $\downarrow$ and $\uparrow$ respectively.

For expository purposes, I assume the following very simple set of c-structure rules for the analysis of (10):

$$
\text{(13)} \quad
\begin{array}{lllll}
\text{S} & \longrightarrow & \underset{(\uparrow \textsc{subj}) = \downarrow}{\text{NP}} & \underset{\uparrow = \downarrow}{\text{VP}} \\[2ex]
\text{NP} & \longrightarrow & \underset{(\uparrow \textsc{spec}) = \downarrow}{\text{D}} & \underset{\uparrow = \downarrow}{\text{N}} \\[2ex]
\text{VP} & \longrightarrow & \underset{\uparrow = \downarrow}{\text{V}} & \underset{(\uparrow \textsc{obj}) = \downarrow}{\text{NP}}
\end{array}
$$

The lexical entries in (11) and the c-structure rules in (13) give the following c-structure for (10):

(14)


The nodes in the c-structure have been assigned unique numbers as names (see Kaplan 1987); these node names are used to instantiate the f-structure variables in node annotations. The terminals are the lexical entries from (11), including all the equational information, but have been labeled in an abbreviated form.

Solving the f-structure equations in the lexical entries and c-structure, we get the following functional structure:

$$
\text{(15)} \quad
r\left[\begin{array}{ll}
\textsc{pred} & \text{'read}\langle(r\ \ \textsc{subj}),(r\ \ \textsc{obj})\rangle\text{'} \\
\textsc{subj} & m\left[\begin{array}{ll} \textsc{pred} & \text{'man'} \\ \textsc{spec} & \left[\begin{array}{ll}\textsc{pred} & \text{'some'}\end{array}\right] \end{array}\right] \\
\textsc{obj} & b\left[\begin{array}{ll} \textsc{pred} & \text{'book'} \\ \textsc{spec} & \left[\begin{array}{ll}\textsc{pred} & \text{'every'}\end{array}\right] \end{array}\right] \\
\textsc{tense} & \textsc{past}
\end{array}\right]
$$

I have followed the convention of labeling f-structures mnemonically based on their PRED value. In this case, this means that $r = \phi(1) = \phi(3) = \phi(6)$, $m = \phi(2) = \phi(5)$, and $b = \phi(7) = \phi(9)$.

The σ projection function from f-structure to s-structure maps from nodes in (15) to the s-structure in (16). Note that I have labeled $(m_\sigma\text{VAR})$ as *v1*, etc.; these abbreviations will be useful below. I have also followed the convention of writing $\sigma(x)$ as x_σ.

(16) $r_\sigma\Big[\ \Big] \qquad m_\sigma\begin{bmatrix}\text{VAR} & v1\big[\ \big] \\ \text{RESTR} & r1\big[\ \big]\end{bmatrix} \qquad b_\sigma\begin{bmatrix}\text{VAR} & v2\big[\ \big] \\ \text{RESTR} & r2\big[\ \big]\end{bmatrix}$

Notice that semantic structure is both very sparse and unconnected. It is unconnected because no notion of a semantic structure head path has been defined on a par with the f-structure head paths defined by $\uparrow = \downarrow$ equations. Such a semantic notion of head could easily be constructed through the specification $\uparrow_\sigma = \downarrow_\sigma$, but a theoretical need for this has yet to be identified.

The nodes of s-structure fill in variables in lexically contributed meaning constructors, yielding the set of premises in (17) for the linear logic proof of the semantics, based on the contributions in (11).

(17)
1. $\mathit{some}' : (v1 \multimap r1) \multimap ((m \multimap X) \multimap X)$
2. $\mathit{man}' : v1 \multimap r1$
3. $\mathit{read}' : b \multimap m \multimap r$
4. $\mathit{every}' : (v2 \multimap r2) \multimap ((b \multimap Y) \multimap Y)$
5. $\mathit{book}' : v2 \multimap r2$

Based on these premises, we can construct two valid linear logic proofs. Both proofs share the same initial sub-proof, shown in (20). The proofs then diverge, depending on which quantifier is scoped first. The proof in figure 2 provides the surface scope reading and the proof in figure 3 provides the inverse scope reading. For presentational purposes, I have left implicit in figures 2 and 3 the sub-proofs that show the composition of the quantificational determiners with their nominal restrictions; these are presented separately in (18) and (19).

(18) $$\frac{\mathit{some}' : (v1 \multimap r1) \multimap ((m \multimap X) \multimap X) \qquad \mathit{man}' : v1 \multimap r1}{\mathit{some}'(\mathit{man}') : ((m \multimap X) \multimap X)} \multimap_{\mathcal{E}}$$

(19) $$\frac{\mathit{every}' : (v2 \multimap r2) \multimap ((b \multimap Y) \multimap Y) \qquad \mathit{book}' : v2 \multimap r2}{\mathit{every}'(\mathit{book}') : ((b \multimap Y) \multimap Y)} \multimap_{\mathcal{E}}$$

(20) $$\frac{\dfrac{\mathit{read}' : b \multimap m \multimap r \quad [y : b]^1}{\mathit{read}'(y) : m \multimap r} \multimap_{\mathcal{E}} \qquad [x : m]^2}{\mathit{read}'(y)(x) : r} \multimap_{\mathcal{E}}$$

In Glue Semantics, the two alternative scopings are thus completely based on alternative linear logic derivations on the same set of premises.

$$\dfrac{\begin{array}{c}\vdots\\ some'(man'):\\ ((m \multimap X) \multimap X)\end{array} \qquad \dfrac{\dfrac{\begin{array}{c}\vdots\\ every'(book'):\\ ((b \multimap Y) \multimap Y)\end{array} \qquad \dfrac{\begin{array}{c}\vdots\\ read'(y)(x) : r\end{array}}{\lambda y.read'(y)(x) : b \multimap r}\multimap_{\mathcal{I},1}}{every'(book')(\lambda y.read'(y)(x)) : r}\multimap_{\mathcal{E}},\ [r/Y]}{\lambda x.every'(book')(\lambda y.read'(y)(x)) : m \multimap r}\multimap_{\mathcal{I},2}}{some'(man')(\lambda x.every'(book')(\lambda y.read'(y)(x))) : r}\multimap_{\mathcal{E}},\ [r/X]$$

FIGURE 2 Surface scope proof

$$\dfrac{\begin{array}{c}\vdots\\ every'(book'):\\ ((b \multimap Y) \multimap Y)\end{array} \qquad \dfrac{\dfrac{\begin{array}{c}\vdots\\ some'(man'):\\ ((m \multimap X) \multimap X)\end{array} \qquad \dfrac{\begin{array}{c}\vdots\\ read'(y)(x) : r\end{array}}{\lambda x.read'(y)(x) : m \multimap r}\multimap_{\mathcal{I},2}}{some'(man')(\lambda x.read'(y)(x)) : r}\multimap_{\mathcal{E}},\ [r/X]}{\lambda y.some'(man')(\lambda x.read'(y)(x)) : b \multimap r}\multimap_{\mathcal{I},1}}{every'(book')(\lambda y.some'(man')(\lambda x.read'(y)(x))) : r}\multimap_{\mathcal{E}},\ [r/Y]$$

FIGURE 3 Inverse scope proof

No syntactic Quantifier Raising ambiguity is assumed — there is a single c-structure and f-structure for (10) — and there is no type shifting. The Glue Semantics approach to scope ambiguity is therefore distinct from both Logical Form and Categorial Grammar approaches.

With this exposition of scope in Glue Semantics in hand, we can now return to the question of direct compositionality. The use of semantic structure as an input to the linear logic derivation of the semantics of (10) is an apparent rejection of part (6a) of the hypothesis, which postulates that there is no intermediate level between surface structure and model-theoretic interpretation. However, as discussed in section 17.2, in LFG the intervening levels between form and meaning are dispensable, via composition of the correspondence functions. The composition in question is the following:

(21) $\mathit{\Gamma} = \pi \circ \phi \circ \sigma \circ \psi$

Recalling the discussion of ambiguity in section 17.2, this function is short for a family of functions. Thus, each function in (21) admits several instances, and there may therefore be multiple $\mathit{\Gamma}$ functions that map the string to different meanings.

In this case there is an ambiguity in meaning. The two instances of $\mathit{\Gamma}$ functions are shown here:

(22) $\mathit{\Gamma}^{1}(\textit{some man read every book})$
$= \textit{some}'(\textit{man}')(\lambda x.\textit{every}'(\textit{book}')(\lambda y.\textit{read}'(y)(x)))$

(23) $\mathit{\Gamma}^{2}(\textit{some man read every book})$
$= \textit{every}'(\textit{book}')(\lambda y.\textit{some}'(\textit{man}')(\lambda x.\textit{read}'(y)(x)))$

The ambiguity arises only at the last point, in the ψ mapping (characterized by linear logic proofs) from semantic structure to meaning. The ambiguity derives from multiple proofs from a single set of premises.

The fact that alternative scopings arise from the same premise set means that Glue Semantics espouses a notion of purely semantic ambiguity. In other words, Glue Semantics rejects the conception of compositionality in which there is a functional relation between syntactic structures and meanings (as in, for example, the classic Montague Semantics of Montague 1970, 1973). In functional compositionality, for each distinct meaning there is at most one distinct syntactic structure: there are no one-to-many mappings from syntax to meanings. The result of functional compositionality is that every semantic ambiguity forces a syntactic ambiguity; in other words, there is no pure notion of semantic ambiguity. Glue Semantics instead espouses a relational view of compositionality: there can be a one-to-many mapping from syntax to semantics. This preserves semantic ambiguity without forcing syntactic ambiguity. However, each distinct interpretation corresponds to a distinct proof. There is therefore a purely functional mapping from the syntax of semantic composition (proofs) to model-theoretic interpretations: each proof maps to a single interpretation.

Although LFG-Glue brings relational compositionality to the fore, note that this view of compositionality is also a feature of other modern approaches to the syntax–semantics interface. In LF semantics, a string can be assigned multiple logical forms. Thus, although there is a functional mapping from logical forms to model-theoretic interpretation (each LF has a single interpretation), the mapping from a *string* to its interpretation(s) is relational. Similarly, consider Categorial Grammar. There is a functional mapping from each categorial proof of syntax to model-theoretic interpretation. However, unary operations like type-shifting mean that the set of lexical items that parse the string can correspond to multiple syntactic analyses. Indeed, quantifier scope ambiguity is precisely a case that leads to multiple logical forms in LF semantics and to type-shifting in Categorial Grammar. If we are considering the mapping from a string to its interpretations, the real question is thus not whether compositionality is relational — the existence of ambiguity dictates that at some point there has to be a one-to-many mapping from a string to its interpretations. The real question is: What is the point identified by the grammatical architecture at which the mapping from syntax to semantics becomes purely functional? In Glue Semantics, this point happens very late in the pipeline from form to meaning, at the proof level. In LFG-Glue terms, it happens in the ψ mapping from s-structure to model-theoretic interpretation. In LF semantics it also happens late, at the point of mapping from logical forms to model-theoretic interpretation. In Categorial Grammar, it happens early, in the syntactic analysis.

Let me unpack in a little more detail how the composition of projection functions works, since the pieces are already in place. The initial π projection function from the string in (10) to the c-structure in (14) is characterized by the annotated phrase structure rules in (13). The ϕ projection function maps the c-structure in (14) to the f-structure in (15). The σ projection function maps the f-structure in (15) to the s-structure in (16). Lastly, the ψ function maps from the s-structure to model-theoretic meaning. The ψ function is characterized by a fragment of linear logic. The Curry-Howard Isomorphism relates operations in the linear logic to operations in the related meaning language. Interpretation of the meaning language yields the model-theoretic meaning. In sum, although there are many different levels between the string and its meaning in LFG-Glue, including a level between the syntax and semantics (s-structure), these levels are all 'formally and mathematically, and also computationally, eliminable' (Kaplan 1987:363). Thus, although LFG postulates a level of semantic structure between syntax and model-theoretic meaning, the theory nonetheless upholds the first part of the hypothesis of direct compositionality, because the level in question is eliminable. Nevertheless, as Kaplan (1987:363) also notes, '[T]he structures related by multiple correspondences might be descriptively or linguistically motivated levels of representation, justified by sound theoretical argumentation.' For example, in Asudeh (2005), I argue that a crucial distinction between pronouns and relational nouns can be explained by a theoretically motivated distinction at s-structure.

Lastly, let us consider in what precise sense LFG-Glue meets direct compositionality. Jacobson (2002:603) characterizes a grammar that satisfies Strong Direct Compositionality as one that uses context-free phrase structure rules or the equivalent for its syntax. More generally, trees or other structured objects in the relevant grammars should constitute proofs of string well-formedness, but should not be directly referred to by the grammar. The phrase structure component of LFG satisfies this conception, since LFG has a context-free base (Kaplan 1987). In fact, Roach (1985) shows that this context-free base can in certain circumstances be further reduced to a finite-state base (Kaplan 1987:364). Thus, LFG-Glue not only satisfies direct compositionality, it satisfies the strongest version, Strong Direct Compositionality.

17.5 The Proper Use of Intermediate Structures

The grammatical architecture of LFG-Glue in principle upholds direct compositionality, in both senses of (6); however, the architecture also permits the direct mapping from surface structure to models to be taken apart. This allows relevant linguistic generalizations to be made straightforwardly about points in the mapping (intermediate structures in the projection architecture) that are hidden in the direct mapping.

This sort of architecture is similar in principle to that of Montague's PTQ, Jacobson's exemplar of direct compositionality. PTQ's intermediate representation is intensional logic, but this is merely an eliminable intermediate step to interpretation (Gamut 1991), as demonstrated by Montague (1970), which provides a model-theoretic interpretation for a fragment of English without using IL. Janssen (1997) provides a useful discussion of this:

> Since meanings are generally formalized as model-theoretic entities, such as truth values, sets of sets, etc., functions have to be spec-

> ified which operate on such meanings ... Such descriptions are not easy to understand, nor convenient to work with. Therefore almost always a logical language is used to represent meanings and operations on meanings ... So in practice associating meanings with natural language amounts to translating sentences into logical formulas. (Janssen 1997:434)

Janssen (1997:434) provides a telling example of a complicated and hard to understand operation from Montague (1970) that translates into the simple intensional logic formula $\hat{}\lambda t \lambda u[t = u]$. In other words, Montague's PTQ is a perfect example of what Kaplan is pointing out in the quote on page 366: although intermediate levels can be eliminated, this may be lead to a linguistic theory that is harder to understand and that, as a result, is less revealing than an equivalent theory that uses the intermediate levels appropriately.

Although the architecture of LFG-Glue can uphold direct compositionality, other work in Glue Semantics has postulated that proof-theoretic properties of linear logic proofs can explain linguistic phenomena, such as grammatical violations of the Coordinate Structure Constraint (Asudeh and Crouch 2002a) and scope parallelism in ellipsis (Asudeh and Crouch 2002b). This amounts to a denial of direct compositionality, because the linear logic proofs are themselves considered an aspect of interpretation in this work, rather than merely an eliminable step to model-theoretic interpretation.

With respect to this use of linear logic proofs, the following continuation of the passage by Janssen is pertinent:

> Working in accordance with compositionality of meaning puts a heavy restriction on the translations into logic, because the goal of the translations is to assign meanings. The logical representations are just a tool to reach this goal. The representations are not meanings themselves, and should not be confused with them. This means for instance, that two logically equivalent representations are equally good as representation [sic.] of the associated meaning. (Janssen 1997:434)

Linear logic proofs have strong identity criteria that allow them to be properly individuated, thus avoiding the trap of false distinctions that Janssen identifies here. For the fragment of linear logic used in Glue Semantics, there are two convergent ways of stating the identity criteria. One is based on the Curry-Howard Isomorphism and the other on proof normalization/cut elimination (Prawitz 1965); see Asudeh and Crouch (2002b) for some discussion with respect to linguistic generalizations and Crouch and van Genabith (2000) for detailed theoretical discussion.

17.6 Conclusion

In this paper, I have built on Ron Kaplan's work on LFG's parallel projection architecture and presented a synthesis of subsequent proposals for the architecture. I considered the question of whether the architecture satisfies the hypothesis of direct compositionality, discussed in recent work by Jacobson, in the context of LFG with Glue Semantics as its semantic theory. I identified two components of the hypothesis and argued that LFG-Glue satisfies both components. Lastly, I argued that the grammatical architecture of LFG-Glue sheds new light on the hypothesis of direct compositionality: intermediate levels of representation can be appropriate and useful if well-understood, a point long anticipated by Kaplan (1987).

Acknowledgments

Many thanks to Mary Dalrymple, Ida Toivonen, and an anonymous reviewer for their invaluable comments. Thanks also to Mary for her very patient editorial and formatting work. I am especially indebted to Chris Potts for his thoughtful feedback on the paper in general, and section 17.4 in particular. Any remaining errors are my own.

References

Asudeh, Ash and Richard Crouch. 2002a. Coordination and parallelism in Glue Semantics: Integrating discourse cohesion and the element constraint. In M. Butt and T. H. King, eds., *Proceedings of the International Lexical-Functional Grammar Conference 2002 (LFG'02)*, pages 19–39. Athens, Greece: CSLI Online Publications.

Asudeh, Ash and Richard Crouch. 2002b. Derivational parallelism and ellipsis parallelism. In L. Mikkelsen and C. Potts, eds., *Proceedings of the 21st West Coast Conference on Formal Linguistics (WCCFL XXI)*, pages 1–14. Somerville, MA: Cascadilla Press.

Asudeh, Ash. 2004. *Resumption as Resource Management*. Ph.D. thesis, Stanford University.

Asudeh, Ash. 2005. Relational nouns, pronouns, and resumption. *Linguistics and Philosophy* 28:375–446.

Butt, Miriam, María-Eugenia Niño, and Frédérique Segond. 1996. Multilingual processing of auxiliaries in LFG. In *Natural Language Processing and Speech Technology: Results of the 3rd KONVENS Conference, Bielefeld*, pages 111–122. Berlin, Germany: Mouton de Gruyter.

Butt, Miriam, Mary Dalrymple, and Anette Frank. 1997. An architecture for linking theory in LFG. In M. Butt and T. H. King, eds., *Proceedings of the International Lexical-Functional Grammar Conference 1997 (LFG'97)*. San Diego, CA: CSLI Online Publications.

Butt, Miriam and Tracy H. King. 1998. Interfacing phonology with LFG. In M. Butt and T. H. King, eds., *Proceedings of the International Lexical-Functional Grammar Conference 1998 (LFG'98)*. Brisbane, Australia: CSLI Online Publications.

Butt, Miriam, Tracy H. King, María-Eugenia Niño, and Frédérique Segond. 1999. *A Grammar Writer's Cookbook*. Stanford, CA: CSLI Publications.

Butt, Miriam and Tracy H. King. 2000. Null elements in discourse structure. In K. V. Subbarao, ed., *Papers from the NULLS Seminar*. Delhi, India: Motilal Banarsidass. To appear.

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

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

Crouch, Richard and Josef van Genabith. 2000. Linear logic for linguists. Ms., PARC and Dublin City University.
`http://www2.parc.com/istl/members/crouch/`; checked 24/04/2006.

Curry, Haskell B. and Robert Feys. 1958. *Combinatory Logic*, vol. 1. Amsterdam, The Netherlands: North-Holland.

Dalrymple, Mary. 1993. *The Syntax of Anaphoric Binding*. Stanford, CA: CSLI Publications.

Dalrymple, Mary, Ronald M. Kaplan, John T. Maxwell, III, and Annie Zaenen, eds. 1995. *Formal Issues in Lexical-Functional Grammar*. Stanford, CA: CSLI Publications.

Dalrymple, Mary, ed. 1999. *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*. Cambridge, MA: The MIT Press.

Dalrymple, Mary, Vineet Gupta, John Lamping, and Vijay Saraswat. 1999a. Relating resource-based semantics to categorial semantics. In *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*, pages 261–280. Cambridge, MA: The MIT Press.

Dalrymple, Mary, John Lamping, Fernando Pereira, and Vijay Saraswat. 1999b. Overview and introduction. In *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*, pages 1–38. Cambridge, MA: The MIT Press.

Dalrymple, Mary. 2001. *Lexical Functional Grammar*. San Diego, CA: Academic Press.

de Groote, Philippe, ed. 1995. *The Curry-Howard Isomorphism*, vol. 8 of *Cahiers du Centre de Logique*. Louvain-la-Neuve, Belgium: Academia.

Frank, Anette and Annie Zaenen. 2002. Tense in LFG: Syntax and morphology. In H. Kamp and U. Reyle, eds., *How do we say WHEN it happens: Contributions to the theory of temporal reference in natural language*, pages 17–51. Tübingen, Germany: Niemeyer.

Gamut, L.T.F. 1991. *Intensional Logic and Logical Grammar*, vol. 2 of *Logic, Language, and Meaning*. Chicago, IL: University of Chicago Press.

Girard, Jean-Yves. 1987. Linear logic. *Theoretical Computer Science* 50:1–102.

Halvorsen, Per-Kristian. 1983. Semantics for Lexical-Functional Grammar. *Linguistic Inquiry* 14:567–615.

Halvorsen, Per-Kristian and Ronald M. Kaplan. 1988. Projections and semantic description in Lexical-Functional Grammar. In *Proceedings of the International Conference on Fifth Generation Computer Systems (FGCS'88)*, pages 1116–1122. Tokyo, Japan. Reprinted in Dalrymple et al. (1995:279–292).

Heim, Irene and Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Oxford, United Kingdom: Blackwell.

Hodges, Wilfrid. 1998. Compositionality is not the problem. *Logic and Logical Philosophy* 6:7–33.

Hodges, Wilfrid. 2001. Formal features of compositionality. *Journal of Logic, Language, and Information* 10:7–28.

Howard, William A. 1980. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, eds., *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 479–490. London, United Kingdom: Academic Press. Circulated in unpublished form from 1969. Reprinted in de Groote (1995:15–26).

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

Jacobson, Pauline. 2002. The (dis)organization of the grammar: 25 years. *Linguistics and Philosophy* 25:601–626.

Jacobson, Pauline. 2004. Direct Compositionality: Is there any reason why not? MS., Brown University. Presented at the University of Michigan Workshop on Linguistics and Philosophy, Ann Arbor, MI.

Jacobson, Pauline. 2005. Direct Compositionality and variable-free semantics: The case of 'Principle B' effects. In C. Barker and P. Jacobson, eds., *Direct Compositionality*. Oxford, United Kingdom: Oxford University Press. To appear.

Janssen, Theo M. V. 1986. *Foundations and Applications of Montague Grammar. Part I: Philosophy, Framework, Computer Science*. No. 19 in CWI Tracts. Amsterdam, The Netherlands: Centre for Mathematics and Computer Science.

Janssen, Theo M. V. 1997. Compositionality. In J. van Benthem and A. ter Meulen, eds., *Handbook of Logic and Language*, pages 417–473. Cambridge, MA: The MIT Press. Co-published with Elsevier Science B.V., Amsterdam, The Netherlands.

Kaplan, Ronald M. and Joan Bresnan. 1982. Lexical-Functional Grammar: A formal system for grammatical representation. In J. Bresnan, ed., *The Mental Representation of Grammatical Relations*, pages 173–281. Cambridge, MA: The MIT Press.

Kaplan, Ronald M. 1987. Three seductions of computational psycholinguistics. In P. Whitelock, M. M. Wood, H. L. Somers, R. Johnson, and P. Bennett, eds., *Linguistic Theory and Computer Applications*, pages 149–181. London, United Kingdom: Academic Press. Reprinted in Dalrymple et al. (1995:339–367).

Kaplan, Ronald M. 1989. The formal architecture of Lexical-Functional Grammar. In C.-R. Huang and K.-J. Chen, eds., *Proceedings of the ROC Computational Linguistics Workshops II (ROCLING II)*, pages 3–18. Taipei, ROC. Reprinted in Dalrymple et al. (1995:7–27).

May, Robert. 1977. *The Grammar of Quantification*. Ph.D. thesis, MIT.

Montague, Richard. 1970. English as a formal language. In B. Visentini et al., eds., *Linguaggi nella Societá e nella Tecnica*, pages 189–224. Milan, Italy: Edizioni di Communità. Reprinted in Montague (1974:188–221).

Montague, Richard. 1973. The proper treatment of quantification in ordinary English. In J. Hintikka, J. Moravcsik, and P. Suppes, eds., *Approaches to Language*, pages 221–242. Dordrecht, The Netherlands: Reidel. Reprinted in Montague (1974:247–270).

Montague, Richard. 1974. *Formal Philosophy: Selected Papers of Richard Montague*. New Haven, CT: Yale University Press. Edited and with an introduction by R. H. Thomason.

Partee, Barbara H. 1984. Compositionality. In F. Landman and F. Veltman, eds., *Varieties of Formal Semantics*, pages 281–311. Dordrecht, The Netherlands: Foris. Reprinted in Partee (2004:153–181).

Partee, Barbara H., Alice ter Meulen, and Robert E. Wall. 1993. *Mathematical Methods in Linguistics*. Dordrecht, The Netherlands: Kluwer Academic Publishers.

Partee, Barbara H. 2004. *Compositionality in Formal Semantics: Selected Papers of Barbara H. Partee*. Oxford, United Kingdom: Blackwell.

Pollard, Carl and Ivan A. Sag. 1987. *Information-Based Syntax and Semantics*. Stanford, CA: CSLI Publications.

Pollard, Carl and Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. Chicago, IL and Stanford, CA: The University of Chicago Press and CSLI Publications.

Prawitz, Dag. 1965. *Natural Deduction: A Proof-Theoretical Study*. Stockholm, Sweden: Almquist and Wiksell.

Roach, Kelly. 1985. The mathematics of LFG. Ms., Xerox Palo Alto Research Center.

Tarski, Alfred. 1983 [1935]. The concept of truth in formalized languages. In *Logic, Semantics, Metamathematics*, pages 152–278. Indianapolis, IN: Hackett, 2nd edn. Translated by J. H. Woodger. Edited by J. Corcoran.

Zadrozny, Wlodek. 1994. From compositional semantics to systematic semantics. *Linguistics and Philosophy* 17:329–342.