Lexical-Functional Grammar*

Ash Asudeh & Ida Toivonen

August 10, 2009 [19:17]
CORRECTED FINAL DRAFT

To appear in Bernd Heine and Heiko Narrog, eds., *The Oxford Handbook of Linguistic Analysis*. Oxford: Oxford University Press.

Contents

*This work was supported by the Social Sciences and Humanities Research Council of Canada, Standard Research Grant 410-2006-1650. We would like to thank Mary Dalrymple, Bernd Heine, Daniela Isac, Helge Lødrup, Kumiko Murasugi, Heiko Narrog, Kenji Oda, Chris Potts, and Nigel Vincent for helpful comments and suggestions. We would also like to thank Daniela Isac and Cristina Moldoveanu for checking the Romanian data. Any remaining errors are our own.

Abstract

Lexical-Functional Grammar (LFG) is a lexicalist, declarative (non-transformational), constraint-based theory of generative grammar. LFG has a detailed, industrial-strength computational implementation. The theory has also proven useful for descriptive/documentary linguistics. The grammatical architecture of LFG, sometimes called the 'Correspondence Architecture', posits that different kinds of linguistic information are modelled by distinct, simultaneously present grammatical structures, each having its own formal representation. These separate structures are formally related by correspondence functions. The different grammatical structures are subject to separate principles and formal descriptions and have distinct primitives. The two core syntactic structures are constituent structure and functional structure, and they are the central focus of this chapter. Other grammatical structures that have been proposed concern argument structure, information structure, semantics and the syntax–semantics interface, prosody and the syntax–phonology interface, and the morphology–syntax interface.

Keywords: constraint-based syntax, morphosyntax, computational linguistics, constraint, syntactic feature, feature structure, Correspondence Architecture, correspondence function, c(onstituent)-structure, f(unctional)-structure, structural description, Lexical Integrity, grammatical function, functional control, functional uncertainty

1 Introduction

Lexical-Functional Grammar (LFG) was first developed in the 1970's by Joan Bresnan, a linguist at MIT, and Ron Kaplan, a psychologist at Harvard. Bresnan and Kaplan were concerned with the related issues of psychological plausibility and computational tractability. They wanted to create a theory that could form the basis of a realistic model for linguistic learnability and language processing. Since its foundation, the theory has been applied to numerous new areas, undergoing some modifications in the process, and has incorporated insights from a variety of morphological, syntactic, and semantic theories. However, the basic tenets of the theory and the formal framework have remained remarkably stable. For more on the history of LFG, see Kaplan (1987), Dalrymple et al. (1995: ix–5) and Dalrymple (2001: 1–5).

LFG is a theory of generative grammar, in the sense of Chomsky (1957, 1965). The goal is to explain the native speaker's knowledge of language by specifying a grammar that models the speaker's knowledge explicitly and which is distinct from the computational mechanisms that constitute the language processor (Kaplan and Bresnan 1982). The central questions for LFG are thus largely the same as for other varieties of generative grammar: What is knowledge of language? How is it acquired? How is the knowledge embedded in a psycho-computational system? How do languages differ and how are they the same? The questions of acquisition and psychological processing were pursued particularly vigorously early in the theory's development; see various papers in Bresnan (1982b) and Pinker (1984). Computational questions have been investigated in detail in numerous publications, many of them stemming from work by the Natural Language Theory and Technology group at the Palo Alto Research Center (`http://www2.parc.com/isl/groups/nltt/`), as well as from work by research teams in Europe and Asia. The typological question of similarities and differences among languages has been particularly central to the subsequent development of the theory.

In answering these questions, LFG research draws on a wide variety of evidence: native speaker intuitions, corpora, psycholinguistic evidence, typological patterns, and computational models. The Xerox Linguistic Environment (XLE; Crouch et al. 2008) is a robust computational implementation of LFG that has allowed explicit testing of theoretical hypotheses, leading to new research areas and formal innovations in the process. The development of XLE has led to computational work on efficient parsing (e.g., Maxwell and Kaplan 1991, 1993, 1996) . XLE also forms the basis for a variety of industrial applications, such as the Powerset search engine, which is based on linguistically sophisticated natural language understanding (as opposed to the more superficial 'bag of words' approach that is the norm).

A central idea of Lexical-Functional Grammar is that different kinds of linguistic information are modelled by distinct, simultaneously present grammatical modules, each having its own formal representation. The grammatical architecture of LFG thus postulates a number of simple data structures with mappings defining the relationships between structures. The different grammatical modules are subject to separate principles and formal descriptions and have distinct primitives. However, at the heart of the architecture are simple set-theoretic concepts. The structures are defined in terms of sets of primitive elements and functions and relations on these sets. The mappings between structures are also defined in terms of functions and relations. LFG's formal architecture is thus typically referred to as a *Parallel Projection Architecture* or *Correspondence Architecture* (Kaplan 1987, Halvorsen and Kaplan 1988, Kaplan 1989, Asudeh 2006),[1] because different grammatical components are present in parallel and correspond to or are projected to each other by what are alternatively called correspondence or projection functions. This kind of architecture contrasts strongly with architectures in which different kinds of grammatical information are modelled by identical data structures and are subject to the same operations. LFG can be contrasted, for example, with some versions of Principles & Parameters Theory, where morphological, syntactic and semantic information alike are modelled with phrase structure trees and where phrases, words, morphemes and features alike are combined with the same operations for insertion and manipulation of syntactic structures. Similarly, in Head-Driven Phrase Structure Grammar, all linguistic information is modelled by directed acyclic graphs (Pollard and Sag 1994). LFG is often

[1] In order to avoid potential confusion with the distinct 'Parallel Architecture' developed by Jackendoff (1997, 2002), we will use the latter name.

viewed as a syntactic framework, but it is important to note that other modules of grammar have also been developed within the correspondence architecture of LFG. We return to this topic in section 5.

2 C-structure and F-structure

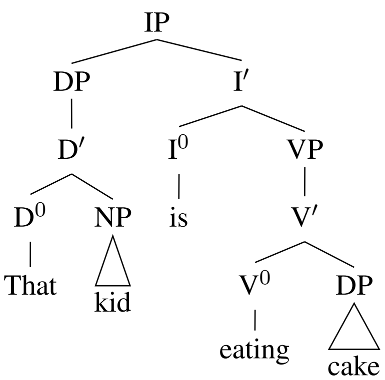
LFG posits two syntactic structures: constituent structure (c-structure; occasionally also referred to as *categorial structure*) and functional structure (f-structure). This section describes the two structures and presents the linguistic intuitions that lie behind their separation.

C-structures are represented as phrase structure trees and model precedence (word order), dominance, constituency and syntactic categories. F-structures are represented as feature structures (also known as attribute value matrices). An f-structure is a finite set of attribute–value pairs, such that an attribute is a symbol and its value is: a) a symbol (e.g., SINGULAR or +); b) a semantic form (a potentially complex symbol in single quotes); c) a set; or d) an f-structure. The f-structure of a sentence contains the grammatical functions that the head verb subcategorizes for (SUBJECT, OBJECT, etc.) and also represents a range of morphosyntactic information, such as case, agreement features, tense and aspect. F-structure is the level at which abstract syntactic relations are captured, such as agreement, control and raising, binding, and unbounded dependencies.

Turning to an example, the c-structure and f-structure for (1) are shown in (2) and (3) respectively.

(1) That kid is eating cake.

(2)



(3)

$$\begin{bmatrix} \text{PRED} & \text{`eat}\langle\text{SUBJ,OBJ}\rangle\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`kid'} \\ \text{DEIXIS} & \text{DISTAL} \\ \text{DEFINITE} & + \\ \text{NUMBER} & \text{SINGULAR} \\ \text{PERSON} & 3 \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`cake'} \\ \text{NUMBER} & \text{SINGULAR} \\ \text{PERSON} & 3 \end{bmatrix} \\ \text{TENSE} & \text{PRESENT} \\ \text{ASPECT} & \text{PROGRESSIVE} \\ \text{PARTICIPLE} & \text{PRESENT} \end{bmatrix}$$

The vertical order of features in an f-structure is not important, since an f-structure is just an unordered set of attribute–value pairs. A richer example f-structure can be found in appendix A.

F-structure can be compared to the 'relational networks' of Relational Grammar (Perlmutter and Postal 1977, Perlmutter 1983), since both structures model grammatical functions (or relations). However, the formalization is very different. First, the values of LFG grammatical functions are feature structures which contain morphosyntactic information, such as case and agreement features. In general, LFG f-structures thus tend to contain considerably more grammatical information than relational networks. Second, relational networks represent relation changes in tiered strata — where subsequent strata are derived derivationally from previous strata. In contrast, such changes are not represented in f-structure, since a key tenet of LFG theory is that all relation changes are lexical (Bresnan 1978, 1982c, Kaplan and Bresnan 1982; see Bresnan (2001b: 25–40) for an overview of some of the main arguments).[2]

C-structure models the surface exponence of syntactic information, such as word order and constituency, whereas f-structure models more abstract syntactic information and relations. C-structures may vary widely between languages, but f-structural information remains relatively constant across languages. It is thus at f-structure that we observe many cross-linguistic universals. Consider passives, for example. Perlmutter and Postal (1977) show that it is not possible to describe the passive cross-linguistically with reference to verbal morphology, case marking or word order. What regular passives[3] have in common cross-linguistically is that the subject is demoted and the object is promoted to subject. However, not all languages mark their subjects and objects the same: in some languages, subjects are distinguished from other functions with case marking, in some with agreement on the verb, and yet others distinguish between these (and other) grammatical functions with word order and phrase structure. Of course, many languages use a combination of several linguistic devices to distinguish between grammatical functions. F-structure directly models grammatical functions, such as subjects and objects, whereas c-structure displays the more superficial information about how the functions are encoded in a given language. The LFG analysis of passives and other relation changes is captured in the mapping between argument roles (such as *agent*, *patient*, etc.) and grammatical functions. The difference between an active and a passive sentence lies in which argument is realized as the subject at f-structure. How this subject is expressed in c-structure is language-specific. The theory of these mappings was initially developed in the eighties and nineties and is called Lexical Mapping Theory (Levin 1985, Bresnan and Kanerva 1989, Alsina 1996, Butt 1995, Butt and King 2000a), and this has been a very active area of research in LFG. See Bresnan (2001b: 318–320) for numerous additional references.

The remainder of this section examines some characteristics of c-structure and f-structure in turn.

2.1 C-structure

C-structures are constrained by the principle of Lexical Integrity (see Bresnan (2001b: 91–93) for an overview):

(4) **Lexical Integrity**
The terminal nodes of c-structures are morphologically complete words.

This has two immediate consequences. First, terminal nodes in c-structure cannot be morphemes or morphological structures smaller than words, in contrast to what obtains in certain other theories (e.g., Distributed Morphology; Halle and Marantz 1993, Embick and Noyer 2007, among others). The syntax is therefore blind to the internal structure of words and sees only their category. This has a number of further consequences, which are explored in the LFG literature. For example, morphological units are correctly predicted not to support certain syntactic operations, such as extraction, gapping, coordination, certain anaphoric dependencies, and certain kinds of recursion. These consequences of Lexical Integrity are considered by Bresnan and Mchombo (1995), who show that Lexical Integrity provides a principled explanation of the complex syntactic, morphological and prosodic properties of Bantu noun class markers.

[2] Role and Reference Grammar (RRG; Van Valin 1993, 2005) also posits grammatical functions, as syntactic arguments tied to semantic roles such as Actor and Undergoer. RRG is based on quite different conceptual foundations from LFG, since the former is a functionalist linguistic theory and LFG is not. See Farrell (2005) for a comparison of grammatical relations in LFG and RRG.

[3] The 'regular' passive can be compared to *impersonal passives*, where the object is not promoted, and *pseudo-passives*, where a prepositional object is promoted.

A second consequence of Lexical Integrity, which has not thus far received as much attention in the LFG literature (although, see Asudeh et al. 2008), is that terminal nodes in c-structure cannot be syntactic units larger than morphological words. That is, the lexicon does not provide c-structure with fully formed phrases; compare, for example, the lexically stored phrases of Construction Grammar (Fillmore 1988, Goldberg 1995, Kay and Fillmore 1999).

Pre-terminals are labelled with the syntactic category of the word that fills the terminal node. The set of category labels includes a number of lexical categories: N(oun), V(erb), P(reposition), A(djective), and Adv(erb). Many LFG researchers have also adopted a limited set of functional categories and projections (see, e.g., Kroeger 1993, King 1995, Bresnan 2001b, Dalrymple 2001). The functional categories assumed are typically C(omplementizer), I(nflection) and D(eterminer). In general, the only functional categories adopted in LFG are ones involved in word order and distributional generalizations. For example, the categories C and I are involved in LFG analyses of head displacement phenomena, such as verb-second in Germanic languages and the distribution of English auxiliaries. Functional categories such as K (Case) and Agr(eement) are therefore not adopted,[4] since information about case and agreement is captured in the morphology and at f-structure.

The *exocentric* (i.e., lacking a phrase structure head) category S is widely adopted within LFG. It serves two purposes. First, it is used in analyses of languages that lack a VP and display a flat constituent structure, such as Warlpiri (Simpson 1983, 1991, Nordlinger 1998, Bresnan 2001b). Second, it is used in analyses of $[_S$ NP XP$]$ predication structures, where the predicate phrase XP may be VP, NP, AP or PP. These sorts of predication structures are common in Celtic languages (see, e.g., Chung and McCloskey 1987 for Irish).

2.2 F-structure

One of the principal motivations for the name *functional* structure is the fact that grammatical functions are represented at f-structure. A second motivation is that functional structures are finite functions in the mathematical sense, due to the condition on f-structure wellformedness known as Uniqueness or Consistency:[5]

(5) **Uniqueness/Consistency**
Every f-structure is such that every attribute has exactly one value.

F-structures are thus total functions from attributes to values. However, they may be many-to-one functions: different attributes may have the same value. Shared values can be observed in standard LFG analyses of raising, obligatory control, and unbounded dependencies.

Grammatical functions are a reflection of predicate-argument relations, and a central purpose of f-structure is to capture these relations. One motivation for this is the typological observation that nonconfigurational languages (e.g., Warlpiri) encode similar predicate-argument relations to configurational languages. A non-configurational language and a configurational language may have the same f-structure corresponding to strikingly different c-structures; see Bresnan (2001b: 5–10) for an expository discussion of this point with respect to Warlpiri and English. A second, overarching motivation is the observation that many syntactic phenomena can be compellingly analyzed in terms of predicate-argument relations (cf. the discussion of passives above). A distinguishing feature of LFG is its adoption of a rich inventory of grammatical functions as primitives of the theory. Table 1 contains an overview of LFG's grammatical functions.

The grammatical functions (GFs) in table 1 can be cross-classified in a number of ways. First, a subset of the grammatical functions — the *governable grammatical functions* — may be directly selected by predicates.

(6) **Governable grammatical functions**: SUBJ, OBJ, OBJ_θ, OBL_θ, COMP, XCOMP

[4] There are exceptions. For example, Butt and King (2004) adopt the category K for Hindi–Urdu case endings, because they argue that these are clitics and they want to maintain a generalization that only functional heads may be clitics.

[5] LFG is not committed to 'functional' in the sense of functionalist linguistics, namely having to do with communicative functions of language.

SUBJECT (SUBJ)	Some people with no shame walked in and wrecked the party. The party was wrecked by some people with no shame.
OBJECT (OBJ)	*First object.* Ricky trashed the hotel room. Ricky gave John a glass. Ricky gave a glass to John.
OBJECT$_\theta$ (OBJ$_\theta$)	*Second object. Thematically restricted object.* Sandy gave John a glass. Tom baked Susan a cake. #Tom baked a cake Susan. (OBJ$_\theta$ *in English restricted to theme, cannot be beneficiary*)
OBLIQUE$_\theta$ (OBL$_\theta$)	*A complement (non-subject argument) that has oblique case or is a PP.* Julia placed the vase on the desk. Ricky gave a glass to John.
COMPLEMENT (COMP)	*Closed (saturated) complement: A clausal argument which has its own subject.* Peggy told Matt that she had won the prize.
XCOMP	*Open (unsaturated) predicate complement: A predicative argument with no overt subject of predication.* I told Patrick to quit. Peggy-Sue seems to be a complete fraud.
ADJUNCT (ADJ)	*A modifier, a non-argument.* Mary read a good book. Mary counted the cars very quickly. Sally killed a bug in the yard. Since she had no money, Mary was forced to get a job.
XADJ	*Open predicate adjunct.* Having no money, Mary was forced to get a job.
SPECIFIER (SPEC)	*Possessor or quantificational determiner phrase.* John's book's cover is red. At least three books are red.
TOPIC (TOP)	*Grammaticalized discourse function.* *Must be identified with or anaphorically linked to another grammatical function.* Mary met the author whose books __ annoyed Peggy. (TOPIC = SUBJ) Bagels, Mary loves __. (TOPIC = OBJ) As for bagels, Mary loves them. (TOPIC anaphorically linked to OBJ)
FOCUS (FOC)	*Grammaticalized discourse function.* *Must be identified with or anaphorically linked to another grammatical function.* Which author do the critics praise __? (FOCUS = OBJ) Cén t-údar a molann na léirmheastóirí é? (FOC anaphorically linked to OBJ) Which author COMP praise the critics him (Irish; McCloskey 1979: 53)

Table 1: Grammatical functions in LFG

All other GFs are non-governable and cannot be specifically selected, but rather occur freely, subject to other constraints of the theory.

The core nominal grammatical functions are further decomposed in Lexical Mapping Theory according to the features [± r(estricted)] and [± o(bjective)], as follows:

(7) **LMT decomposition of core nominal GFs**

	−o	+o
−r	SUBJ	OBJ
+r	OBL_θ	OBJ_θ

Grammatical functions that are tied to specific thematic roles are [+r], whereas functions that are semantically unrestricted are [−r]. For example, subjects and direct objects can bear any semantic role, whereas second objects are typically restricted to patients and themes. Subjects and objects are also unrestricted in another sense: they can be expletives, lacking a semantic role entirely. The feature [+o] distinguishes the object functions, OBJ and OBJ_θ, from the subjects and obliques.

A second cross-classification of grammatical functions is according to whether the GF is *closed* or *open*. The open grammatical functions are the open complement function XCOMP and the open adjunct function XADJ. Open grammatical functions contain a predicate that requires a subject of predication, but which depends on external specification of the subject through the functional predication relation known as *functional control*. A functional control equation relates the XCOMP or XADJ's subject to a grammatical function in the f-structure in which the XCOMP or XADJ occurs (Bresnan 1982a). A typical instance of an XCOMP is the complement of a raising verb:

(8) Alfie seemed to laugh.

(9) Alfie vaikutti nauravan. Finnish
Alfie seemed laugh
Alfie seemed to laugh.

(10)
$$\left[\begin{array}{ll} \text{PRED} & \text{`seem}\langle\text{XCOMP}\rangle\text{SUBJ'} \\ \text{SUBJ} & \left[\begin{array}{ll}\text{PRED} & \text{`Alfie'}\end{array}\right] \\ \text{XCOMP} & \left[\begin{array}{ll}\text{PRED} & \text{`laugh}\langle\text{SUBJ}\rangle\text{'} \\ \text{SUBJ} & \end{array}\right] \end{array}\right]$$

The connecting line in the f-structure represents the functional control relation, which in this case is lexically specified by the raising verb *seemed* (informally: 'my SUBJ is my XCOMP's SUBJ'). In contrast, the functional control relation for an English XADJ would be associated with a c-structure position rather than specified lexically, since XADJ is a non-governable grammatical function that is not selected for, but rather appears freely, like other adjuncts, in certain structural positions.

A third cross-classification of grammatical functions is according to whether they are grammaticalized discourse functions or not. The discourse functions are typically structurally prominent in some way, as an expression of their discourse prominence.

(11) **Discourse functions**: TOPIC, FOCUS, SUBJ

SUBJECT is the only discourse function that is also a governable GF. TOPIC and FOCUS are not selected directly, but are rather integrated into the f-structure by the Extended Coherence Condition (see (55) below). In many languages, the SUBJ is also identified by default as the TOPIC. See Falk (2006) for a recent in-depth LFG-theoretic investigation of subjects.

Grammatical functions are subcategorized for in lexically specified PRED features, which we have already encountered in (3) and (10). For example, the verb *eating* has the PRED value 'eat⟨SUBJ,OBJ⟩'. The first part of this value is the predicate function, which is conventionally the stem form of the lexical item that contributes the PRED. It is also a common convention for the predicate function to be written in a convenient meta-language for the linguist, rather than in the language of analysis. For example, the PRED value for the Finnish *vaikutti* in (9) is 'seem...'. The PRED feature also specifies how many and which governable grammatical functions the verb selects, as indicated in its argument list (the grammatical functions specified after the predicate function).

Lastly, a distinction is drawn between thematic and non-thematic arguments. Thematic arguments are written within the angled brackets, whereas non-thematic arguments are written following the angled brackets. For example, the PRED of *seemed* in (10) is 'seem⟨XCOMP⟩SUBJ', which indicates that the XCOMP complement is the only thematic argument of *seem* and that the raising verb's SUBJ is non-thematic. There is a general requirement that thematic arguments must have a PRED feature, since they are semantically contentful, whereas non-thematic arguments need not have a PRED, since they are not semantically contentful. For example, the SUBJ of a raising-to-subject verb may be an expletive. Expletives in LFG are analyzed as lacking a PRED feature but having appropriate agreement features. It is important to realize, though, that PRED is not a semantic representation, but rather the syntactic exponent of certain semantic information.

The value of a PRED attribute is a *semantic form*, which is indicated by the enclosing single quotes. Semantic forms are special, complex symbols that are always uniquely instantiated. This is captured formally through indexation on semantic forms, e.g. 'eat$_{12}$⟨SUBJ,OBJ⟩', but the indices are typically suppressed. Unique instantiation of semantic forms ensures that semantically relevant information is not multiply specified syntactically. For example, consider the following examples from Irish (McCloskey and Hale 1984: 489–490):

(12) Chuirfinn isteach ar an phost sin. — Irish
put.COND.1SG in on that job
I would apply for that job.

(13) * Chuirfinn mé isteach ar an phost sin.
put.COND.1SG I in on that job

Irish has both *synthetic* and *analytic* forms of verbs in certain paradigms. Synthetic forms contain complete pronominal information and cannot occur with an overt pronominal, even if the pronoun is compatible in agreement features with the verb. *Chuirfinn* is the synthetic form of the conditional form of *cuir* ('put') in the first person singular. Example (13) is thus ungrammatical because the synthetic verb form cannot occur with the overt pronominal. Andrews (1990) shows that this falls out neatly from the uniqueness of semantic forms in LFG. The synthetic verb form incorporates pronominal information (McCloskey and Hale 1984) and therefore contributes its SUBJECT's PRED feature, specifying its value as 'pro' (the standard LFG-theoretic PRED value for non-expletive pronominals). The independent pronoun *mé* also contributes a PRED feature with value 'pro'. However, the two instances of 'pro' are unique semantic forms and thus cannot simultaneously be the value of a single PRED feature. This results in a violation of Consistency, defined in (5) above. Example (13) is thus correctly predicted to be ungrammatical. The situation exemplified here can be contrasted with 'pro-drop' languages, in which the verb's contribution of its SUBJ's PRED is optional and the verb therefore may appear with a suitably agreeing overt subject; see the Romanian examples in (42–44) below. The theory thus derives the distinction between obligatory suppression of a pronominal subject, as in these Irish cases, from optional suppression of a pronominal subject, as in Romanian, based on obligatoriness versus optionality of relevant lexical information.

In addition to Consistency, there are two other general wellformedness conditions which apply to all f-structures:

(14) **Completeness**
An f-structure is *complete* if and only if it contains all the governable grammatical functions that its predicate governs.

(15) **Coherence**
An f-structure is *coherent* if and only if all the governable grammatical functions it contains are governed by a predicate.

Note that the term 'govern' means nothing more than to be listed in the argument list of a PRED feature.

Completeness and Coherence serve a similar role in LFG as the Projection Principle, the Theta Criterion and Full Interpretation do in P&P and that the Subcategorization or Valence Principle does in HPSG. They ensure that the subcategorization requirements of a predicate are met exactly. Coherence violations occur if a constituent cannot be mapped to any GF (i.e., if there are "extra" arguments):

(16) *Thora remembered every movie most videos.

(17) *That the earth is round did not surprise Columbus that he could sail west without danger.

Completeness violations occur if subcategorized GFs are not present, as in the following examples:

(18) *Alfie devoured.

(19) *John wondered if seemed to be a problem.

Example (19) illustrates that Completeness requires even non-thematic governed GFs to be present. Even though the SUBJ of *seemed* is non-thematic it is still required by Completeness; that is, Completeness applies to *all* GFs in a PRED's argument list, both inside and outside the angled brackets.

3 Structures and Structural Descriptions

LFG distinguishes sharply between formal structures, such as c-structures and f-structures, and structural descriptions that wellformed structures must satisfy. The structural descriptions are sets of constraints. A constraint is a statement that is either true or false of a structure. This section provides an overview of the most important sorts of constraints. For a more thorough discussion, see in particular Dalrymple (2001: 91–176).

3.1 Constraints on C-structures

The formal structures in c-structure are phrase structure trees, as illustrated in (2) above. The structural descriptions that constrain the phrase structure trees are formalized as phrase structure rules, such as (20):

(20) $\text{IP} \quad \rightarrow \quad \text{DP} \quad \text{I}'$

A wellformed c-structure must satisfy all applicable phrase structure rules and every sub-tree in a wellformed c-structure must satisfy some phrase structure rule. The body of LFG's phrase structure rules are *regular expressions*, which support optionality, disjunction, negation, and arbitrary repetition. Regular expression repetition uses the Kleene operators (Kleene 1956): Kleene star (*), which means 'zero or more occurrences of the annotated expression', and Kleene plus ($^+$), which means 'one or more occurrences of the annotated expression'. LFG's phrase structure rules are comparable, in this specific respect, to the phrase structure rules of Generalized Phrase Structure Grammar (GPSG; Gazdar et al. 1985), which also support regular expressions (Gazdar et al. 1985: 54–55). A formal exposition of regular expressions can be found in Partee et al. (1993: 462–464).

Consider, for example, the following V′ rule, proposed solely for illustration:

(21) $\text{V}' \quad \rightarrow \quad \text{V}^0 \quad (\text{NP}) \quad (\,\{\,\text{CP} \mid \text{VP}\,\}\,) \quad \text{PP}^*$

Optionality is indicated by parentheses around a rule element. Disjunction is indicated with the notation $\{\,\text{X} \mid \text{Y}\,\}$. Rule (21) has a single obligatory element, the V^0. The verb may be immediately followed by an NP sister. The verb may also have either a CP or a VP sister or neither (since the entire disjunction is within the scope of optionality parentheses). Lastly, the V′ may end in any number of PPs, including none.

Phrase structure rules are posited separately for independent languages, subject to certain universal principles. A structure is allowed only if it is linguistically motivated for that language. The motivation consists primarily of distributional evidence (for category assignment), constituency evidence and word order. For example, if the verb appears after its complements in a given language, the VP rule for that language is V-final. There is no attempt to derive all surface word orders from a universal underlying word order, such as SVO (Kayne 1994); this notion makes no sense in LFG, since the theory is not derivational and does not postulate underlying word order that is distinct from surface word order. LFG's 'surface-true' approach to phrase structure is further evidenced by the fact that a VP is posited only if there is distributional or constituency evidence for such a category. A language without a VP is a non-configurational language (see Nordlinger 1998 and references cited therein for definitions of non-configurationality).

Although c-structures vary greatly cross-linguistically, the variation seems to be limited in a principled way. This is captured in LFG with X-bar theory (Chomsky 1970, Bresnan 1977, Jackendoff 1977) and certain universal principles on the c-structure to f-structure mapping. The mapping principles are discussed in detail in Bresnan (2001b: 98–109) and Toivonen (2003: 66-69). One principle states that "c-structure heads are f-structure heads". This means that a c-structure head maps its featural information into the same f-structure as its c-structural mother. Such principles sharply limit the combinatorial possibilities at c-structure.

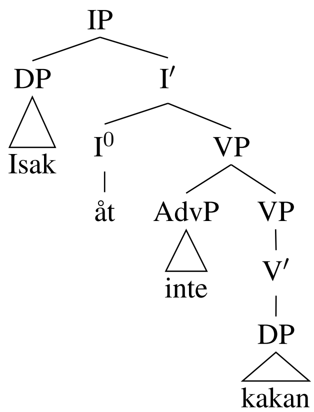
LFG allows for both *endocentric* c-structures and *lexocentric* c-structures, the latter rooted in the exocentric category S, as discussed in section 2.1. Lexocentric phrase structure is instantiated in languages where grammatical functions are encoded morphologically rather than configurationally. Lexocentric structure is both typologically common and diverse (instantiated in genetically and geographically unrelated languages). However, the theory assumes that S is the only exocentric category and that, even within lexocentric languages, other categories are endocentric. The theory thus posits S as a principled exception to X-bar theory in order to capture phrase-structural properties of lexocentric languages without forcing them into a configurational mould.

LFG's use of X-bar theory provides a good illustration of the conceptual difference between structures and structural descriptions. Consider a typical LFG analysis of 'head movement' phenomena (Travis 1984), which in LFG do not involve movement at all, but rather lexical specification of a functional category such as I^0 for a verb (King 1995). For example, consider Germanic verb-second, as instantiated in Swedish:

(22) Isak åt inte kakan. — Swedish
Isak ate not cookie.DEF
Isak did not eat the cookie.

The Swedish finite verb in this example has the category I^0 in the lexicon and is thus analyzed as base-generated in I^0 (Sells 2001a, Toivonen 2001, 2003), yielding the following structure:

(23)



The V′ in (23) does not contain a V^0, a violation of X-bar theory as a theory of c-structures. However, the relevant phrase structure rule — which a tree rooted in V′ must satisfy — does contain a V^0, although an optional one:

(24) $V' \quad \rightarrow \quad (V^0) \quad \ldots$

Thus, X-bar theory in LFG holds as a theory of structural descriptions. For more detailed discussions of X-bar theory and LFG's theory of phrase structure, see Bresnan (2001b: chapter 6) and Toivonen (2003: chapter 3).

Lastly, LFG's theory of c-structure does not posit any principle that dictates that multiply branching structures are disallowed. For example, both objects of a ditransitive verb are sisters of the verb. Coordination structures are also multiply branching. LFG rejects the contention that all phrase structure is binary branching (Kayne 1984). The putative evidence for that claim concerns phenomena that are analyzed at f-structure.

3.2 Constraints on F-structures

F-structure constraints are stated in a quantifier-free theory of equality. F-structure constraints are specified in lexical entries and in annotations on nodes of c-structures, as explained in more detail in section 4 below. The set of all f-structure constraints obtained from the lexical entries and c-structure of a given analysis is called a *functional description* or *f-description*.

A common kind of constraint is a *defining equation*, which specifies the value of some attribute in an f-structure. For example, the following defining equation specifies that the NUMBER attribute of some f-structure f has the value SINGULAR:

(25) $(f\ \textsc{number}) = \textsc{singular}$ *defining equation*

The values of f-structures can also be semantic forms and other f-structures, so we also get these sorts of defining equations:

(26) $(f\ \textsc{pred}) = \text{'laugh}\langle\textsc{subj}\rangle\text{'}$

(27) $(f\ \textsc{subj}) = g$

The equation in (27) states that the SUBJECT of f-structure f is f-structure g.

Recall that f-structures are functions. Thus, an equation such as (25) can be understood as a kind of functional application, where we write the parentheses as above instead of the more standard (28):

(28) $f(\textsc{number}) = \textsc{singular}$

The reason this notional difference was instituted is that it makes iterative functional applications easier to understand. For example, consider the partial f-structure, f, in (29).[6]

(29)
$$f\colon \begin{bmatrix} \textsc{pred} & \text{'smile}\langle\textsc{subj}\rangle\text{'} \\ \textsc{tense} & \textsc{present} \\ \textsc{subj} & g\begin{bmatrix} \textsc{person} & 3 \\ \textsc{number} & \textsc{singular} \end{bmatrix} \end{bmatrix}$$

Now suppose that (29) represents part of the f-structural information of sentence (30) and we want to specify subject-verb agreement.

(30) Alfie smiles.

We can capture the agreement by specifying the following two equations in the verb's lexical entry:

(31) $(f\ \textsc{subj number}) = \textsc{singular}$
$(f\ \textsc{subj person}) = 3$

Given that f's SUBJ is g in (29), these simplify to:

[6] We have written a colon after the f-structure label f to make clear that the f-structure is the function f. We will henceforth suppress the colon, but a label f on an f-structure should be read as the name of the f-structure, not as a function f applied to an unnamed f-structure.

(32) $(g \text{ NUMBER}) = \text{SINGULAR}$
$(g \text{ PERSON}) = 3$

These equations will only be satisfied if the subject *Alfie* is compatible with the number and person specifications indicated in (32). Since this is indeed the case, (30) is correctly predicted to be grammatical. In sum, successive functional applications can be represented by writing what amount to *paths* of attributes.

Defining equations can be expressed in terms of relations other than basic equality. One common relation is set membership, since at f-structure modification is represented as a set that is the value of an ADJUNCT grammatical function. ADJUNCTS are represented as sets because there is no upper bound on the number of modifiers that a constituent may have.

(33) $g \in (f \text{ ADJ})$

This equation states that the f-structure g is a member of the ADJ set of f-structure f; see appendix A for an f-structure containing ADJUNCT sets. Sets are also used in the f-structural representation of coordination (Kaplan and Maxwell 1988, Maxwell and Manning 1996) and in a more articulated theory of morphosyntactic features that accommodates resolution of coordinated morphosyntactic information (Dalrymple and Kaplan 2000).

The solution for a given f-description is the *minimal* f-structure that satisfies the set of constraints. The minimal f-structure contains only the attributes and values that are explicitly mentioned in the f-description. If the minimality constraint did not hold, then the f-structure for (34) would equally satisfy the f-description for (35), since the additional modifier *quickly* contributes information that is not inconsistent with the smaller f-structure.

(34) Alfie quickly ran out.

(35) Alfie ran out.

However, it is clear that we would not want (34) and (35) to have the same f-structural parse, because they are syntactically distinct sentences.

A second kind of equation, the *constraining equation*, takes advantage of the minimality requirement. Constraining equations do not define the features and relations in an f-structure, but rather check that the minimal f-structure has the features or relations specified by the constraining equation. Formally, the constraining equations are evaluated once the set of defining equations has been satisfied by the minimal f-structure. A constraining equation is written with a subscripted c:

(36) $(f \text{ PARTICIPLE}) =_c \text{PRESENT}$ *constraining equation*

This equation does not result in f-structure f having the feature PARTICIPLE with value PRESENT. Rather, it checks that f contains that feature and value. An independent defining equation must actually specify the feature and value.

In order to see how this is useful, consider these examples (following a similar discussion in Kaplan and Bresnan 1982):

(37) Thora is giving Harry a toy.

(38) *Thora is gives Harry a toy.

Let us assume that the progressive auxiliary *is* and the participle *giving* map to the same f-structure, f, and that the constraining equation (36) is part of the lexical information associated with the auxiliary. Let us make the natural assumption that the present participle *giving* has a PARTICIPLE feature with value PRESENT, lexically specified through a defining equation associated with the participle. See (65) in section 4.1 below for the relevant f-descriptions. Since the auxiliary's constraining equation is thus satisfied in (37), the sentence is correctly predicted to be grammatical. In contrast, let us assume that the present tense form *gives* does not specify any participial information, since it is not a participle form of the verb. Example (38) is thus ruled out, because the auxiliary's constraining equation cannot be satisfied, since *gives* does not provide the required information.

Now consider what would be the case if (36) were a defining equation rather than a constraining equation. If *gives* did not provide any information to the contrary, then the progressive auxiliary would actually just add the feature PARTICIPLE with value PRESENT and (38) would incorrectly be predicted to be grammatical. In order to block (38), every non-participial verb would have to specify a participle feature with a value such as NIL or NON-PARTICIPIAL. The constraining equation allows us to avoid this inelegant and unnatural situation, since only participles need be marked as such. This participial example demonstrates one of the key uses of constraining equations, which is to control co-occurrence of words or phrases through their associated f-structure features.

There are three other useful kinds of constraints on minimal solutions. *Negative equations* are satisfied if and only if a feature has a value other than the one specified (including complete absence of the feature):

(39) $(f\ \textsc{case}) \neq \textsc{nominative}$ or $\neg[(f\ \textsc{case}) = \textsc{nominative}]$ *negative equation*

The first notation is somewhat more common. The negative equation (39) is satisfied if and only if f has no CASE feature or if the value of CASE is something other than NOMINATIVE.

The last two kinds of constraint are the *existential constraint*, which is satisfied if and only if the attribute in question is present (regardless of its value), and the related *negative existential constraint*, which is satisfied if and only if the attribute in question is absent (regardless of its value). Here is an example of each kind of constraint:

(40) $(f\ \textsc{case})$ *existential constraint*

(41) $\neg(f\ \textsc{case})$ *negative existential constraint*

The existential constraint (40) requires f to have a CASE feature. The negative existential constraint (41) requires f not to have a CASE feature.

The boolean connectives of conjunction, disjunction and negation can be used in f-descriptions. Conjunction is typically implicit: in any f-description, all the constraints must hold. Conjunction can also be explicitly indicated with the standard symbols '&' or '$\wedge$'. Disjunction is indicated either with the symbol '$\vee$' or in the form '$\{X \mid Y\}$'. Negation is indicated with the symbol '$\neg$'. Grouping is indicated by square brackets, '[...]'. Optionality is once again indicated by parentheses, '(...)'.

Judicious use of these connectives allows for compact specification of f-structure constraints. For example, consider the following two examples from Romanian, a pro-drop language that shows syncretism of first and second person singular in certain conjugations (Cojocaru 2003: 120–126):

(42) Eu/tu continui. Romanian
I/you continue.PRES.[1.SG/2.SG]
I/you continue.

(43) Continui.
continue.PRES.[1.SG/2.SG]
I/you continue.

(44) * Ea continui.
she continue.PRES.[1.SG/2.SG]

The verb *continui* ('continue') lexically contributes the following f-description, where f is the f-structure of the sentence:

(45) *continui* $(f\ \textsc{pred}) = \text{`continue}\langle\textsc{subj}\rangle\text{'}$
$(f\ \textsc{tense}) = \textsc{present}$
$(\ (f\ \textsc{subj pred}) = \text{`pro'}\ )$
$(f\ \textsc{subj number}) = \textsc{singular}$
$(f\ \textsc{subj person}) \neq 3$

The negative equation for SUBJ PERSON in (45) correctly blocks the ungrammatical (44), while specifying no positive person information about the subject, which correctly reflects uncertainty of knowledge about the form (i.e., ambiguity). Another example of syncretism of agreement features is shown for English main verbs in appendix A.

The f-description in (45) also demonstrates the standard LFG treatment of pro-drop: the verb optionally specifies that its SUBJ has the PRED value 'pro'. This allows the f-structure for a pro-drop sentence, such as (43), to satisfy Completeness, since the thematic SUBJ that the verb governs is present and has a PRED. The f-structure for (43), which satisfies the f-description (45), is:

(46)
$$f\begin{bmatrix}\text{PRED} & \text{`continue}\langle\text{SUBJ}\rangle\text{'}\\ \text{TENSE} & \text{PRESENT}\\ \text{SUBJ} & \begin{bmatrix}\text{PRED} & \text{`pro'}\\ \text{NUMBER} & \text{SINGULAR}\end{bmatrix}\end{bmatrix}$$

We noted above that multiple functional applications can be written as an f-structure label followed by a string of symbols, as in (f SUBJ NUMBER). Kaplan and Zaenen (1989) develop an f-structure-based theory of unbounded dependencies that relies on a simple extension to this, such that the string of symbols is drawn from a regular language. This means that optionality, negation, disjunction, complementation and Kleene star and plus are valid operations on the string of attributes in an f-structure constraint. The regular expression operators allow the statement of f-structure constraints that contain *functional uncertainty* and are thus resolvable in a (potentially unlimited) number of ways. This use of regular expressions is similar to the GPSG theory of unbounded dependencies, which is stated in terms of of slash categories in phrase structure rules that support regular expressions (Gazdar 1981, Gazdar et al. 1985). One crucial difference, discussed below, is that the LFG functional uncertainty approach does not need to posit traces in phrase-structure.

Let us consider an example. We noted in table 1 that *wh*-phrases in interrogatives are assigned the discourse grammatical function FOCUS. Suppose that we want to allow the *wh*-phrase to correspond to the grammatical functions SUBJ or OBJ. We could then write the following equation:

(47) $(f\ \text{FOC}) = (f\ \{\text{SUBJ} \mid \text{OBJ}\})$

The right-hand side of the equation contains an uncertainty about which grammatical function the *wh*-phrase is identified with.

The equation in (47) does not yet capture the unbounded nature of *wh*-dependencies. Using the Kleene operators, we add a further, unbounded uncertainty over the grammatical functions in the f-structure that the dependency may licitly pass through. For example, the following equation states that the *wh*-dependency may pass through any number (including zero) of XCOMP or COMP grammatical functions and must be identified at the bottom of the dependency with a SUBJ or OBJ:

(48) $(f\ \text{FOC}) = (f\ \{\text{XCOMP} \mid \text{COMP}\}^*\ \{\text{SUBJ} \mid \text{OBJ}\})$

This captures the same effects as (47), but now allows for unboundedness, generating examples such as:

(49) Who saw this?

(50) What did John see?

(51) What did Mary say that John saw?

(52) What did Mary seem to say that John saw?

Island constraints and other constraints on extraction are captured through the path specification in the functional uncertainty equation. For example, the equation in (48) already captures the Sentential Subject Constraint, ruling out (53), because SUBJ is not on the extraction path: the dependency can terminate in a SUBJ, but cannot pass through one. Similarly, the equation captures the Left Branch Condition, ruling out (54), because the path cannot terminate in SPEC.

(53) *Who does [that John likes __] surprised Mary?

(54) *Whose did they steal [__ car]?

Equation (48) is just meant to be illustrative and does not capture the full range of grammatical possibilities nor rule out the full range of ungrammatical cases. What (48) shows, though, is that conditions on extraction are captured in LFG by appropriately limiting the extraction path, as expressed in a functional uncertainty equation. For a more complete specification of functional uncertainty paths, including pied-piping, see Dalrymple (2001: chapter 14). Some recent in-depth investigations of unbounded dependencies in LFG are Berman (2003), Asudeh (2004), and Mycock (2006). Berman (2003) and Asudeh (2004) consider the question of successive-cyclic effects in unbounded dependencies and consider an alternative to functional uncertainty based on functional control.

The LFG approach to unbounded dependencies that developed from Kaplan and Zaenen's functional uncertainty approach is notable in that it posits no traces or copies in the syntax — whether in c-structure or f-structure. See the appendix for an illustration. Bresnan (1995, 1998, 2001b) has argued from cross-linguistic data on weak crossover that traces are required in certain narrowly circumscribed circumstances, but see Dalrymple et al. (2001, 2007) for a traceless alternative and Berman (2003: chapter 5) for a critical appraisal of both sides of the debate.

The non-argument discourse functions FOCUS and TOPIC are subject to the following general principle (Zaenen 1980, Bresnan and Mchombo 1987):[7]

(55) **Extended Coherence Condition**
FOCUS and TOPIC must be linked to the semantic predicate argument structure of the sentence in which they occur through proper integration with the sentence's f-structure. Proper integration is either functional equality with or anaphoric binding of a grammatical function.

Functional equality is the integration mechanism that we have seen so far, which is appropriate for filler-gap dependencies. Anaphoric binding is appropriate for resumption, left-dislocation, hanging topics, and other phenomena in which the discourse function has a corresponding pronoun in the clause. See Asudeh (2004) for an in-depth treatment of resumption and discussion of related cases of satisfaction of the Extended Coherence Condition through anaphoric binding.

The functional applications and functional uncertainties we have examined thus far have all been *outside-in*: in stating the constraint, some path is examined from an outer f-structure to an inner f-structure. The extension of the f-structure constraint language to allow functional uncertainty also enables *inside-out functional application* and *inside-out functional uncertainty* (first published in Halvorsen and Kaplan 1988), which permit constraints to be placed on paths from an inner f-structure to an outer f-structure.

Inside-out functional application is the formal foundation of the theory of *constructive case* developed by Nordlinger (1998) in her analysis of the Australian language Wambaya. In this theory, the case inflection directly determines the grammatical function of the nominal by stating which GF the nominal's f-structure must be the value of. We can demonstrate the generality of the idea by looking at an example from a typologically unrelated language, Malayalam (Mohanan 1982):

(56) Kuṭṭi aanaye aar̄aadiccu. — Malayalam
child.NOM elephant.ACC worship.PAST
The child worshipped the elephant.

Mohanan (1982) notes that, in Malayalam, case-marking together with animacy determines the grammatical function of the nominal. For example, an animate nominative is a subject. This is captured through the following f-description that is part of the lexical information contributed by *kuṭṭi* ('child'), where f is the f-structure of the noun:

[7] Some formulations of the Extended Coherence Condition also apply to ADJUNCTS; see, e.g., Bresnan (2001b: 63) and Falk (2001: 64).

(57) *kuṭṭi* (f PRED) = 'child'
(f ANIMATE) = +
(f CASE) = NOMINATIVE
(SUBJ f)

The final constraint in (57) is an inside-out existential constraint which requires that there is an f-structure, call it g, such that the noun's f-structure is the value of g's SUBJ attribute. For formal definitions of outside-in and inside-out functional application and uncertainty, see Dalrymple (2001: 100–104, 143–146).

Inside-out functional uncertainty plays an important role in LFG's binding theory, as initially explored in Dalrymple (1993). Constraints on antecedents of anaphors are stated in f-descriptions according to the following general schema, where f is the f-structure of the anaphor:

(58) ((DomainPath f) AntecedentPath)

DomainPath is a path that states in which domain the antecedent of the anaphor must occur. It is stated in terms of an inside-out functional uncertainty relative to the f-structure of the anaphor. AntecedentPath then specifies where within this domain the antecedent may occur and which grammatical function the antecedent has. (Dalrymple 1993) shows that this kind of equation, including constraints on properties of f-structures that DomainPath passes through, gives both a formally precise and typologically appropriate explanation of anaphoric binding possibilities.

As an example, let us consider the long-distance reflexive *aapaṇ* in Marathi, as discussed in Dalrymple (1993). This pronominal must be bound within the sentence, so it is an anaphor, but it cannot be bound locally (Dalrymple 1993: 14, 77):

(59) Tom mhanat hota ki Sue ni aaplyaalaa maarle. Marathi
Tom$_i$ said that Sue ERG self$_i$.ACC hit
Tom said that Sue hit him (Tom).

(60) * Jane ne aaplyaalaa bockaarle.
Jane ERG self.ACC scratched
Jane scratched herself.

The binding constraint on how *aapaṇ* is permitted to take an antecedent can be captured with the following inside-out functional uncertainty, where f is the f-structure of the reflexive:

(61) ((GF$^+$ GF f) GF)

The specification of DomainPath as (GF$^+$ GF f) means that the antecedent is not in the f-structure of the reflexive, which is just (GF f), but rather at least one further f-structure out (due to Kleene plus). This captures the fact that the reflexive cannot be bound locally. The AntecedentPath is simply GF, which allows the antecedent to bear any grammatical function, but this can be further restricted.

4 The C-structure to F-structure Correspondence

We have now briefly looked at c-structure and f-structure and constraints on each kind of structure, but we have yet to explain how the two structures are related by structural correspondences. This section first explains how the mapping works, and then how LFG captures the empirical observation that radically different c-structures can correspond to the same f-structure: languages can express the same basic relation with strikingly different structural and morphological tools at their disposal.

4.1 How the C-structure to F-structure Mapping Works

The correspondence function ϕ maps c-structure nodes to f-structures. The mapping is deterministic (since it is a function) and many-to-one. The mapping is determined by language-specific instantiations of general mapping principles (Bresnan 2001b, Toivonen 2003) on annotated phrase structure rules.

Lexical information is mapped from terminal nodes in c-structure, which contain all of the information lexically associated with the word. The annotations on c-structure nodes are functional constraints of the kind discussed in the previous section.

The mapping is stated in terms of two metavariables over f-structure labels, as defined in (62). These f-structure metavariables are defined in terms of a c-structure variable, $*$, which stands for 'the current node', and the mother (i.e., immediate dominance) function on tree nodes, $\mathcal{M}$, where $\mathcal{M}(*)$ is 'the node immediately dominating the current node'. It is a common LFG convention to write $\widehat{*}$ instead of $\mathcal{M}(*)$.

(62) $\downarrow \equiv \phi(*)$
i.e., 'the f-structure of the current c-structure node' or 'my f-structure'

$\uparrow \equiv \phi(\widehat{*})$
i.e., 'the f-structure of the node that immediately dominates the current c-structure node' or 'my mother's f-structure'

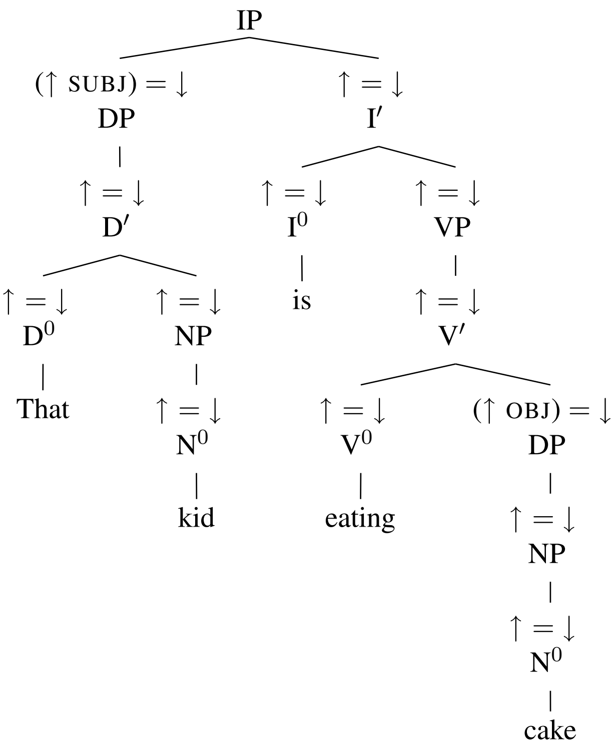
The up and down arrows are meant to symbolize their meaning graphically: since the annotations on non-terminals are typically written above the category label, the up arrow is pointing at the mother and the down arrow is pointing at the current node. This is essentially the original formalization of Kaplan and Bresnan (1982); see also Kaplan (1987, 1989). An alternative, strongly model-theoretic specification of the metavariables and LFG grammars more generally is provided by Blackburn and Gardent (1995).

The sample annotated phrase structure rule in (63) states that IP dominates a DP and an I′. The annotations specify that the information in I′ maps to the same f-structure as the information of its mother (the IP) and that the information contained in the DP maps into an f-structure that is the value of the SUBJECT grammatical function in the f-structure of the IP.

(63) $$\text{IP} \quad \longrightarrow \quad \underset{(\uparrow \text{SUBJ}) = \downarrow}{\text{DP}} \qquad \underset{\uparrow = \downarrow}{\text{I}'}$$

The annotated version of the c-structure in (2) above, which presupposes a number of additional annotated phrase structure rules like (63), is given in (64). For presentational purposes, we henceforth suppress intermediate (bar-level) categories in non-branching sub-trees; this is common practice in the LFG literature.

(64)



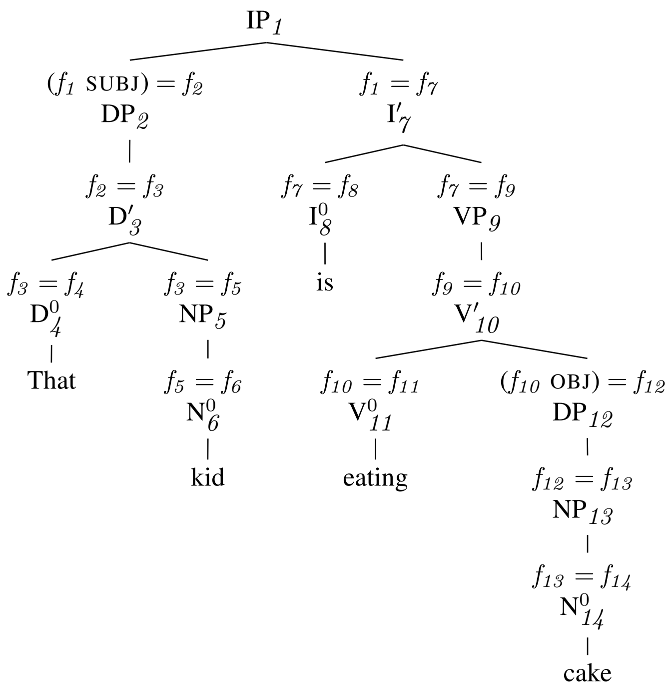
The terminal nodes in c-structure are lexical entries, which specify the form of the word, its syntactic category, and a set of f-structure constraints (the lexical item's f-description). It is more strictly correct to write the f-description of the lexical item immediately below the word form in the c-structure, since the lexical item's f-description is actually part of the terminal node's information. However, for presentational reasons, we instead specify the lexical entries separately in (65):

(65)

that, D^0	$(\uparrow \text{DEFINITE}) = +$ $(\uparrow \text{DEIXIS}) = \text{DISTAL}$ $(\uparrow \text{NUMBER}) = \text{SG}$ $(\uparrow \text{PERSON}) = 3$
kid, N^0	$(\uparrow \text{PRED}) = \text{'kid'}$ $(\uparrow \text{NUMBER}) = \text{SG}$ $(\uparrow \text{PERSON}) = 3$
is, I^0	$(\uparrow \text{SUBJ NUMBER}) = \text{SG}$ $(\uparrow \text{SUBJ PERSON}) = 3$ $(\uparrow \text{TENSE}) = \text{PRESENT}$ $(\uparrow \text{PARTICIPLE}) =_c \text{PRESENT}$
eating, V^0	$(\uparrow \text{PRED}) = \text{'eat}\langle\text{SUBJ,OBJ}\rangle\text{'}$ $(\uparrow \text{ASPECT}) = \text{PROGRESSIVE}$ $(\uparrow \text{PARTICIPLE}) = \text{PRESENT}$
cake, N^0	$(\uparrow \text{PRED}) = \text{'cake'}$ $(\uparrow \text{NUMBER}) = \text{SG}$ $(\uparrow \text{PERSON}) = 3$

The metavariables are instantiated as follows. Each c-structure node is assigned an arbitrary, unique index. The c-structure variable $*$ for each node is instantiated as the node's index and the f-structure metavariable is instantiated accordingly. Up arrow metavariables in lexical f-descriptions are instantiated according to the label of the pre-terminal node that dominates the item in question. This should be intuitively clear if one bears in mind that the f-description is actually part of the terminal node. The instantiated version of (64) and its corresponding f-structure is shown in (66). Notice that we have adopted a typical convention of writing f_1 instead of $\phi(1)$ and so on.

(66)



$$\begin{array}{l} f_1\\ f_7\\ f_8\\ f_9\\ f_{10}\\ f_{11} \end{array}
\left[\begin{array}{lll}
\text{PRED} & & \text{`eat}\langle\text{SUBJ,OBJ}\rangle\text{'} \\
\text{SUBJ} & \begin{array}{l} f_2\\ f_3\\ f_4\\ f_5\\ f_6 \end{array} & \left[\begin{array}{ll} \text{PRED} & \text{`kid'} \\ \text{DEIXIS} & \text{DISTAL} \\ \text{DEFINITE} & + \\ \text{NUMBER} & \text{SG} \\ \text{PERSON} & 3 \end{array}\right] \\
\text{OBJ} & \begin{array}{l} f_{12}\\ f_{13}\\ f_{14} \end{array} & \left[\begin{array}{ll} \text{PRED} & \text{`cake'} \\ \text{NUMBER} & \text{SG} \\ \text{PERSON} & 3 \end{array}\right] \\
\text{TENSE} & & \text{PRESENT} \\
\text{ASPECT} & & \text{PROGRESSIVE} \\
\text{PARTICIPLE} & & \text{PRESENT}
\end{array}\right]$$

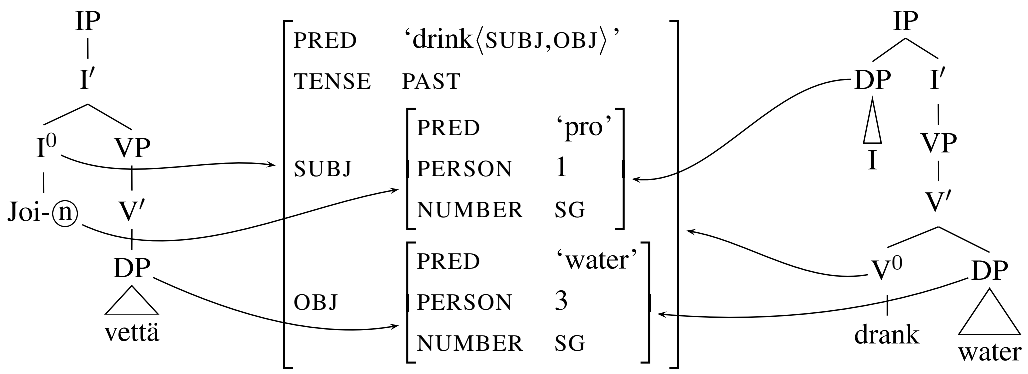
It should be noted that the features provided here reflect a specific analysis, and individual researchers may disagree on what the best analysis of a given phenomenon is. For example, we have treated the demonstrative *that* as just contributing features to the f-structure of the nominal head (*kid*). Others might propose that *that* projects to a SPEC f-structure and contains its own PRED.

4.2 Flexibility in Mapping

The mappings between c-structure and f-structure and other structures are principled and unambiguous, based on the mechanisms presented in section 4.1. However, there is cross-linguistic variation in exponence of linguistic information. For example, many languages rely more on morphology than hierarchical phrase structure in expressing syntactic information. This generalization is taken very seriously in LFG and is encapsulated in the slogan "morphology competes with syntax" (Bresnan 2001b: 6). Morphological information can be mapped directly into f-structure and there is thus no need to assume that all languages have the same, or similar, c-structure at some underlying level. In order to posit a highly articulated phrase structure for a given language, there must be evidence for such a structure. If a language expresses a grammatical function with a bound morpheme, the information is mapped directly from that morpheme onto the f-structure function: there is thus no need to posit an empty c-structure node for the grammatical function. Similarly, morphosyntactic information that is contributed by functional projections in other theories can be directly contributed morphologically in LFG.

Examples of cross-linguistic differences in c-structural expression abound. A pronominal subject may be expressed as an independent DP in some languages and a bound morpheme in others. Tense information is hosted by V^0 in some languages and I^0 in others, and in some languages it can be hosted by either I^0 or V^0. There is nothing about the mapping algorithm or the theory of c-structure that prohibits such c-structural differences between languages. Comparing two sentences with similar meanings in two different languages, the f-structures will look similar or identical and the c-structures may look radically different. Furthermore, f-structure information may be contributed simultaneously from different nodes in c-structure. In (67) we see an illustration of these points: the Finnish c-structure on the left side and the English c-structure on the right side map to the same f-structure:[8]

(67)



In sum, radically different c-structures may map to f-structures that are identical or near-identical.

A language often has more than one way to express the same function. For example, Finnish has c-structurally independent subjects in addition to the morphologically bound pronominal subjects (compare examples (9) and (67)). Also, compare the two English examples in (68):

(68) a. Hanna poured out the milk.

b. Hanna poured the milk out.

The word *out* has the same basic function in (68a) and (68b). However, the phrase structural realization is different, as evidenced by the basic fact that the word order differs, but also by the observation that

[8] This is a slight oversimplification. F-structures expressing the same basic relations in two languages may contain certain differences. For example, languages can differ in the tense and aspect distinctions they make, whether they mark evidentiality, case marking, etc.

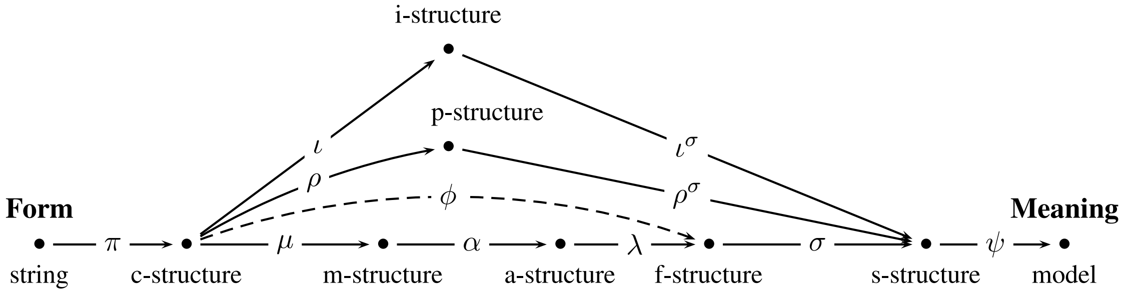


Figure 1: The Correspondence Architecture (Asudeh 2006)

out in (68a) cannot have a complement or be modified, whereas *out* in (68b) can (for references and LFG analyses of the verbal particles in several Germanic languages, see Toivonen (2003)). The key point with respect to the examples in (68) is that their f-structural representation would be the same, while their c-structures differ.

The flexibility in mapping between c-structure and f-structure renders unnecessary highly abstract phrase structure representations that contain empty categories and functional projections hosting tense, aspect, case and other functional information. Instead, c-structural representations are faithful to the word order and constituency of the sentences they model. The theory of c-structure is very much a 'what you hear is what you get' theory of surface syntax.

5 The Correspondence Architecture

The two structures, c-structure and f-structure, and correspondence function, ϕ, that we have examined so far constitute the original architecture of LFG, as laid out by Kaplan and Bresnan (1982). This architecture was subsequently generalized (Kaplan 1987, 1989, Halvorsen and Kaplan 1988) and the resulting architecture became known as the Parallel Projection Architecture or Correspondence Architecture.

The essential insight behind the Correspondence Architecture is that it is possible to resolve the apparent contradiction between, on the one hand, the empirically motivated proliferation of levels of representation and the resulting rich array of structures and constraints, and, on the other hand, formal elegance and theoretical parsimony (Kaplan 1987: 363). The resolution is accomplished as follows in the architecture. The notion of correspondence function is generalized from the ϕ function to include a number of other functions relating other structures. A rich set of structures and correspondences can be posited as constituting the linguistic form–meaning relation. However, since the correspondence functions are functions in the mathematical sense, they can be composed into larger functions. Thus, despite the linguistic richness they offer, the correspondences are mathematically and computationally eliminable (Kaplan 1987: 363).

Kaplan (1987, 1989) suggests a programmatic version of the architecture, but the first theoretically well-developed version of the architecture added semantic structure (abbreviated alternatively as s-structure or sem-structure; Halvorsen and Kaplan 1988, Dalrymple 1993). Semantic structure then formed part of the basis for Glue Semantics, a theory of the syntax–semantics interface and semantic composition (Dalrymple et al. 1993, Dalrymple 1999, 2001, Asudeh 2004, Lev 2007, Kokkonidis 2008). Glue Semantics has become the predominant semantic theory for LFG, but is actually an independent theory that could in principle be integrated with other syntactic theories; for example, Asudeh and Crouch (2002) define Glue Semantics for Head-Driven Phrase Structure Grammar and Frank and van Genabith (2001) for Lexicalized Tree-Adjoining Grammar. Asudeh (2006) considers questions of semantic composition in light of the Correspondence Architecture and provides a version of the architecture that incorporates numerous proposals in the LFG literature subsequent to the addition of semantic structure. Asudeh's presentation of the Correspondence Architecture is shown in figure 1.

Let us examine this version of the architecture briefly. There is an explicit correspondence, π, between the string and the c-structure, as proposed by Kaplan (1987, 1989). An alternative theory of the string to c-structure mapping is pursued by Wescoat (2002, 2005, 2007) in a theory of *lexical sharing* that defines a way for words to project to more than one terminal node (interestingly, without changing the formal status of c-structures as trees). Information structure (i-structure; Butt and King 2000b) encodes notions like discourse topic and focus and old and new information. Phonological structure (p-structure; Butt and King 1998a, O'Connor 2006) models phrasal phonology and prosody (a more accurate name might in fact be prosodic structure). Mycock (2006) uses p-structure in her analysis of *wh*-in-situ phenomena, which uses the correspondence architecture to account for these in terms of p- and f-structure rather than positing covert movement or empty c-structure positions. Information structure and phonological structure have both been proposed as projections from c-structure. Argument structure (a-structure; Butt et al. 1997) has been proposed for modelling semantic role information. Morphological structure (m-structure; Butt et al. 1996, 1999, Frank and Zaenen 2002, Sadler and Spencer 2004) has been proposed as an interface between syntax and morphology to capture in a more principled manner information that might otherwise be included in f-structure (e.g., tense-aspect information from auxiliaries). There has been some debate over the proper location for m-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. We assume, following Asudeh (2006), that morphological information should feed both argument structure and functional structure and therefore place m-structure between c-structure and a-structure. The resulting architecture demonstrates Kaplan's point about function composition. The original ϕ function of Kaplan and Bresnan (1982) is the composition of μ, α and λ: $\phi = \lambda \circ \alpha \circ \mu$. Lastly, we note that the mapping ψ from semantic structure to meaning is assumed to be characterized by proofs in Glue Semantics; see Asudeh (2006) for more details.

6 Some Recent Developments

Optimality-Theoretic LFG (OT-LFG) is a relatively recent outgrowth of the theory that uses LFG as the GEN component in an Optimality Theory (OT; Prince and Smolensky 1993, 2004) syntax. Parts of the constraints in the EVAL component in OT-LFG are also stated using formal notions from LFG. This extension of LFG was launched by Bresnan (1997, 2001a, 2000). It has been pursued in numerous publications in the proceedings of the annual LFG conference. Some other major works on OT-LFG are Morimoto (2000), Lee (2001), Sells (2001a,b), Kuhn (2003), and Clark (2004). An interesting recent development has seen OT-LFG applied to explaining dialect variation (Bresnan et al. 2007). Lastly, Optimality Theory has also influenced computational work on LFG, where the OT-inspired notion of *optimality marks* (Frank et al. 1998) is used for robustness of parsing and control of generation (Butt et al. 1999: 199–204). However, this latter application of OT stops short of OT-LFG's tight integration of the two theories; rather, a simple OT-inspired preference mechanism is overlaid on an LFG grammar to guide the grammar's parsing and generation.

Computational work on LFG continues to be a vital research area. There are several noteworthy research programs; here we identify just three. The Parallel Grammar project (ParGram; `http://www2.parc.com/isl/groups/nltt/pargram/`) is a collaborative international effort that seeks to develop implemented wide coverage LFG grammars based on a common inventory of f-structure features, with the goal of ensuring substantial commonality of f-structures (Butt et al. 1999, 2002). This collaborative activity not only has the consequence of testing and developing typological aspects of LFG, it also provides important insights and resources for machine translation. A recent off-shoot of ParGram is the Parallel Semantics project (ParSem), which seeks to develop semantic structures for the grammars in the ParGram project. ParSem is strongly influenced by the second computational trend: inference of semantic representations from f-structures. This approach to semantics is often called Transfer Semantics, because the aim is to transfer relevant predicate-argument relations encoded in informationally rich 'packed f-structures' to (packed) semantic representations in a computationally efficient manner (Crouch 2005, 2006, Crouch and King 2006). Transfer Semantics is an important component in industrial applications, such as the Powerset search engine. A third trend in computational work is research on automatic induction of LFG grammars (Cahill et al. 2005, O'Donovan et al. 2005, Cahill et al. 2008).

7 Concluding Remarks

LFG differs from other syntactic theories in its adoption of formally and conceptually distinct syntactic structures (c-structure and f-structure). Although Relational Grammar has a structure that is similar to f-structure in that it models grammatical functions, it does not articulate a theory of constituent structure. Head-Driven Phrase Structure Grammar represents constituency and grammatical functions — indeed, all grammatical information — in a single formal structure. Principles & Parameters Theory does not acknowledge grammatical functions as such at all, attempting to derive them from phrase structure, which is the representation used to model all syntactic information.

In addition to grammatical modularity, another underlying principle of LFG theory is that grammatical information grows *monotonically* (Bresnan 2001b: chapter 5), i.e. in an information-preserving manner. For example, as an f-description grows in size through the addition of new defining equations, the minimal f-structure that models the description also grows in size, becoming increasingly specific. Addition of constraining equations and other constraints similarly does not remove information, but rather constrains the existing minimal model. Growth of an f-description never results in information loss. This has a number of further consequences. One general consequence is that there can be no destructive operations in syntax. For example, relation changing operations, such as passive, cannot be syntactic, because that would require destructive remapping of grammatical functions. Another general consequence is that grammatical information of parts of linguistic expressions are preserved in the grammatical information of the whole. This in turn means that the parts can form informative fragments (Bresnan 2001b: 79–81). Fragments are an important part of LFG's robustness for computational parsing, since parts of ungrammatical sentences are often grammatical, and these grammatical parts can be returned in a set of wellformed fragments (Crouch et al. 2008). Cognitive aspects of fragments have also been explored, in a psycholinguistic model of human parsing and production (Asudeh 2004: chapter 8).

LFG is unique in its popularity both among computational linguists, who investigate and capitalize on formal and algorithmic properties of LFG grammars, and among descriptive and documentary linguists, who use the theory as a tool to understand and document understudied languages. We have already mentioned some of the research in computational linguistics and grammar engineering that relies on and develops LFG grammars and theory. LFG's usefulness for language description is summarized aptly by Kroeger (2007):

> LFG has a number of features that make it an attractive and useful framework for grammatical description, and for translation. These include the modular design of the system, the literal representation of word order and constituency in c-structure, a typologically realistic approach to universals (avoiding dogmatic assertions which make the descriptive task more difficult), and a tradition of taking grammatical details seriously. (Kroeger 2007: 1)

Last, but not least, the third group of researchers who have adopted LFG are traditional theoretical linguists. The characteristics that Kroeger lists above are also useful for theoretical analysis and have resulted in substantial insights into natural language. Also, many theoretical linguists find it useful that there are computational tools available to implement and test new theoretical claims. This is further facilitated by the fact that the major computational implementation, the XLE grammar development platform (Crouch et al. 2008), reflects LFG theory directly. In other words, the implementation and the theory are congruent, rather than the XLE implementing some ad hoc version of the theory.

The correspondence architecture of LFG has also proven useful for purposes that the main architects perhaps had not anticipated. For example, it offers an excellent framework for analyzing historical change (Vincent 2001). The framework allows us to pose and answer questions such as: What is the nature of the change: Is the change morphological? C-structural? F-structural? Does the change concern a specific type of linguistic information, or does the change concern the mapping between different types of information? A further advantage of LFG is its explicit and detailed representation of lexical information as lexical features. A small change in lexical information can have major syntactic consequences. Thus, both synchronic and diachronic variation can be readily represented as lexical variation. LFG has been used to model historical change by Allen (1995), and others (see, e.g., the collection of papers in Butt and King 2001b).

Further Resources

Dalrymple (2001) is a standard reference work on LFG that reviews and develops the formal theory in considerable detail against a wide-ranging empirical backdrop. Bresnan (2001b) is an advanced textbook on LFG that also introduces certain theoretical innovations; the second edition is currently in preparation (Bresnan et al., in prep.). Two introductory textbooks are Falk (2001) and Kroeger (2004). Butt et al. (1999) is an introduction to grammar engineering with LFG grammars in XLE, although there have been many subsequent developments since its publication. The authoritative source for the Xerox Linguistic Environment is the included documentation (Crouch et al. 2008). XLE is not currently open source or freely available, but a free educational license may be obtained from the NLTT group at PARC. Bresnan (1982a), Dalrymple et al. (1995) and Butt and King (2006) are collections of many of the seminal early papers on LFG. Numerous monographs and edited volumes on LFG are published by CSLI Publications, who also publish online the proceedings of the annual LFG conference (`http://csli-publications.stanford.edu/site/ONLN.shtml`); the proceedings are freely available. Lastly, there is an LFG web page that serves as a general portal (`http://www.essex.ac.uk/linguistics/external/LFG/`).

A Example: Unbounded Dependency, Adjuncts, Raising, Control

(69) What did the strange, green entity seem to try to quickly hide?

(70) **Lexicon**

what, D^0 $(\uparrow \text{PRED}) = \text{'pro'}$
$(\uparrow \text{PRONTYPE}) = \text{WH}$
$((\text{FOCUS} \uparrow)\ \text{MOOD}) =_c \text{INTERROGATIVE}$

did, I^0 $(\uparrow \text{TENSE}) = \text{PAST}$
$(\uparrow \text{MOOD}) = \text{DECLARATIVE}$
$(\uparrow \text{VFORM}) =_c \text{BASE}$
$\vee$
C^0 $(\uparrow \text{TENSE}) = \text{PAST}$
$(\uparrow \text{MOOD}) = \text{INTERROGATIVE}$
$(\uparrow \text{VFORM}) =_c \text{BASE}$

the, D^0 $(\uparrow \text{DEFINITE}) = +$

strange, A^0 $(\uparrow \text{PRED}) = \text{'strange'}$

green, A^0 $(\uparrow \text{PRED}) = \text{'green'}$

entity, N^0 $(\uparrow \text{PRED}) = \text{'entity'}$
$(\uparrow \text{NUMBER}) = \text{SG}$
$(\uparrow \text{PERSON}) = 3$

seem, V^0 $(\uparrow \text{PRED}) = \text{'seem}\langle\text{XCOMP}\rangle\text{SUBJ'}$
$(\uparrow \text{SUBJ}) = (\uparrow \text{XCOMP SUBJ})$
$\{\ \neg[\ (\uparrow \text{SUBJ NUMBER}) = \text{SG}$
$(\uparrow \text{SUBJ PERS}) = 3\ ]$
$(\uparrow \text{TENSE}) = \text{PRESENT}\ |$
$(\uparrow \text{VFORM}) = \text{BASE}\ \}$

to, I^0 $\neg(\uparrow \text{TENSE})$

try, V^0 $(\uparrow \text{PRED}) = \text{'try}\langle\text{SUBJ,XCOMP}\rangle\text{'}$
$(\uparrow \text{SUBJ}) = (\uparrow \text{XCOMP SUBJ})$
$\{\ \neg[\ (\uparrow \text{SUBJ NUMBER}) = \text{SG}$
$(\uparrow \text{SUBJ PERS}) = 3\ ]$
$(\uparrow \text{TENSE}) = \text{PRESENT}\ |$
$(\uparrow \text{VFORM}) = \text{BASE}\ \}$

quickly, Adv^0 $(\uparrow \text{PRED}) = \text{'quickly'}$

hide, V^0 $(\uparrow \text{PRED}) = \text{'hide}\langle\text{SUBJ,OBJ}\rangle\text{'}$
$\{\ \neg[\ (\uparrow \text{SUBJ NUMBER}) = \text{SG}$
$(\uparrow \text{SUBJ PERS}) = 3\ ]$
$(\uparrow \text{TENSE}) = \text{PRESENT}\ |$
$(\uparrow \text{VFORM}) = \text{BASE}\ \}$

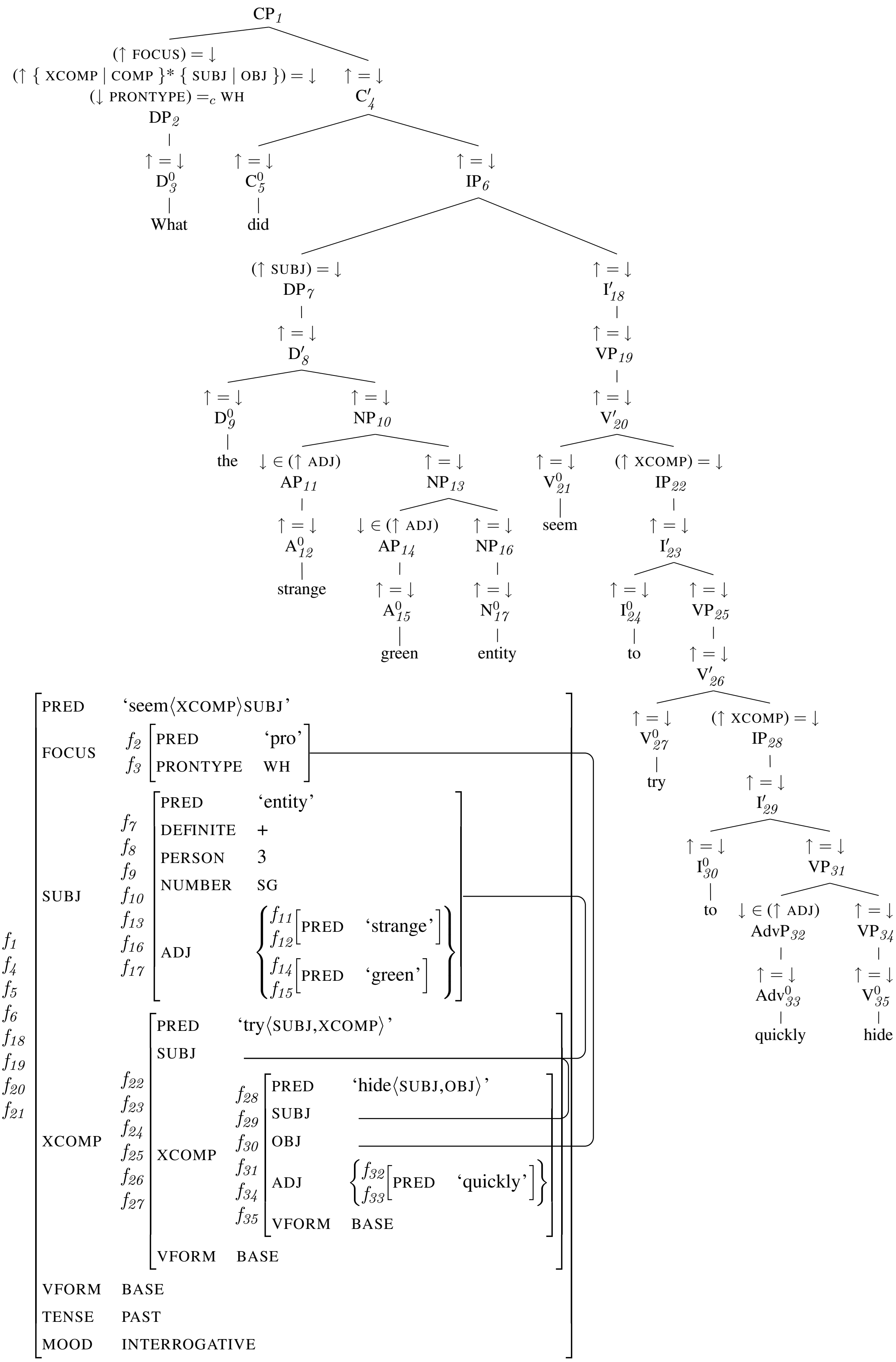


Figure 2: C-structure and f-structure for (69)

References

Allen, Cynthia. 1995. *Case-Marking and Reanalysis: Grammatical Relations from Old to Early Modern English*. Oxford: Clarendon Press.

Alsina, Alex. 1996. *The Role of Argument Structure in Grammar*. Stanford, CA: CSLI Publications.

Andrews, Avery. 1990. Unification and Morphological Blocking. *Natural Language and Linguistic Theory* 8(4): 507–557.

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

—. 2006. Direct Compositionality and the Architecture of LFG. In Butt et al. 2006, 363–387.

Asudeh, Ash, and Richard Crouch. 2002. Glue Semantics for HPSG. In Frank van Eynde, Lars Hellan, and Dorothee Beermann, eds., *Proceedings of the 8th International HPSG Conference*, 1–19. Stanford, CA: CSLI Publications.

Asudeh, Ash, Mary Dalrymple, and Ida Toivonen. 2008. Constructions with Lexical Integrity: Templates as the Lexicon-Syntax Interface. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG08 Conference*. Stanford, CA: CSLI Publications.

Berman, Judith. 2003. *Clausal Syntax of German*. Stanford, CA: CSLI Publications.

Blackburn, Patrick, and Claire Gardent. 1995. A Specification Language for Lexical Functional Grammars. In *Proceedings of the Seventh Conference of the European Chapter of the Association for Computational Linguistics*, 39–44. San Francisco: Morgan Kaufmann.

Bresnan, Joan. 1977. Transformations and Categories in Syntax. In Robert E. Butts and Jaakko Hintikka, eds., *Basic Problems in Methodology and Linguistics. Part Three of the Proceedings of the Fifth International Congress of Logic, Methodology and Philosophy of Science*.

—. 1978. A Realistic Transformational Grammar. In Morris Halle, Joan Bresnan, and George A. Miller, eds., *Linguistic Theory and Psychological Reality*, 1–59. Cambridge, MA: MIT Press.

—. 1982a. Control and Complementation. *Linguistic Inquiry* 13: 343–434.

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

Bresnan, Joan. 1982c. The Passive in Lexical Theory. In Bresnan 1982b, 3–86.

—. 1995. Linear Order, Syntactic Rank, and Empty Categories: On Weak Crossover. In Dalrymple et al. 1995, 241–274.

—. 1997. The Emergence of the Unmarked Pronoun: Chichewâ Pronominals in Optimality Theory. In Ashlee C. Bailey, Kevin E. Moore, and Jeri L. Moxley, eds., *Proceedings of the Berkeley Linguistics Society 23. Special Session on Syntax and Semantics in Africa*. Berkeley, CA: Berkeley Linguistics Society.

—. 1998. Morphology Competes with Syntax: Explaining Typological Variation in Weak Crossover Effects. In Pilar Barbosa, Danny Fox, Paul Hagstrom, Martha McGinnis, and David Pesetsky, eds., *Is the Best Good Enough? Optimality and Competition in Syntax*, 59–92. Cambridge, MA: MIT Press.

—. 2000. Optimal Syntax. In Joost Dekkers, Frank van der Leeuw, and Jeroen van de Weijer, eds., *Optimality Theory: Phonology, Syntax, and Acquisition*, 334–385. Oxford: Oxford University Press.

—. 2001a. Explaining Morphosyntactic Competition. In Mark Baltin and Chris Collins, eds., *Handbook of Contemporary Syntactic Theory*, 11–44. Oxford: Blackwell.

—. 2001b. *Lexical-Functional Syntax*. Oxford: Blackwell.

Bresnan, Joan, Ash Asudeh, Ida Toivonen, and Stephen Wechsler. In prep. *Lexical-Functional Syntax*. Oxford: Wiley-Blackwell, 2nd edn.

Bresnan, Joan, Ashwini Deo, and Devyani Sharma. 2007. Typology in Variation: A Probabilistic Approach to *be* and *n't* in the *Survey of English Dialects*. *English Language and Linguistics* 11(2): 301–346.

Bresnan, Joan, and Jonni M. Kanerva. 1989. Locative Inversion in Chicheŵa: A Case Study of Factorization in Grammar. *Linguistic Inquiry* 20(1): 1–50.

Bresnan, Joan, and Sam A. Mchombo. 1987. Topic, Pronoun, and Agreement in Chicheŵa. *Language* 63(4): 741–782.

—. 1995. The Lexical Integrity Principle: Evidence from Bantu. *Natural Language and Linguistic Theory* 13(2): 181–254.

Butt, Miriam. 1995. *The Structure of Complex Predicates in Urdu*. Stanford, CA: CSLI Publications.

Butt, Miriam, Mary Dalrymple, and Anette Frank. 1997. An Architecture for Linking Theory in LFG. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG97 Conference*. Stanford, CA: CSLI Publications.

Butt, Miriam, Mary Dalrymple, and Tracy Holloway King, eds. 2006. *Intelligent Linguistic Architectures: Variations on Themes by Ronald M. Kaplan*. Stanford, CA: CSLI Publications.

Butt, Miriam, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*, 1–7.

Butt, Miriam, and Tracy Holloway King, eds. 1996. *Proceedings of the LFG96 Conference*. Stanford, CA: CSLI Publications.

Butt, Miriam, and Tracy Holloway King. 1998a. Interfacing Phonology with LFG. In Butt and King 1998b.

Butt, Miriam, and Tracy Holloway King, eds. 1998b. *Proceedings of the LFG98 Conference*. Stanford, CA: CSLI Publications.

—. 2000a. *Argument Realization*. Stanford, CA: CSLI Publications.

Butt, Miriam, and Tracy Holloway King. 2000b. Null Elements in Discourse Structure. In K. V. Subbarao, ed., *Papers from the NULLS Seminar*. Delhi: Motilal Banarasidas.

Butt, Miriam, and Tracy Holloway King, eds. 2001a. *Proceedings of the LFG01 Conference*. Stanford, CA: CSLI Publications.

—. 2001b. *Time Over Matter: Diachronic Perspecitves on Morphosyntax*. Stanford, CA: CSLI Publications.

Butt, Miriam, and Tracy Holloway King. 2004. The Status of Case. In Veneeta Dayal and Anoop Mahajan, eds., *Clause Structure in South Asian Languages*, 153–198. Dordrecht: Kluwer.

Butt, Miriam, and Tracy Holloway King, eds. 2006. *Lexical Semantics in LFG*. Stanford, CA: CSLI Publications.

—. 2007. *Proceedings of the LFG07 Conference*. Stanford, CA: CSLI Publications.

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

Butt, Miriam, María-Eugenia Niño, and Frédérique Segond. 1996. Multilingual Processing of Auxiliaries within LFG. In Dafydd Gibbon, ed., *Natural Language Processing and Speech Technology: Results of the 3rd KONVENS Conference*, 111–122. Berlin: Mouton de Gruyter.

Cahill, Aoife, Michael Burke, Ruth O'Donovan, Stefan Riezler, Josef van Genabith, and Andy Way. 2008. Wide-Coverage Deep Statistical Parsing Using Automatic Dependency Structure Annotation. *Computational Linguistics* 34(1): 81–124.

Cahill, Aoife, Martin Forst, Michael Burke, Mairéad McCarthy, Ruth O'Donovan, Christian Rohrer, Josef van Genabith, and Andy Way. 2005. Treebank-Based Acquisition of Multilingual Unification Grammar Resources. *Research on Language and Computation* 3(2): 247–279.

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

—. 1965. *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

—. 1970. Remarks on Nominalization. In Roderick A. Jacobs and Peter S. Rosenbaum, eds., *Readings in English Transformational Grammar*, 184–221. Waltham, MA: Ginn and Company.

Chung, Sandra, and James McCloskey. 1987. Government, Barriers and Small Clauses in Modern Irish. *Linguistic Inquiry* 18: 173–237.

Clark, Brady Z. 2004. A Stochastic Optimality Theory Approach to Syntactic Change. Ph.D. thesis, Stanford University.

Cojocaru, Dana. 2003. *Romanian Grammar*. Durham, NC: Slavic and East European Language Research Center, Duke University.

Crouch, Richard. 2005. Packed Rewriting for Mapping Semantics to KR. In *Sixth International Workshop on Computational Semantics (IWCS-6)*, 103–114. Tilburg, Netherlands.

—. 2006. Packed Rewriting for Mapping Text to Semantics and KR. In Butt et al. 2006, 389–416.

Crouch, Richard, Mary Dalrymple, Ron Kaplan, Tracy King, John Maxwell, and Paula Newman. 2008. *XLE Documentation*. Palo Alto Research Center (PARC), Palo Alto, CA.

Crouch, Richard, and Tracy Holloway King. 2006. Semantics via F-structure Rewriting. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG06 Conference*, 145–165. Stanford, CA: CSLI Publications.

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

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

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

Dalrymple, Mary, and Ronald M. Kaplan. 2000. Feature Indeterminacy and Feature Resolution. *Language* 74(4): 759–798.

Dalrymple, Mary, Ronald M. Kaplan, and Tracy Holloway King. 2001. Weak Crossover and the Absence of Traces. In Butt and King 2001a, 66–82.

—. 2007. The Absence of Traces: Evidence from Weak Crossover. In Annie Zaenen, Jane Simpson, Tracy Holloway King, Jane Grimshaw, Joan Maling, and Christopher Manning, eds., *Architectures, Rules, and Preferences: Variations on Themes by Joan W. Bresnan*. 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, John Lamping, and Vijay Saraswat. 1993. LFG Semantics via Constraints. In *Proceedings of the Sixth Meeting of the European ACL*, 97–105. European Chapter of the Association for Computational Linguistics, University of Utrecht.

Embick, David, and Rolf Noyer. 2007. Distributed Morphology and the Syntax–Morphology Interface. In Gillian Ramchand and Charles Reiss, eds., *The Oxford Handbook of Linguistic Interfaces*, 289–324. Oxford: Oxford University Press.

Falk, Yehuda. 2001. *Lexical-Functional Grammar: An Introduction to Parallel Constraint-Based Syntax*. Stanford, CA: CSLI Publications.

—. 2006. *Subjects and Universal Grammar: An Explanatory Theory*. Cambridge: Cambridge University Press.

Farrell, Patrick. 2005. *Grammatical Relations:*. Oxford: Oxford University Press.

Fillmore, Charles. 1988. The Mechanisms of 'Construction Grammar'. In *Proceedings of the 14th Annual Meeting of the Berkeley Linguistics Society*, 35–55. Berkeley: University of California, Berkeley.

Frank, Anette, Tracy Holloway King, Jonas Kuhn, and John T. Maxwell, III. 1998. Optimality Theory Style Constraint Ranking in Large-Scale LFG Grammars. In Butt and King 1998b.

Frank, Anette, and Josef van Genabith. 2001. LL-based Semantics Construction for LTAG — And What It Teaches Us About the Relation Between LFG and LTAG. In Butt and King 2001a, 104–126.

Frank, Anette, and Annie Zaenen. 2002. Tense in LFG: Syntax and Morphology. In Hans Kamp and Uwe Reyle, eds., *How do we say WHEN it happens: Contributions to the theory of temporal reference in natural language*. Tübingen: Niemeyer.

Gazdar, Gerald. 1981. Unbounded Dependencies and Coordinate Structure. *Linguistic Inquiry* 12: 155–184.

Gazdar, Gerald, Ewan Klein, Geoffrey K. Pullum, and Ivan A. Sag. 1985. *Generalized Phrase Structure Grammar*. Cambridge, MA: Harvard University Press.

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago: University of Chicago Press.

Halle, Morris, and Alec Marantz. 1993. Distributed Morphology and the Pieces of Inflection. In Kenneth Hale and Samuel Jay Keyser, eds., *The View from Building 20*, 111–176. Cambridge, MA: MIT Press.

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*, 1116–1122. Institute for New Generation Systems, Tokyo. Reprinted in Dalrymple et al. (1995: 279–292).

Jackendoff, Ray. 1977. *$\overline{X}$ Syntax: A Study of Phrase Structure*. Cambridge, MA: MIT Press.

—. 1997. *The Architecture of the Language Faculty*. Cambridge, MA: MIT Press.

—. 2002. *Foundations of Language: Brain, Meaning, Grammar, Evolution*. Oxford: Oxford University Press.

Kaplan, Ronald M. 1987. Three Seductions of Computational Psycholinguistics. In Peter Whitelock, Mary McGee Wood, Harold L. Somers, Rod Johnson, and Paul Bennett, eds., *Linguistic Theory and Computer Applications*, 149–181. London: Academic Press. Reprinted in Dalrymple et al. (1995: 339–367).

—. 1989. The Formal Architecture of Lexical-Functional Grammar. In Chu-Ren Huang and Keh-Jiann Chen, eds., *Proceedings of ROCLING II*, 3–18. Reprinted in Dalrymple et al. (1995: 7–27).

Kaplan, Ronald M., and Joan Bresnan. 1982. Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Bresnan 1982b, 173–281. Reprinted in Dalrymple et al. (1995: 29–135).

Kaplan, Ronald M., and John T. Maxwell, III. 1988. Constituent Coordination in Lexical-Functional Grammar. In *Proceedings of COLING-88*, vol. 1. Budapest. Reprinted in Dalrymple et al. (1995: 199–214).

Kaplan, Ronald M., and Annie Zaenen. 1989. Long-Distance Dependencies, Constituent Structure, and Functional Uncertainty. In Mark Baltin and Anthony Kroch, eds., *Alternative Conceptions of Phrase Structure*, 17–42. Chicago, IL: University of Chicago Press. Reprinted in Dalrymple et al. (1995: 137–165).

Kay, Paul, and Charles Fillmore. 1999. Grammatical Constructions and Linguistic Generalization: The *What's X doing Y?* Construction. *Language* 75(1): 1–33.

Kayne, Richard S. 1984. *Connectedness and Binary Branching*. Dordrecht: Foris.

—. 1994. *The Antisymmetry of Syntax*. Cambridge, MA: MIT Press.

King, Tracy Holloway. 1995. *Configuring Topic and Focus in Russian*. Stanford, CA: CSLI Publications.

Kleene, Stephen C. 1956. Representation of Events in Nerve Nets and Finite Automata. In Claude Shannon and John McCarthy, eds., *Automata Studies*, 3–41. Princeton, NJ: Princeton University Press.

Kokkonidis, Miltiadis. 2008. First-Order Glue. *Journal of Logic, Language and Information* 17(1): 43–68.

Kroeger, Paul. 1993. *Phrase Structure and Grammatical Relations in Tagalog*. Stanford, CA: CSLI Publications.

—. 2004. *Analyzing Syntax: A Lexical-Functional Approach*. Cambridge: Cambridge University Press.

—. 2007. LFG as a Framework for Descriptive Grammar. In Butt and King 2007.

Kuhn, Jonas. 2003. *Optimality-Theoretic Syntax: A Declarative Approach*. Stanford, CA: CSLI Publications.

Lee, Hanjung. 2001. Optimization in Argument Expression and Interpretation: A Unified Approach. Ph.D. thesis, Stanford University.

Lev, Iddo. 2007. Packed Computation of Exact Meaning Representations. Ph.D. thesis, Stanford University.

Levin, Lori S. 1985. Operations on Lexical Forms: Unaccusative Rules in Germanic Languages. Ph.D. thesis, MIT, Cambridge, MA. Reprinted as Levin (1988).

—. 1988. *Operations on Lexical Forms: Unaccusative Rules in Germanic Languages*. New York: Garland.

Maxwell, John T., III, and Ronald M Kaplan. 1991. A Method for Disjunctive Constraint Satisfaction. In Masaru Tomita, ed., *Current Issues in Parsing Technology*, 173–190. Dordrecht: Kluwer. Reprinted in Dalrymple et al. (1995: 381–401).

—. 1993. The Interface Between Phrasal and Functional Constraints. *Computational Linguistics* 19(4): 571–590. Reprinted in Dalrymple et al. (1995: 403–429).

—. 1996. Unification-Based Parsers that Automatically Take Advantage of Context Freeness. In Butt and King 1996.

Maxwell, John T., III, and Christopher D. Manning. 1996. A Theory of Non-Constituent Coordination Based on Finite-State Rules. In Butt and King 1996.

McCloskey, James. 1979. *Transformational Syntax and Model Theoretic Semantics: A Case-Study in Modern Irish*. Dordrecht: Reidel.

McCloskey, James, and Kenneth Hale. 1984. On the Syntax of Person-Number Inflection in Modern Irish. *Natural Language and Linguistic Theory* 1(4): 487–533.

Mohanan, K. P. 1982. Grammatical Relations and Clause Structure in Malayalam. In Bresnan 1982b, 504–589.

Morimoto, Yukiko. 2000. Discourse Configurationality in Bantu Morphosyntax. Ph.D. thesis, Stanford University.

Mycock, Louise Jane. 2006. The Typology of Constituent Questions: A Lexical-Functional Grammar Analysis of 'Wh'-Questions. Ph.D. thesis, University of Manchester.

Nordlinger, Rachel. 1998. *Constructive Case: Evidence from Australian Languages*. Stanford, CA: CSLI Publications.

O'Connor, Robert. 2006. Information Structure in Lexical-Functional Grammar: The Discourse-Prosody Correspondence. Ph.D. thesis, University of Manchester.

O'Donovan, Ruth, Michael Burke, Aoife Cahill, Josef van Genabith, and Andy Way. 2005. Large-Scale Induction and Evaluation of Lexical Resources from the Penn-II and Penn-III Treebanks. *Computational Linguistics* 31(3): 328–365.

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

Perlmutter, David M., ed. 1983. *Studies in Relational Grammar 1*. Chicago, IL: The University of Chicago Press.

Perlmutter, David M., and Paul M. Postal. 1977. Toward a Universal Characterization of Passivization. In *Proceedings of the 3rd Annual Meeting of the Berkeley Linguistics Society*. Berkeley: University of California, Berkeley. Reprinted in Perlmutter (1983: 3–29).

Pinker, Steven. 1984. *Language Learnability and Language Development*. Cambridge, MA: Harvard University Press.

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.

Prince, Alan, and Paul Smolensky. 1993. Optimality Theory: Constraint Interaction in Generative Grammar. Tech. rep., RuCCS, Rutgers University, New Brunswick, NJ. Technical Report #2.

—. 2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Oxford: Blackwell.

Sadler, Louisa, and Andrew Spencer, eds. 2004. *Projecting Morphology*. Stanford, CA: CSLI Publications.

Sells, Peter, ed. 2001a. *Formal and Empirical Issues in Optimality-Theoretic Syntax*. Stanford, CA: CSLI Publications.

Sells, Peter. 2001b. *Structure, Alignment and Optimality in Swedish*. Stanford, CA: CSLI Publications.

Simpson, Jane. 1983. Aspects of Warlpiri Morphology and Syntax. Ph.D. thesis, MIT, Cambridge, MA.

—. 1991. *Warlpiri Morpho-Syntax: A Lexicalist Approach*. Dordrecht: Kluwer.

Toivonen, Ida. 2001. The Phrase Structure of Non-Projecting Words. Ph.D. thesis, Stanford University.

—. 2003. *Non-Projecting Words: A Case Study of Swedish Verbal Particles*. Dordrecht: Kluwer.

Travis, Lisa. 1984. Parameters and Effects of Word Order Variation. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Van Valin, Robert D., Jr. 1993. A Synopsis of Role and Reference Grammar. In Robert D. Van Valin, Jr., ed., *Advances in Role and Reference Grammar*, 1–164. Amsterdam: John Benjamins.

—. 2005. *Exploring the Syntax-Semantics Interface*. Cambridge: Cambridge University Press.

Vincent, Nigel. 2001. LFG as a Model of Syntactic Change. In Butt and King 2001b, 1–42.

Wescoat, Michael T. 2002. On Lexical Sharing. Ph.D. thesis, Stanford University.

—. 2005. English Nonsyllabic Auxiliary Contractions: An Analysis in LFG with Lexical Sharing. In Miriam Butt and Tracy Holloway King, eds., *Proceedings of the LFG05 Conference*, 468–486. Stanford, CA: CSLI Publications.

—. 2007. Preposition-Determiner Contractions: An Analysis in Optimality-Theoretic Lexical-Functional Grammar with Lexical Sharing. In Butt and King 2007, 439–459.

Zaenen, Annie. 1980. Extraction Rules in Icelandic. Ph.D. thesis, Harvard University. Reprinted as Zaenen (1985).

—. 1985. *Extraction Rules in Icelandic*. New York, NY: Garland.