ANNUAL REVIEWS

Annual Review of Linguistics

Lexical-Functional Grammar: An Overview

Kersti Börjars

Department of Linguistics and English Language, University of Manchester, Manchester M13 9PL, United Kingdom; email: k.borjars@manchester.ac.uk

Annu. Rev. Linguist. 2020. 6:155–72

The *Annual Review of Linguistics* is online at linguistics.annualreviews.org

https://doi.org/10.1146/annurev-linguistics-062419-125014

Keywords

syntactic theory, Lexical-Functional Grammar, parallel correspondence model, c-structure, f-structure, a-structure, Lexical Mapping Theory

Abstract

Lexical-Functional Grammar (LFG) is a model for the analysis of language in which different types of linguistic information are represented in separate dimensions, each with its own formalism. These dimensions are linked by mapping principles. In this article, I describe the architecture of the model and illustrate some dimensions of information and the mapping between them in more detail. I also provide an outline of the analysis of long-distance dependencies and control to illustrate the advantages of this type of model. I briefly mention some further areas where LFG has proven to be a useful tool for analysis and provide references for the reader to follow up.

1. INTRODUCTION

A crucial idea underpinning Lexical-Functional Grammar (LFG) is that a linguistic element of any size—a word, a phrase, or a sentence—is associated with different types of linguistic information, for instance, information about prosody, about category and constituency, about grammatical relations, and about semantics. The dimensions of linguistic information formalized within the LFG architecture are

- p-structure (phonology and prosody),
- m-structure (morphology),
- c-structure (constituent and category),
- f-structure (functional),
- a-structure (argument),
- s-structure (syntax–semantics interface), and
- i-structure (information).

Each dimension is represented in its own formal notation—for instance, c-structure takes the shape of labeled trees, whereas f-structure consists of sets of feature–value pairs—and mapping principles connect the different dimensions. In this review, I illustrate the workings of LFG with reference to f-structure, c-structure, and a-structure, which I address in greater detail below. I focus on these because there is general agreement on their formal representation and the mapping between them.[1]

Since dimensions are connected by mapping principles, LFG is what is known as a parallel correspondence architecture. I use the term architecture, rather than theory, because "the formal model of LFG is not a syntactic theory in the linguistic sense. Rather, it is an architecture for syntactic theory. Within this architecture, there is a wide range of possible syntactic theories and subtheories, some of which closely resemble syntactic theories within alternative architectures, and others of which differ radically from familiar approaches" (Bresnan et al. 2016, p. 39). It is a parallel correspondence model because the different dimensions of linguistic information are represented separately but are connected by well-defined principles of correspondence, or mappings. This formal robustness of LFG makes it suited to computational implementation and testing.

The ideas behind the theory as an alternative to the versions of transformational theory available at the time were first proposed by Bresnan (1978), and the first extensive work on LFG was also by Bresnan (1982a). Bresnan et al. (2016) and Dalrymple (2001) present comprehensive accounts of LFG, and Börjars et al. (2019) offer an introductory text. In this overview, I aim to provide an account that is easily accessible to readers with relatively little experience with formal frameworks for morphosyntactic analysis, as well as those familiar with other frameworks. This means that, at times, I express things somewhat informally.[2]

The structure of this review is as follows: I describe c-structure in Section 2 and f- and a-structure, as well as the mapping between them, in Section 3. In Section 4, I show how the mapping between c- and f-structure works by using basic examples illustrating mapping based

[1] Butt & King (1998) offer an early discussion of p-structure and its connection to c-structure, and Bögel et al. (2009) present a revised proposal. Butt et al. (1996) assume that m-structure is mapped from c-structure, whereas Frank & Zaenen (2002) argue that it should be mapped from f-structure [both papers are reprinted in the volume edited by Sadler & Spencer (2004)]. Dalrymple (2015) provides an overview of the role of morphology in LFG. King (1997) presents an early treatment of i-structure. O'Connor (2006) and Mycock (2007) provide accounts of i- and p-structure within the overall architecture. Dalrymple & Nikolaeva (2011, especially pp. 90–93) present a more recent discussion of i-structure. The formal language used for s-structure is generally Glue semantics (see Section 6 for references). There is no complete consensus on the details of the interaction between the dimensions in the overall architecture, but see Asudeh (2006, p. 369) for one proposal.
[2] For an excellent brief overview of LFG with more formal detail, see Asudeh & Toivonen (2009).

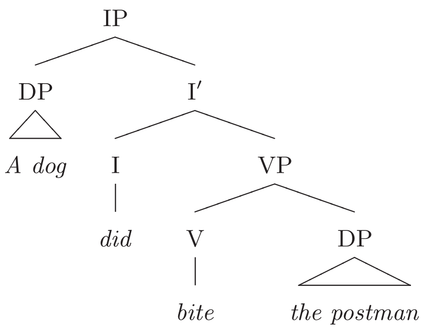
on structure (Section 4.2) and on morphology (Section 4.3). In Section 5, I use long-distance dependencies (Section 5.1) and control (Section 5.2) to illustrate the role of f-structure in LFG analyses. Finally, in Section 6, I briefly discuss and provide references for some of the areas of linguistics in which LFG has been used as a tool for linguistic analysis that offers advantages compared with other approaches.

2. C-STRUCTURE

C-structure captures information about constituency and categories. Underlying LFG c-structure is a form of X-bar syntax (Jackendoff 1977), with specifier defined as daughter of XP and sister of X′ and complement as daughter of X′ and sister of X. However, in comparison to other frameworks, LFG's approach to X-bar syntax is unorthodox in that, for instance, nonbinary branching as well as exocentric categories is permitted. The c-structure of a language is determined by constituency tests, and since the evidence for constituency varies crosslinguistically, c-structure may vary substantially between languages. English, which is a highly configurational language, has an articulated structure. The sentence in example 1 would have the structure in tree 2 (in all trees, I have excluded the X′ level for lexical categories for the sake of convenience):

(1) A dog did bite the postman.

(2)



LFG takes a restrictive approach to the use of functional categories. As discussed in Section 4.3, functional information such as case or agreement features can be mapped directly from a lexical item with appropriate morphological marking to f-structure, without mediation of a functional category in c-structure. Functional categories are generally motivated when a functional feature is associated with a structural position (e.g., Kroeger 1993, pp. 6–7; Börjars et al. 1999). In tree 2, the special positional properties associated with finite auxiliaries in English motivate the assumption of a functional category I. Although the arguments for D heading the noun phrase are not quite so clear cut, DPs are generally assumed for English. C is also a frequently used functional category in LFG; it is employed for clauses with complementizers as well as for inverted auxiliaries in English. The LFG analysis of long-distance dependencies presented in Section 5.1 includes examples of the use of a C projection.

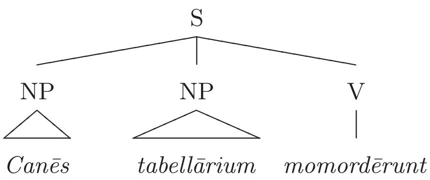
For languages that are less configurational, an exocentric category may be used. In Latin, the order between the major clausal constituents is relatively free, so that all orders would be possible for the constituents of the (invented) sentence in example 3, although some would be more frequent than others. Therefore, there is no argument in favor of assuming a VP. This leads to the c-structure in tree 4, with an exocentric category S (compare with Jøhndal 2012):[3]

[3]The glossing uses the conventions of the *Leipzig Glossing Rules* (**https://www.eva.mpg.de/lingua/pdf/Glossing-Rules.pdf**). The only slight departure from these rules is that A in example 5 refers to a verbal

Lexical Integrity Principle: morphologically complete words are leaves of the c-structure tree, and each leaf corresponds to one and only one c-structure node

(3) Canēs tabellārium momordērunt.
dog.NOM.3PL courier.ACC.3SG bite.PRF.3PL
'The dogs bit the postman.'

(4)



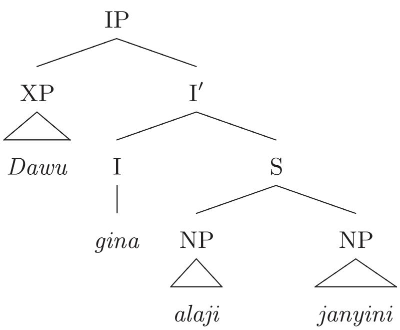
Languages can show different degrees of configurationality, and some combination of endocentric and exocentric categories may be warranted, as illustrated by example 5 from Wambaya (Nordlinger 1998, pp. 27–28). Here, word order is syntactically free, apart from the fact that the tensed auxiliary-like element *gina* has to occur in second position. I provide three examples here, but all permutations are grammatical as long as *gina* occurs in second position. Since the functional feature TENSE, here with the value PAST, is associated with the second position, a category I is assumed, but it takes an exocentric category S as its complement, resulting in the c-structure shown in tree 6 for example 5a:

(5a) Dawu gin-a alaji janyi-ni.
bite 3SG.M.A-PST boy.I(ACC) dog.I-ERG

(5b) Alaji gin-a janyi-ni dawu.
boy.I(ACC) 3SG.M.A-PST dog.I-ERG bite

(5c) Janyi-ni gin-a dawu alaji.
dog.I-ERG 3SG.M.A-PST bite boy.I(ACC)
'The dog bit the boy.'

(6)



Although the c-structure of LFG is flexible, it is generally assumed that there is a finite set of rules generating structure from which a language selects a subset.

A key principle of LFG is Lexical Integrity. This means that morphological units smaller than words cannot occur under the terminal nodes in c-structure and that the morphological structure of words is not visible to syntactic processes. Furthermore, it implies that units larger than words cannot be leaves in the c-structure tree. However, Asudeh et al. (2013) show that some multiword

marker that agrees with a transitive subject. Given the lack of articles in Latin, the idiomatic translation in example 3 is only one possibility. A similar point holds for example 5.

expressions behave like words with respect to the Lexical Integrity Principle and that an analysis of such constructions can be provided within LFG without violating the principle.

An unorthodox aspect of LFG's use of X-bar syntax is that all nodes, including preterminal and head nodes, are optional. Regarding example 1, if the auxiliary *did* had not been present, there would have been nothing to fill the I node since, even when finite, lexical verbs in English do not have the special properties that allow finite auxiliaries to occur under I. Had the sentence been *The dog bit the postman*, the I would have been deleted in tree 2, and the I′ would have dominated a VP only. Bresnan et al. (2016, p. 90) refer to this as the Principle of Economy of Expression, but see Dalrymple et al. (2015) for a discussion of the role of economy within the formal model of LFG.

3. F-STRUCTURE AND A-STRUCTURE

Unlike c-structure, f-structure is a level of linguistic information that is relatively invariant between languages. It captures syntactic predicate–argument structure in terms of grammatical functions (abbreviated as GF) such as subject and object, and it is also where information about properties such as tense and case is found. F-structure takes the form of unordered sets of feature–value pairs, also referred to as attribute–value matrices, or AVM. Three kinds of features can be distinguished on the basis of the type of value that they are associated with. There are simple features that take an atomic value, for instance, NUM = SG and TENSE = PAST. Grammatical functions such as SUBJ or OBJ are features at f-structure and take an f-structure as their value. Finally, there is the feature PRED, which takes as its value a semantic form. The semantic form appears within single quotation marks, as in PRED = 'DOG', because it is instantiated to a unique value every time it is used. Therefore, in a sentence such as *Sam*$_i$ *gave Sam*$_j$ *a dog*, there are two different instances of *Sam*, as indicated by the subscripts. Formally, the two instances have different PRED values, 'SAM$_i$' and 'SAM$_j$', but the indices are by convention generally left out. When an element subcategorizes for some function, this is indicated as part of the PRED feature within angle brackets, as in PRED = 'DEVOUR <SUBJ, OBJ>'. **Table 1** lists the three types of features, and **Table 2** presents the range of standardly assumed grammatical functions.

The first seven of the grammatical functions listed in **Table 2** are selected by some predicate and are referred to as governable grammatical functions. ADJ and XADJ are optional adjuncts, and since they can occur recursively, their value is a set of f-structures. In the two functions XCOMP and XADJ, the X indicates that they are open functions in the sense that the predicate within them is unsaturated; there is a SUBJ "missing" (see Section 5.2). TOPIC and FOCUS are grammaticalized discourse functions. These are used when a language has a structural position reserved for a specific discourse function. An example in English is *Cats, Oscar loves*, where the fronted position of *cats* makes it the FOCUS of the sentence. The SUBJ function can also be used for information-structural purposes, for instance, in active–passive alternations, and is hence a grammaticalized discourse function as well as a governable grammatical function. TOPIC and FOCUS must be linked with some

Table 1 F-structure features and values

Feature type	Value	Example
Simple feature	Atomic value	[TENSE PAST], [NUM PL], [DEF +]
Grammatical functions	F-structure	$\left[\text{SUBJ} \begin{bmatrix} \text{PRED} & \text{`MOUSE'} \\ \text{NUM} & \text{PL} \end{bmatrix}\right]$
PRED	Semantic form	[PRED 'MOUSE'] [PRED 'TICKLE <SUBJ, OBJ>']

Table 2 Grammatical functions in f-structure

Grammatical function	Explanation	Example(s)
SUBJ	Subject	Oscar drank his tea.
OBJ	Object (first)	Oscar drank his tea. Oscar gave the dog the sausage.
OBJ$_\theta$	Object (second)	Oscar gave the dog the sausage.
OBL$_\theta$	Oblique argument	Oscar placed the box in the cupboard.
POSS	Possessor	that dog's sausages
COMP	Clausal complement	Fred said that the dog liked sausages.
XCOMP	Open clausal complement	The dog appears to like sausages.
ADJ	Adjunct, nonargument	the incredibly lazy cat
		The cat slept on the back of the sofa.
		Because it was hungry, the cat ate the sausage.
XADJ	Open adjunct	Being hungry, the cat ate the sausage.
TOPIC/FOCUS	Grammaticalized discourse functions	Cats, Oscar loves. Which sausage did the dog eat?

other grammatical function. In the examples in **Table 2**, the TOPIC/FOCUS—*cats*/*which sausage*—also functions as the OBJ of each sentence (see Section 5.1).

The subcategorization information in the PRED value is derived from the element's a-structure. The a-structure takes the form of a list of the participants required for the action or event described by the predicate, as well as an indication of their semantic roles, or θ-roles, as illustrated in example 7:

(7a) bite <Agent, Patient>
(7b) place <Agent, Theme, Location>

These arguments are mapped to appropriate grammatical functions in the verb's PRED value by use of Lexical Mapping Theory (LMT).[4] An observation that underpins the mapping principles of LMT is that while there is no one-to-one correspondence between arguments and grammatical functions, there are some systematic correspondences. As illustrated in example 8, an Agent or a Location can occur as a subject or an oblique, but it cannot occur as an object. A Patient or a Theme, by contrast, can be either a subject or an object:

(8a) [A dog]$_{\text{Subj/Ag}}$ bit [the postman]$_{\text{Obj/Pat}}$.
(8b) [On the beach]$_{\text{Subj/Loc}}$ is a great place to be.
(8c) [The postman]$_{\text{Subj/Pat}}$ was bitten [by a dog]$_{\text{Obl/Ag}}$.
(8d) Oscar placed [the painting]$_{\text{Obj/Theme}}$ [above the window]$_{\text{Obl/Loc}}$.
(8e) [The painting]$_{\text{Subj/Theme}}$ was placed [on the white wall]$_{\text{Obl/Loc}}$.

LMT uses two features to capture these correspondences: [±r] (for restricted) and [±o] (for objective). Agents and Locations cannot have the function OBJ; therefore, they are associated intrinsically with the feature value [−o]. Patients and Themes, by contrast, are not restricted in this way but can be either SUBJ or OBJ; therefore, they do not have an intrinsic value for [±o] but rather have the feature [−r] intrinsically.

[4]There are slightly different versions of LMT in the literature; here I follow the one used by Dalrymple (2001), but for a different formulation, see Bresnan et al. (2016). An alternative Mapping Theory has been proposed by Kibort (see Kibort 2014 and references therein).

Table 3 Features and grammatical functions

	[−r]	[+r]
[−o]	SUBJ	OBL_θ
[+o]	OBJ	OBJ_θ

Considering the LMT features from the point of view of the functions, we see that SUBJ is quite unrestricted as to what θ-roles can fill it. Example 8 involves SUBJs that are Agent, Location, Patient, and Theme. Although an OBJ is not as unrestricted as a SUBJ, it is still relatively unrestricted. Both SUBJ and OBJ are then associated with the LMT feature [−r]. An OBL_θ, by contrast, is always assigned a specific θ-role, as indicated by the subscript θ, which is instantiated as some specific θ-role, and is therefore [+r]. The OBJ_θ is a function restricted to Theme, and it is also restricted in that not all languages use it. This gives us the LMT values in **Table 3**.

A hierarchy of θ-roles is assumed (see Bresnan & Kanerva 1989 and references therein) on the basis of typological evidence; I do not discuss the details of the hierarchy here. The relations between the four θ-roles used in example 7 are as follows:

(9) Agent >...> Theme/Patient >...> Location

Using this hierarchy, a default marking of [−r] is assigned to the highest θ-role on an element's a-structure, and all the others are assigned [+r] if this is compatible with their intrinsic feature. Applying LMT to the a-structure in example 7b yields the following mapping:

$$
\text{(10)}\quad
\begin{array}{llcccl}
\text{Place} & < & \text{Agent,} & \text{Theme,} & \text{Location} & > \\
 & & | & | & | & \\
\text{INTRINSIC} & & [-\text{o}] & [-\text{r}] & [-\text{o}] & \\
\text{DEFAULT} & & [-\text{r}] & & [+\text{r}] & \\
 & & \text{SUBJ} & \text{SUBJ/OBJ} & \text{OBL}_\theta &
\end{array}
$$

The Theme, which has [−r] intrinsically, cannot be assigned [+r]. Since it is unspecified for [±o], it could in principle be mapped to either SUBJ or OBJ, but the Function–Argument Bi-Uniqueness Condition, which constrains mapping between a-structure and f-structure, applies to disambiguate; since the Agent is SUBJ, the Theme cannot also be SUBJ but rather must be linked to OBJ. Given the argument structure for *place* in example 7, the application of LMT gives the following grammatical functions subcategorized by this verb: PRED = 'PLACE ⟨SUBJ, OBJ, OBL_θ⟩'.

We can now use the feature types introduced above to provide f-structure 11 for the sentence in example 1. Note that this f-structure is partial; more information could have been included (a more elaborate f-structure is provided in Section 4.2, below):

$$
\text{(11)}\quad
\begin{bmatrix}
\text{PRED} & \text{`BITE⟨SUBJ, OBJ⟩'} \\
\text{TENSE} & \text{PAST} \\
\text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`DOG'} \\ \text{DEF} & - \end{bmatrix} \\
\text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`POSTMAN'} \\ \text{DEF} & + \end{bmatrix}
\end{bmatrix}
$$

Function–Argument Bi-Uniqueness Condition: each a-structure role corresponds to a unique f-structure function, and each f-structure function corresponds to a unique a-structure role

The f-structure is the locus of explanation for a number of phenomena that receive a structural account in some other models, such as control, binding, and long-distance dependencies. Examples are given in Section 5; here, I illustrate the well-formedness conditions on f-structure that rule out sentences such as those in example 12. The relevant parts of their respective f-structures are presented in f-structures 13–15:

(12a) *A dog bit the postman the window cleaner.
(12b) *A dog bit.
(12c) *A dogs barked.

Coherence Condition: all argument functions must occur in the value of a local PRED feature, and all functions that have a PRED value must have a θ-role

Uniqueness Condition: every feature must have exactly one value

Completeness Condition: all argument functions that are part of the value of a PRED feature must be present in the local f-structure, and all functions that have a θ-role must themselves have a PRED value

(13) $*\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{bite}\langle\textsc{subj, obj}\rangle\text{'} \\ \textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{dog}\text{'}\end{array}\right] \\ \textsc{obj} & \left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{postman}\text{'}\end{array}\right] \\ \textsc{obj}_{\theta} & \left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{window cleaner}\text{'}\end{array}\right] \end{array}\right]$

(14) $*\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{bite}\langle\textsc{subj, obj}\rangle\text{'} \\ \textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{dog}\text{'}\end{array}\right] \end{array}\right]$

(15) $*\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{bark}\langle\textsc{subj}\rangle\text{'} \\ \textsc{subj} & \left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{dog}\text{'} \\ \textsc{num} & \textsc{sg, pl}\end{array}\right] \end{array}\right]$

The PRED feature of the main predicate in f-structure 13 requires two grammatical functions, SUBJ and OBJ, but its local f-structure contains three, SUBJ, OBJ, and $\textsc{obj}_{\theta}$. This means that the f-structure violates the Coherence Condition on f-structures. In f-structure 14, the problem is, in a sense, the reverse: The predicate requires two grammatical functions, but there is only one, the SUBJ, in the local f-structure. This renders the f-structure incomplete. Finally, in f-structure 15, the feature NUM has two incompatible values, SG from *a* and PL from *dogs*, which violates the Uniqueness Condition (how the features from the two elements end up in the same place in the f-structure is discussed in Section 4.2, below).

The second parts of the Coherence and Completeness Conditions, which establish a link between a θ-role and the presence of a value for PRED, require further explanation. Some predicates may require a grammatical function that is not mapped from any of its arguments and hence does not have a θ-role, and there are some noun phrases that cannot take a θ-role. *It* and *there* are two such elements in English, illustrated in the following examples with nonthematic subjects:

(16a) It thundered loudly.
(16b) There is a dog in the kitchen.

Thundered in example 16a does not require any argument, but it still requires a SUBJ; it takes a nonargument SUBJ *it*. As shown above, the grammatical functions listed inside the angle brackets in the PRED value are mapped from a-structure and are linked to θ-roles. This means that a nonargument grammatical function—that is, one that is not subcategorized by the verb—must occur outside the angle brackets, and the resulting PRED value for *thundered* is '$\textsc{thunder}\ \langle\ \rangle\ \textsc{subj}$'. The expletive *it*, which fills the subject position in example 16a, cannot occur in a position in which it is mapped to a θ-role. It does not have the required semantics, something that is captured in LFG through the absence of a PRED feature. Example 16a is then complete and coherent, since the element that lacks a PRED value is not in an argument function and hence does not have a θ-role. The sentences in example 17, by contrast, would be ruled out by the conditions on f-structure. Example 17a is incomplete because there is an element that lacks a PRED feature (the expletive *there*) in an argument function (SUBJ of *barked*). Example 17b is incoherent because an element that has a PRED feature occurs in a nonargument function (SUBJ of *thundered*):[5]

[5] I have used question marks rather than an asterisk for example 17b since in an idiomatic reading, where the weather is "personalized," some speakers may find it acceptable.

(17a) *There barked.
(17b) ??The weather thundered loudly.

4. MAPPING BETWEEN C-STRUCTURE AND F-STRUCTURE

4.1. Sources of F-Structure Information

The f-structure information associated with any phrase comes from two sources: the f-structure information associated with the lexical entries and the f-structure information associated with structural positions. The more configurational a language is, the more important the latter is. Each c-structure constituent, from words to sentences, is associated with an f-structure. The f-structure of each phrase is built up monotonically from the f-structure information associated with each constituent part, which means that information is never lost or changed as part of this process; it is only added to. The following subsections first consider how this is done in English, a highly configurational language, and then turn to the less configurational Latin.

4.2. Mapping from Structure

The simplified lexical entries required for the sentence in example 1 can be found in example 18. In all c- to f-structure mapping, the up arrow refers to the f-structure associated with the c-structure node above it, and the down arrow refers to the f-structure associated with the node itself. Each equation is a constraint on the f-structure for the phrase in which the element occurs, or more informally, it can be regarded as information being added when the lexical element is added to a phrase:

(18)			
	a	D	$(\uparrow \textsc{def}) = -$
	dog	N	$(\uparrow \textsc{pred}) = \text{‘}\textsc{dog}\text{’}$
			$(\uparrow \textsc{num}) = \textsc{sg}$
	did	I	$(\uparrow \textsc{tense}) = \textsc{past}$
	bite	V	$(\uparrow \textsc{pred}) = \text{‘}\textsc{bite} <\textsc{subj}, \textsc{obj}>\text{’}$
	the	D	$(\uparrow \textsc{def}) = +$
	postman	N	$(\uparrow \textsc{pred}) = \text{‘}\textsc{postman}\text{’}$
			$(\uparrow \textsc{num}) = \textsc{sg}$

The mapping between structural positions and f-structure is captured through annotated phrase structure rules. Annotated rules are exemplified as follows for the IP projection:

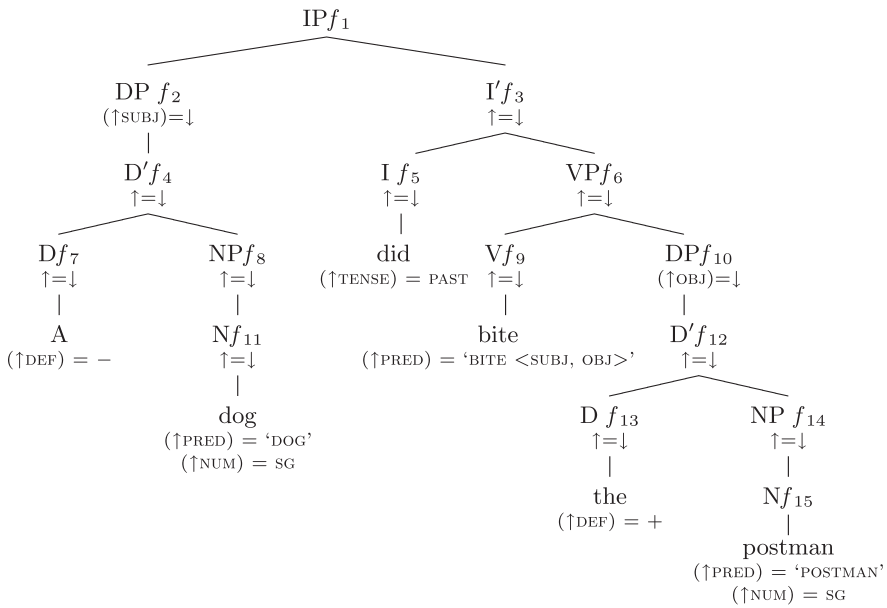
(19)
$$\text{IP} \rightarrow \underset{(\uparrow \textsc{subj})=\downarrow}{\text{DP}} \quad \underset{\uparrow=\downarrow}{\text{I}'}$$

$$\text{I}' \rightarrow \underset{\uparrow=\downarrow}{\text{I}} \quad \underset{\uparrow=\downarrow}{\text{VP}}$$

The first rule in example 19 ensures that the DP daughter of IP is associated with the SUBJ function. Informally, the annotation under DP can be read as follows: The f-structure associated with the mother of this node has a feature SUBJ, and the value of that feature is the f-structure associated with this node. The annotation $\uparrow = \downarrow$ means that the f-structure of the daughter is identical to that of the mother and hence is the functional head of the phrase. In the second rule in example 19, both daughters have this annotation; they are functional coheads and hence are associated with one and the same f-structure. How this works is described more fully below.

The full annotated tree for the sentence in example 1 is tree 20 (I have inserted arbitrary names for the f-structures associated with each node: f_1, f_2, etc.):

(20)



We can now replace the arrows with the names of the f-structures associated with each node to obtain a set of equations that define the resulting f-structure. This is the f-description of the sentence:

$(f_1\ \textsc{subj}) = f_2$	$f_4 = f_8$	$f_{12} = f_{14}$	$(f_5\ \textsc{tense}) = \textsc{past}$
$f_1 = f_3$	$f_6 = f_9$	$f_{14} = f_{15}$	$(f_9\ \textsc{pred}) = \text{‘}\textsc{bite} <\textsc{subj}, \textsc{obj}>\text{’}$
$f_2 = f_4$	$(f_6\ \textsc{obj}) = f_{10}$	$(f_7\ \textsc{def}) = -$	$(f_{13}\ \textsc{def}) = +$
$f_3 = f_5$	$f_8 = f_{11}$	$(f_{11}\ \textsc{pred}) = \text{‘}\textsc{dog}\text{’}$	$(f_{15}\ \textsc{pred}) = \text{‘}\textsc{postman}\text{’}$
$f_3 = f_6$	$f_{10} = f_{12}$	$(f_{11}\ \textsc{num}) = \textsc{sg}$	$(f_{15}\ \textsc{num}) = \textsc{sg}$
$f_4 = f_7$	$f_{12} = f_{13}$		

The resulting three equations, $f_2 = f_4 = f_7 = f_8 = f_{11}$, $f_1 = f_3 = f_5 = f_6 = f_9$, and $f_{10} = f_{12} = f_{13} = f_{14} = f_{15}$, illustrate the notion of cohead. In each of these equations there is identity between the f-structures associated with all the nodes involved; therefore, the result is a single f-structure in each case to which all component parts contribute.

These equations define f-structure 21:

(21)

$$\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{bite} < \textsc{subj}, \textsc{obj} >\text{’} \\ \textsc{tense} & \textsc{past} \\ \textsc{subj} & \left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{dog}\text{’} \\ \textsc{num} & \textsc{sg} \\ \textsc{def} & - \end{array}\right]_{f_2, f_4, f_7, f_8, f_{11}} \\ \textsc{obj} & \left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{postman}\text{’} \\ \textsc{num} & \textsc{sg} \\ \textsc{def} & + \end{array}\right]_{f_{10}, f_{12}, f_{13}, f_{14}, f_{15}} \end{array}\right]_{f_1, f_3, f_5, f_6, f_9}$$

A functional category and its complement are always coheads because they contribute to one f-structure. More generally, a number of principles are assumed to hold for the mapping between c-structure and f-structure:

i. A c-structure head and all its projections are mapped onto one f-structure,
ii. functional categories and their complements are coheads,
iii. specifiers of functional categories are grammaticalized discourse functions,
iv. complements of lexical categories are nondiscourse functions, and
v. constituents adjoined to a category by recursive rules are generally adjuncts.

Given the Uniqueness Condition, from principle ii it follows that a functional category cannot have a PRED value, or else the resulting f-structure would have two conflicting values for PRED. An element of a functional category would then contribute only simple features (see **Table 1**). An example of principle iii is introduced in Section 3; SUBJ, which is a grammaticalized discourse function (see Section 3), is defined as specifier of IP in English, whereas OBJ, which is not a discourse function, is complement of V, as captured by principle iv. An additional example of principle iii is provided in the context of long-distance dependencies in Section 5.1.

4.3. Mapping from Morphology

I now turn to Latin, where, as shown in tree 4, an exocentric clause structure is assumed in LFG. With this flat exocentric structure, functions such as SUBJ and OBJ cannot be structurally determined; the functions are not associated with specific structural positions. Instead, information about grammatical functions comes from the morphological marking on the noun phrases (i.e., dependent marking) and on the verb (head marking) (compare Nichols 1986). I consider case marking first.

The nominative marking on the subject signals subject status; the accusative case, object status. In LFG, it is thus the morphological information associated with the different forms of the nouns that contributes the SUBJ and OBJ features directly. This is done by means of lexical entries such as those in example 22:

(22)	canēs	N	($\uparrow$PRED) = 'DOG'	tabellārium	N	($\uparrow$PRED) = 'COURIER'
			($\uparrow$CASE) = NOM			($\uparrow$CASE) = ACC
			($\uparrow$NUM) = PL			($\uparrow$NUM) = SG
			(SUBJ $\uparrow$)			(OBJ $\uparrow$)

The first three equations in each of the above examples should be familiar by now, but the final lines require explanation. The constraints discussed so far provide information about feature–value pairs inside the f-structure referred to, and in this sense they work outside-in. Each final constraint in the lexical entries in example 22, by contrast, uses inside-out functional application. Taking *canēs* as an example, where this word form is inserted, there must be some f-structure that has a feature SUBJ, and the value of that feature must be the f-structure associated with the mother node of this element. It can be said to create a SUBJ function and make the f-structure associated with the mother node of *canēs* the value of that function. If we assign the name f_1 to the f-structure associated with the node above *canēs* in tree 4, so that the up arrow in the equations in the lexical entry for *canēs* in example 22 is replaced by f_1, then the equations define f-structure 23:

(23) $$\left[\text{SUBJ} \quad \left[\begin{array}{ll} \text{PRED} & \text{'DOG'} \\ \text{CASE} & \text{NOM} \\ \text{NUM} & \text{PL} \end{array} \right]_{f_1} \right]$$

Similarly, a connection is established between the accusative form *tabellārium* and the OBJ function. Nordlinger (1998) presents an early example of the use of inside-out functional application; she uses it for the constructive case analysis of the complex distribution of case found in a number of Australian languages.

Regarding verb marking, the verb form *momordērunt* has the partial lexical entry shown in example 24:

(24) momordērunt V (↑PRED) = 'BITE <SUBJ, OBJ>'
(↑SUBJ NUM) = PL
(↑SUBJ PERS) = 3

The last two equations in the lexical entry in example 24 constrain the value of the features NUM and PERS inside the SUBJ within its mother's f-structure. They establish what is referred to as a path through SUBJ NUM/PERS to the value PL/3. Rather than saying that the verb is a third-person plural verb, we say that the verb constrains its subject to be third-person plural.

In the f-structure for example 3, then, there are two separate constraints that ensure that *canēs* is the SUBJ: the nominative case associated with the noun itself, and the verb form requiring a plural subject. This kind of redundancy is common in languages.

The English and Latin examples illustrate how f-structure information from different sources can contribute to build f-structure; it can be associated with structure or with morphology. There is great crosslinguistic variation in the extent to which syntax or morphology—or some combination of the two—is used to signal grammatical functions. In LFG, this can be captured as "morphology competes with syntax" (Bresnan 1998).

5. LEXICAL-FUNCTIONAL GRAMMAR ANALYSES

This section provides brief descriptions of two phenomena in LFG, long-distance dependencies and control, because they illustrate a number of key aspects of the framework. Rather than provide a full account of the data or the analyses, I intend to give the reader some idea about how the interaction between c- and f-structure works in more complex data, as well as about the role that f-structure plays in LFG.

5.1. Long-Distance Dependencies

In this section, the *wh*-interrogative with a questioned object in example 25 is used to illustrate the analysis of long-distance dependencies:

(25) What did the dog devour?

Structurally, constructions of this type involve a finite auxiliary preceding the subject and a *wh*-phrase occurring in front of the auxiliary. Since SUBJ is found in the specifier of IP in English, the fronted elements must occur in an additional functional projection, CP. The *wh*-phrase occurs in the initial position for information-structural reasons; it is a FOCUS and hence a grammaticalized discourse function. This is, then, an example of the structure–function principle iii described in Section 4.2: The specifier of a functional category is associated with a grammaticalized discourse function. However, the *wh*-phrase must also fill the function of OBJ of the verb *devour*. C-structure rule 26 is a simplified version of the annotated phrase structure rule required to map the fronted *wh*-phrase correctly:

(26) CP → XP C′
(↑FOCUS)=↓ ↑=↓
(↑FOCUS)=(↑{COMP | XCOMP}* GF)

The first annotation under the XP ensures that this constituent is associated with the grammaticalized discourse function FOCUS. The second annotation ensures that the FOCUS also has some other grammatical function (marked GF) inside the sentence, but it is not specified which. This is referred to as outside-in functional uncertainty. In this case, the equation will be resolved as FOCUS = OBJ because that is the only way of satisfying the Completeness and Coherence Conditions, since *devour* requires an OBJ and there is no other constituent to fill this function. Because the grammatical relation with which the fronted phrase is linked can in principle be embedded inside another clause, either finite or nonfinite, a path needs to be specified between the FOCUS and the grammatical function. The use of the Kleene star in {COMP | XCOMP}* indicates that the path can go through zero or more COMPS or XCOMPS. The second equation under the XP in c-structure rule 26 also accounts for a sentence such as the following:

(27) What did Fred say that the dog seemed to have devoured?

The resulting partial f-structure for example 25 is provided in f-structure 28 (where the line indicates the shared f-structure):

(28)
$$\begin{bmatrix} \text{PRED} & \text{`DEVOUR <SUBJ, OBJ>'} \\ \text{FOCUS} & \begin{bmatrix} \text{PRED} & \text{`PRO'} \\ \text{PRONTYPE} & \text{WH} \end{bmatrix} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`DOG'} \end{bmatrix} \\ \text{OBJ} & \end{bmatrix}$$

This brief discussion illustrates how f-structure plays a key role in the account of long-distance dependencies and shows that no empty category is used in c-structure; the OBJ is present in f-structure, but there is no element in the OBJ position in c-structure. Since all categories in a phrase structure rule are assumed to be optional, the VP in the c-structure representation of example 25 simply consists of the V, although the OBJ will be there in f-structure. I have followed the analysis presented by Kaplan & Zaenen (1989), also used by Dalrymple (2001). Bresnan et al. (2016) argue that an empty category is motivated in some cases of extraction.

One thing I have not accounted for in c-structure rule 26 is how to ensure that there is a *wh*-word somewhere in the fronted phrase, given that the fronted *wh*-phrase could have been more complex: *which sausage*, *whose sausage*, or *whose dog's sausage*. Due to space constraints I do not explain how this is formalized here but instead refer the reader to Dalrymple (2001, pp. 389–408). Outside-in functional uncertainty is introduced here; inside-out functional uncertainty is also used within LFG, for instance, in the analysis of binding (see Dalrymple 2001, pp. 278–88). There are language-specific restrictions on which XPs can be fronted and which grammatical functions can be focused in this way, which can be captured by defining a subset of categories and grammatical functions—XP_{wh} and GF_{wh}—for c-structure rule 26.

5.2. Control

Table 2 describes XCOMP as an open clausal complement, such as the underlined constituent in example 29:

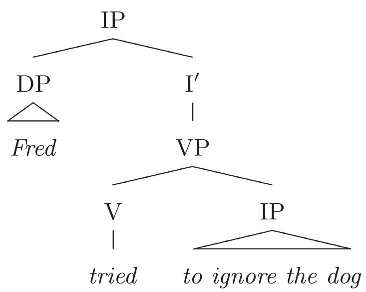
(29) Fred tried <u>to ignore the dog</u>.

The XCOMP *to ignore the dog* is open in the sense that *ignore* lacks a subject in c-structure. However, in order for the sentence not to violate the Completeness Condition, the SUBJ must be there in f-structure. There is, of course, an understood subject of *ignore*, and that is the subject of the matrix clause; the SUBJ of the XCOMP is controlled by the SUBJ of the higher clause. Since this is a property of the verb *try*, the connection is captured as part of its lexical information:

(30) try V (↑PRED) = 'TRY <SUBJ, XCOMP>'
(↑SUBJ) = (↑XCOMP SUBJ)

The second equation in example 30 is a functional control equation. It establishes identity between the value of the SUBJ of *try* and the SUBJ of its complement. The resulting c-structure is shown in tree 31, and f-structure 32 is a partial f-structure for example 29 (as in example 28, the line indicates structure sharing):

(31)



(32)

$$\begin{bmatrix} \text{PRED} & \text{`TRY <SUBJ, XCOMP>'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`FRED'} \end{bmatrix} \\ \text{XCOMP} & \begin{bmatrix} \text{PRED} & \text{`IGNORE <SUBJ, OBJ>'} \\ \text{SUBJ} & \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`DOG'} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Now consider the following superficially similar sentence:

(33) Fred seemed to ignore the dog.

Despite the similarity, there is a well-known difference between the two construction types shown in examples 29 and 33: Whereas *Fred* is assigned a θ-role by *try* in example 29, *seem* in example 33 does not assign a θ-role to its subject. This has led to the traditional terminology of control for *try* and raising for *seem*. However, in LFG, both are analyzed as functional control. The difference lies in the argument status of the subject of the main verb: It is an argument in example 29, but not in example 33. As discussed in relation to example 16 in Section 3, a grammatical function that is not linked to an a-structure argument is placed outside the angle brackets in the PRED value. This means that we obtain the lexical entry in example 34 for *seem*, yielding a c-structure and an f-structure parallel to those for *try*, but with the difference that the SUBJ of the sentence is not an argument of the matrix verb:

(34) seem V (↑PRED) = 'SEEM <XCOMP> SUBJ'
(↑SUBJ) = (↑XCOMP SUBJ)

In examples 29 and 33, the functional control equation ensures absolute identity between the two grammatical functions. They share one f-structure, and the lower subject cannot be anything but identical to the higher one. There are, however, instances such as *want* in example 35a for which this analysis is not an option. As example 35b shows, the SUBJ of the complement can be filled by an overt noun phrase:

(35a) Fred wants to ignore the dog.
(35b) Fred wants Oscar to ignore the dog.

In example 35b, the complement of *wants* cannot be an XCOMP, since the complement is not open but rather contains an overt subject. *Oscar* cannot be an object of *wants*, since the corresponding passive sentence is ungrammatical: **Oscar is wanted to ignore the dog* (*by Fred*). A further difference in behavior between *want* and *try* is that, even for a construction with a lower subject missing in c-structure, there need not be identity between the higher and the lower subject in the case of *want*, whereas for *try* identity is required:

(36a) Fred wanted to withdraw from the EU by the original deadline.
(36b) #Fred tried to withdraw from the EU by the original deadline.

The reference of the understood subject of the lower clause in example 36a is something like 'his country' rather than 'Fred'. The infelicity of example 36b follows from the functional control analysis: The two subjects have to be identical, and the meaning of the example becomes odd. The phenomenon illustrated in example 36a is referred to as partial control, and Haug (2013) provides a detailed discussion of it from an LFG perspective. The fact that the interpretation of the SUBJ of the lower clause is dependent on the context makes it resemble the anaphoric binding of a pronoun, and in LFG *want* is described as an anaphoric control verb. Its lexical entry is as in example 37, with no open complement and no control equation. When there is no overt lower-clause subject, there is a pronominal subject in f-structure, and its interpretation depends on the context. We assume that an equation such as example 38 forms a part of the grammar of English. Informally, it states that when a verb form is not finite, a pronominal SUBJ may optionally be inserted in the local f-structure. F-structure 39 is the resulting partial f-structure for example 35a:

(37) want V $(\uparrow \text{PRED}) = \text{`WANT} <\text{SUBJ, COMP}>\text{'}$

(38) $((\uparrow \text{SUBJ PRED}) = \text{`PRO'})$ where $(\uparrow \text{VFORM}) \neq \text{FIN}$

(39)
$$\begin{bmatrix} \text{PRED} & \text{`WANT} < \text{SUBJ, COMP} >\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`FRED'} \end{bmatrix} \\ \text{COMP} & \begin{bmatrix} \text{PRED} & \text{`IGNORE} < \text{SUBJ, OBJ} >\text{'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`PRO'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`DOG'} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Anaphoric control is also what we get in nonfinite adjunct clauses. Compare the following two sentences:

(40a) Eating fish is healthy.
(40b) Eating fish makes Pam feel queasy.

In both examples, *eating* lacks an overt subject. In example 40a it gets an arbitrary interpretation ('for anyone to eat fish'), whereas in example 40b the natural interpretation is that *Pam* is the subject of *eating*.

What I have presented in this subsection is the standard analysis of control within LFG, which has remained remarkably unchanged since Bresnan (1982b). For a different approach to some control constructions, see Dalrymple (2001, pp. 323–30). Andrews (1990) discusses Icelandic data that provide evidence in favor of the distinction between functional and anaphoric control.

6. AREAS OF APPLICATION

LFG is formally explicit, which makes it eminently suited to computational implementation. Indeed, the openly available Xerox Linguistic Environment (XLE) (see Crouch et al. 2011 and **http://ling.uni-konstanz.de/pages/xle/**) allows formal testing of LFG analyses. The XLE platform has been used for the Parallel Grammar Project (Butt et al. 2002); initially this project included grammars for English, French, and German, but grammars of varying sizes have been added for a number of other languages, including Arabic, Hungarian, Japanese, Norwegian, Welsh, and Wolof.

Glue semantics is an approach to semantic composition and the syntax–semantics interface that has been developed within LFG, most importantly in the articles in the volume edited by Dalrymple (1999). For an introduction and examples of applications, see Dalrymple (2001, chapter 9) and Asudeh (2012). Haug (2013) also uses Glue in his analysis, referred to in Section 5.2.

The fact that LFG does not assume that all languages rely on hierarchical structure, but rather assumes that crucial morphosyntactic information can be provided directly by the morphology, makes it an excellent tool for description and analysis of languages that are typologically different from English. Language descriptions using LFG for analysis include those by Simpson (1991), Kroeger (1993), and Nordlinger (1998). Some typological phenomena that resist analysis in many frameworks have found elegant analyses in LFG. One example, referred to above, is constructive case (Nordlinger 1998); another is when morphological marking occurs "in the wrong place," for instance, clausal tense marking on nouns, as analyzed by Nordlinger & Sadler (2004), or complex agreement behavior involving coordination, as analyzed by, for instance, King & Dalrymple (2004) and Sadler (2016).

The fact that non-one-to-one relations between different dimensions of information are expected makes LFG suitable for diachronic analysis, since change frequently happens at a different pace in different domains, leading to form–function mismatches. Börjars & Vincent (2017) provide an overview of the use of LFG in historical linguistics. Recent diachronic analyses utilizing different aspects of LFG include those by Kibort & Maling (2015), Börjars et al. (2016), and Camilleri & Sadler (2017).

DISCLOSURE STATEMENT

The author is not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

I am grateful to Nigel Vincent for his help with the Latin data. The text has benefited greatly from the comments made by an anonymous referee.

LITERATURE CITED

Andrews A. 1990. Case structures and control in modern Icelandic. In *Modern Icelandic Syntax*, ed. J Maling, A Zaenen, pp. 165–85. San Diego, CA: Academic

Asudeh A. 2006. Direct compositionality and the architecture of LFG. In *Intelligent Linguistic Architectures: Variations on Themes by Ronald M. Kaplan*, ed. M Butt, M Dalrymple, TH King, pp. 363–87. Stanford, CA: Cent. Study Lang. Inf.

Asudeh A. 2012. *The Logic of Pronominal Resumption*. Oxford, UK: Oxford Univ. Press

Asudeh A, Dalrymple M, Toivonen I. 2013. Constructions with lexical integrity. *J. Lang. Model.* 1:1–54

Asudeh A, Toivonen I. 2009. Lexical-Functional Grammar. In *The Oxford Handbook of Linguistic Analysis*, ed. B Heine, H Narrog, pp. 373–406. Oxford, UK: Oxford Univ. Press. 2nd ed.

Bögel T, Butt M, Kaplan RM, King TH, Maxwell JT III. 2009. Prosodic phonology in LFG: a new proposal. In *Proceedings of the LFG09 Conference*, ed. M Butt, TH King, pp. 146–66. Stanford, CA: Cent. Study Lang. Inf.

Börjars K, Chisarik E, Payne J. 1999. On the justification for functional categories in LFG. In *Proceedings of the LFG99 Conference*, ed. M Butt, TH King. Stanford, CA: Cent. Study Lang. Inf.

Börjars K, Harries P, Vincent N. 2016. Growing syntax: the development of a DP in North Germanic. *Language* 92:e1–37

Börjars K, Nordlinger R, Sadler L. 2019. *Lexical-Functional Grammar: An Introduction*. Cambridge, UK: Cambridge Univ. Press

Börjars K, Vincent N. 2017. Lexical-Functional Grammar. In *The Cambridge Handbook of Historical Syntax*, ed. A Ledgeway, I Roberts, pp. 642–63. Cambridge, UK: Cambridge Univ. Press

Bresnan J. 1978. A realistic transformational grammar. In *Linguistic Theory and Psychological Reality*, ed. M Halle, J Bresnan, GA Miller, pp. 1–59. Cambridge, MA: MIT Press

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

Bresnan J. 1982b. Control and complementation. *Linguist. Inq.* 13:343–92

Bresnan J. 1998. Morphology competes with syntax: explaining typological variation in weak crossover effects. In *Is the Best Good Enough? Optimality and Competition in Syntax*, ed. P Barbosa, D Fox, P Hagstrom, M McGinnis, D Pesetsky, pp. 59–92. Cambridge, MA: MIT Press

Bresnan J, Asudeh A, Toivonen I, Wechsler S. 2016. *Lexical-Functional Syntax*. Malden, MA/Oxford, UK: Wiley-Blackwell. 2nd ed.

Bresnan J, Kanerva JM. 1989. Locative inversion in Chicheŵa: a case study in the factorization of grammar. *Linguist. Inq.* 20:1–50

Butt M, Dyvik H, King TH, Masuichi H, Rohrer C. 2002. The Parallel Grammar Project. In *Proceedings of COLING-2002 Workshop on Grammar Engineering and Evaluation*, pp. 1–7. Stroudsburg, PA: Assoc. Comput. Linguist.

Butt M, King TH. 1998. Interfacing phonology with LFG. In *Proceedings of the LFG98 Conference*, ed. M Butt, TH King. Stanford, CA: Cent. Study Lang. Inf.

Butt M, King TH, Niño ME, Segond F. 1999. *A Grammar Writer's Cookbook*. Stanford, CA: Cent. Study Lang. Inf.

Butt M, Niño ME, Segond F. 1996. Multilingual processing of auxiliaries in LFG. In *Natural Language Processing and Speech Technology: Results of the 3rd KONVENS Conference*, ed. D Gibbon, pp. 111–22. Berlin: Mouton de Gruyter

Camilleri M, Sadler L. 2017. Posture verbs and aspect: a view from Vernacular Arabic. In *Proceedings of the LFG17 Conference*, ed. M Butt, TH King, pp. 167–87. Stanford, CA: Cent. Study Lang. Inf.

Crouch R, Dalrymple M, Kaplan RM, King TH, Maxwell J, Newman P. 2011. *XLE Documentation*. Palo Alto, CA: Palo Alto Res. Cent. **https://ling.sprachwiss.uni-konstanz.de/pages/xle/doc/xle_toc.html**

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

Dalrymple M. 2001. *Lexical Functional Grammar*. San Diego, CA/London: Academic

Dalrymple M. 2015. Morphology in the LFG architecture. In *Proceedings of the LFG15 Conference*, ed. M Butt, TH King, pp. 43–62. Stanford, CA: Cent. Study Lang. Inf.

Dalrymple M, Kaplan RM, King TH. 2015. Economy of Expression as a principle of syntax. *J. Lang. Model.* 3:377–412

Dalrymple M, Nikolaeva I. 2011. *Objects and Information Structure*. Cambridge, UK: Cambridge Univ. Press

Frank A, Zaenen A. 2002. Tense in LFG: syntax and morphology. In *How We Say WHEN It Happens: Contributions to the Theory of Temporal Reference in Natural Language*, ed. H Kamp, U Reyle, pp. 17–52. Tbingen, Ger.: Niemeyer

Haug D. 2013. Partial control and anaphoric control in LFG. In *Proceedings of the LFG13 Conference*, ed. M Butt, TH King, pp. 274–94. Stanford, CA: Cent. Study Lang. Inf.

Jackendoff R. 1977. *X-Bar Syntax: A Study of Phrase Structure*. Cambridge, MA: MIT Press

Jøhndal M. 2012. *Non-finiteness in Latin*. PhD Thesis, Univ. Cambridge, Cambridge, UK

Kaplan RM, Bresnan J. 1982. Lexical-Functional Grammar: a formal system for grammatical representation. See Bresnan 1982a, pp. 173–281

Kaplan RM, Zaenen A. 1989. Long-distance dependencies, constituent structure, and functional uncertainty. In *Alternative Conceptions of Phrase Structure*, ed. M Baltin, A Kroch, pp. 17–42. Chicago: Chicago Univ. Press

Kibort A. 2014. Mapping out a construction inventory with (Lexical) Mapping Theory. In *Proceedings of the LFG14 Conference*, ed. M Butt, TH King, pp. 262–82. Stanford, CA: Cent. Study Lang. Inf.

Kibort A, Maling J. 2015. Modelling the syntactic ambiguity of the active versus passive impersonal in LFG. In *Proceedings of the LFG15 Conference*, ed. M Butt, TH King, pp. 145–65. Stanford, CA: Cent. Study Lang. Inf.

King TH. 1997. Focus domains and information-structure. In *Proceedings of the LFG97 Conference*, ed. M Butt, TH King. Stanford, CA: Cent. Study Lang. Inf.

King TH, Dalrymple M. 2004. Determiner agreement and noun conjunction. *J. Linguist.* 40:69–104

Kroeger P. 1993. *Phrase Structure and Grammatical Relations in Tagalog*. Stanford, CA: Cent. Study Lang. Inf.

Mycock L. 2007. Constituent question formation and focus: a new typological perspective. *Trans. Philol. Soc.* 105:192–251

Nichols J. 1986. Head-marking and dependent-marking grammar. *Language* 62:56–119

Nordlinger R. 1998. *Constructive Case: Evidence from Australian Languages*. Stanford, CA: Cent. Study Lang. Inf.

Nordlinger R, Sadler L. 2004. Tense beyond the verb: encoding clausal tense/aspect/mood on nominal dependents. *Nat. Lang. Linguist. Theory* 22:597–641

O'Connor R. 2006. *Information structure in Lexical-Functional Grammar: the discourse-prosody correspondence*. PhD Thesis, Univ. Manchester, Manchester, UK

Sadler L. 2016. Agreement in Archi: an LFG perspective. In *Archi: Complexities of Agreement in Cross-Theoretical Perspective*, ed. O Bond, GC Corbett, M Chumakina, D Brown, pp. 150–83. Oxford, UK: Oxford Univ. Press

Sadler L, Spencer A. 2004. *Projecting Morphology*. Stanford, CA: Cent. Study Lang. Inf.

Simpson J. 1991. *Warlpiri Morpho-Syntax: A Lexicalist Approach*. Dordrecht, Neth.: Kluwer Acad.

RELATED RESOURCES

Proceedings of the annual LFG conference. **https://cslipublications.stanford.edu/LFG/**
Research within LFG. **http://ling.uni-konstanz.de/pages/home/lfg/**