Lexical functional grammar

[Published as article 3.9.20 in Neil J. Smelser and Paul Baltes (eds) 2001
International Encyclopedia of the Social and Behavioral Sciences, 8748-8754]

Peter K. Austin
Department of Linguistics and Applied Linguistics
University of Melbourne
Parkville Vic 3010
Australia

May 2001

p.austin@linguistics.unimelb.edu.au

3.9.20 Lexical Functional Grammar

1. Introduction

Lexical Functional Grammar (LFG) is a theory of language structure that deals with the syntax, morphology, and semantics of natural languages. It is distinguished from other theories by having several parallel representations for sentences, each with its own architecture and vocabulary, and subject to its own organizational constraints. The parallel representations are linked by principles of correspondence (mappings), and are not derived from one another. LFG is thus a constraint-based theory of language and is a non-derivational unification model, unlike other generative grammar models such as Principles and Parameters or Relational Grammar. There is an extensive international community of scholars who work within an LFG framework and who meet annually to discuss ideas – it has been used to describe an impressive array of languages from a large number of families and typological groups. Current developments in LFG include exploration of the role of morphology in language structure, semantic representations, and computational implementation; a number of scholars are also working on an Optimality Theory-based model of LFG.

2. Background

Lexical Functional Grammar arose in the late 1970's through the collaboration of Joan Bresnan (a linguist) and Ronald Kaplan (a computer scientist) who were dissatisfied with then current transformational models of language and were seeking a more 'realistic' approach – from its inception LFG has been concerned to be a model that is typologically grounded, computationally implementable, and consistent with psycholinguistic understanding of language acquisition and comprehension. Bresnan ed. (1982) is a collection of papers setting out the original model and describing individual languages.

As its name suggests, Lexical Functional Grammar emphasizes analysis of certain phenomena in lexical and functional terms, rather than purely in terms of phrase structure configurations (and movement of elements from one configurational position to another). A number of arguments can be put forward for separating constituency and functional representations (see Bresnan 2000, Falk 2001, Carnie 2001:339):

1. *structure-function mismatches* where there is an imperfect correspondence between constituency and function. So for example, the predicate *talked about* requires a Noun Phrase (NP) object and cannot have a tensed complement clause (CP) object:

(1) We talked about $[_{NP}$ the fact that he was unhappy $_{NP}]$ for weeks

(2) *We talked about $[_{CP}$ that he was unhappy $_{CP}]$ for weeks

However, if the CP occurs as a topic in initial position the sentence is perfectly grammatical:

(3) $[_{CP}$ That he was unhappy $_{CP}]$ we talked about for weeks

Transformational models would be required to move the CP from a post-predicate position where it cannot actually occur, while LFG is able to capture the difference in grammaticality in terms of functional differences (topic versus object – see further below);

2. there are *non-configurational languages* in which word order is free and sentences have a 'flat' structure: in these languages there is no evidence for phrasal constituency asymmetries that enable us to distinguish grammatical functions. Such non-configurational languages are typically *dependent-marking* where case morphology encodes functional information, or *head-marking* where agreement morphology serves to distinguish functions such as subject and object, for example. Jiwarli (Western Australia) is a dependent-marking non-configurational language where case marking codes function; word order is free and semantically related items can appear discontinuously in the clause, as in:

(4)	*Juma-ngku*	*ngatha-nha*	*nhanya-nyja*	*ngulu*	*walhirrkura-lu*
	child-erg	1sg-acc	see-past	that:erg	naughty-erg
	'That naughty child saw me'				

Here the subject appears split at the beginning and end of the clause (all elements being marked for the same ergative (transitive subject) case); any other ordering or reordering of these words is perfectly grammatical in Jiwarli, thus:

(5a)	*Ngulu*	*juma-ngku*	*ngatha-nha*	*nhanya-nyja*	*walhirrkura-lu*
(5b)	*Ngatha-nha*	*walhirrkura-lu*	*nhanya-nyja*	*ngulu*	*juma-ngku*
(5c)	*Nhanya-nyja*	*ngatha-nha*	*ngulu*	*juma-ngku*	*walhirrkura-lu*

all mean 'That naughty child saw me' (see Austin and Bresnan 1995, Austin 2001 for further discussion). In non-configurational languages constituency and function must be separately represented, unlike configurational languages (such as English) where the presence of a Verb Phrase constituent linking Verb and Object (but not Subject) allows function to be expressed in constituency (and word order) terms.

3. Parallel structures

Lexical Functional Grammar analyses sentences in terms of (at least) four parallel representations:

- c-structures which deal with *constituency* facts (word order and phrasal grouping) and have the form of context-free phrase structure trees of the usual X-bar theory type;
- f-structures which deal with *functional* information (grammatical functions such as SUBJect and OBJect, but also discourse functions like TOPic) and have the form of matrices of attribute-value pairs. Attributes may be morphosyntactic features such as tense or number, or grammatical functions;
- a-structures which deal with *predicate-argument* information such as the number and type of arguments of a predicate and the semantic role borne by arguments. A-structures

are stated as arrays of predicates and argument slots with associated semantic role values, such as Agent, Patient or Location;

- σ-structures which deal with *semantic* information through a deductive approach to assembling meanings based on reasoning subject to constraints. The σ-structures use a linear logic to build semantic representations from a-structure information.

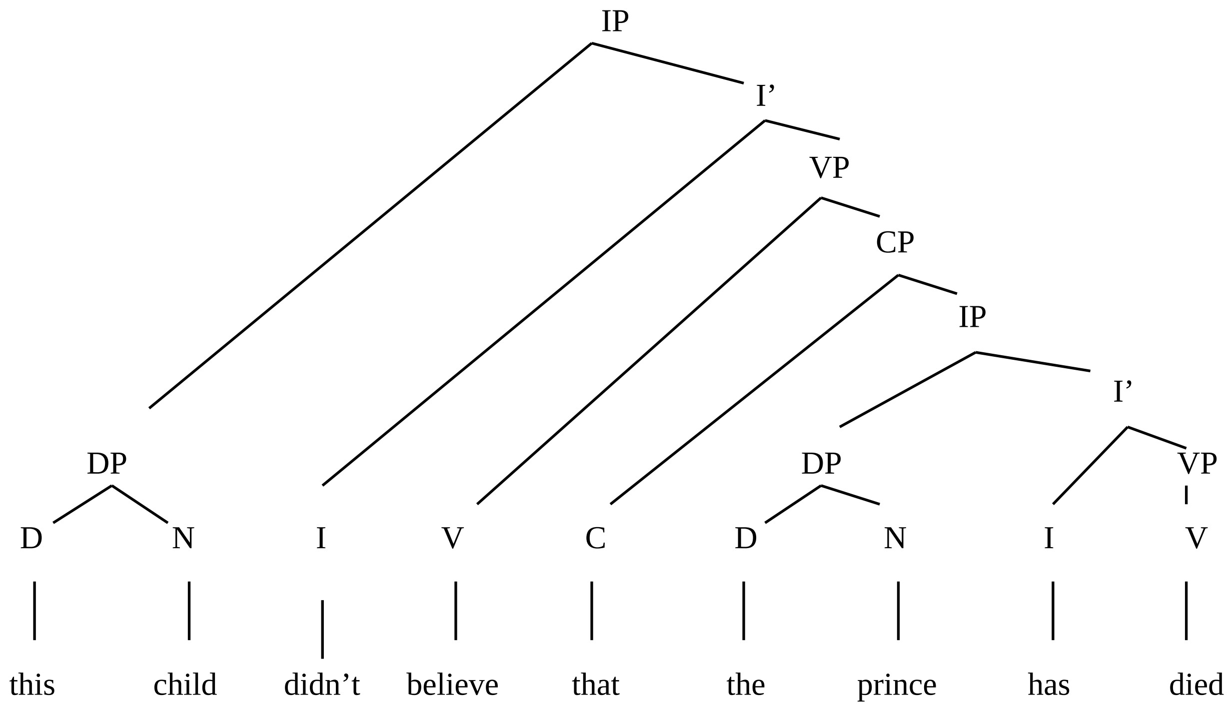
There may be other parallel representations, such as *thematic-structure* that captures discourse pragmatic information, or *phonological-structure* that deals with sounds, however they have not been elaborated in LFG work to date.

The parallel representations are linked by *correspondence principles* where information of one type can be *mapped* to other structures. Thus c-structure nodes in a configurational language map to grammatical function attributes in the corresponding f-structure; this is discussed below.

3.1 *C-structure*

Constituency information is represented as c-structures which are context free labeled phrase markers (trees) of the sort that have been commonly used in linguistics for the past fifty years. LFG c-structures adopt the X-bar model of capturing head-dependent relations, and treat ‘functional’ elements such as Determiners, Complementizers and Inflections as co-heads of lexical elements such as Nouns and Verbs. LFG c-structures however are subject to the *lexical integrity principle* which states that minimal c-structure elements are whole words, not part of words or empty categories. No movement of c-structure constituents (such as V to I movement) is allowed in LFG, unlike in other syntactic theories. Syntax cannot see into the internal composition of words. An example of a c-structure is:

(6)



Notice that LFG uses lexical category information in an innovative manner in order to deal with similarities in distribution that are captured by movement in transformational models. For example, in English auxiliary elements occur in the I c-structure position, however if there is no auxiliary then the verb may appear there, as in:

(7) This child *didn't* believe that the prince has died.

(8) This child *can* believe that the prince has died.

(9) This child *believes* that the prince has died.

In the Principles and Parameters approach the inflected verb is moved (by Head-to-Head movement) from its V position at underlying structure to the I position in a later structure. In LFG the distribution is captured by classifying an inflected verb like *believes* as an I category element marked for TENSE but an untensed form like *believe* as V. A consequence of this is that in sentences like (9) the c-structure VP contains no head V (a principle of *economy* on c-structures says that nodes should only be expanded when necessary).

In a configurational language, c-structure nodes are annotated with functional equations which pass information up the tree from daughter to mother nodes. Functional equations use $\uparrow$ and $\downarrow$ symbolizations: the first can be informally restated as 'information about my mother' and the second as 'information about me'. Thus, in English the DP which is the immediate daughter of IP ('this child' in example (6)) is the subject of the sentence and hence it will be annotated as $\uparrow$ SUBJ = $\downarrow$ ('information about my mother's subject is information about me'). A DP which is the immediate daughter of VP will be annotated as $\uparrow$ OBJ = $\downarrow$. Lexical heads such as V and N simply have $\uparrow = \downarrow$ annotations (i.e. they pass their lexical information up the tree to their mother nodes). Such an annotated tree would be the following:

Figure 1 goes here

LFG assumes that c-structures in configurational languages are generated by context free phrase structure rules where information passing annotations are included, for example:

$$\begin{array}{lll} \mathrm{IP} \rightarrow & \mathrm{DP} & \mathrm{I'} \\ & \uparrow \mathrm{SUBJ} = \downarrow & \uparrow = \downarrow \end{array}$$

In non-configurational languages morphology plays a major role in function assignment.

3.2 *A-structure*

LFG incorporates a richly articulated lexicon where all morpho-syntactically relevant information for individual lexical items is expressed. This includes the a-structure of predicational elements, namely the number and types of argument and complement slots that a predicate subcategorizes for. The following is an example of a lexical entry:

(11) *gives* I (↑ PRED) = ‘give <Agent, Theme, Recipient>’

(↑ TENSE) = present

(↑ SUBJ NUMBER) = sing

(↑ SUBJPERSON) = 3

Here the representation tells us that ‘gives’ is the inflected form of the verb that occurs with a third person singular subject and means an agent gives a theme to a recipient (note that the semantic value of the PRED attribute is only superficially articulated in LFG descriptions). When lexical items are placed in c-structure trees as terminal nodes they will pass their information up to the mother nodes by virtue of the ↑ = ↓ annotation mentioned above. The relation between a-structure and functional information (such as the Agent being the SUBJECT in simple transitive clauses) is captured by mapping correspondence principles (see below).

3.3 *F-structure*

F-structures capture functional information and are sets of paired attributes and values in an *attribute-value matrix*. Attributes are morpho-syntactic features (derived from lexical entries) such as TENSE or NUMBER, or grammatical functions such as SUBJECT and OBJECT. LFG has a richly articulated typology of grammatical functions, including both argument and complement functions subcategorized for by predicates, and adjunct functions that are not. It also has discourse functions such as TOPIC and FOCUS. In configurational languages correspondence principles map from c-structure positions to f-structure functions (see below).

An f-structure for the sentence in (7) above would be:

(12)

$$\left[\begin{array}{ll} \text{SUBJ} & \left[\begin{array}{ll}\text{DEF} & + \\ \text{DIST} & \text{proximate} \\ \text{PRED} & \text{‘child’}\end{array}\right] \\ \text{TENSE} & \text{past} \\ \text{NEG} & + \\ \text{PRED} & \text{‘believe $<$ SUBJ, COMP$>$’} \\ \text{COMP} & \left[\begin{array}{ll} \text{SUBJ} & \left[\begin{array}{ll}\text{DEF} & + \\ \text{PRED} & \text{‘prince’}\end{array}\right] \\ \text{MOOD} & \text{perfect} \\ \text{PRED} & \text{‘die $<$SUBJ$>$’}\end{array}\right]\end{array}\right]$$

Note that we have three layers of f-structure here: the outer f-structure (corresponding to the whole sentence) has five attribute names: SUBJ, TENSE, NEG, PRED and COMP. Three of these are features that have simple values (PRES, +, ‘believe <SUBJ, COMP>’), while SUBJ and COMP are functions that contain subordinate f-structures as their values. These inner f-structures also have attributes and values – the SUBJ function of the embedded COMPlement takes a lower f-structure with grammatical features. In this way, LFG represents all the functional information of the whole sentence.

Note that f-structure has an entirely different organization and vocabulary from c-structure (which are trees made up of constituents like N, DP and VP). F-structure attribute-value matrices are subject to well-formedness conditions of unification, i.e. the information within an f-structure must be:

- *unique* – attributes must have only one value
- *complete* – an f-structure must contain all the grammatical functions that a given predicate requires
- *coherent* – all the grammatical functions must be required by some predicate within the local f-structure

LFG assumes that f-structures have a degree of universality in that sentences in different languages which are translational equivalents will have identical f-structure representations, even if their c-structures are quite different (as they would be for English and Jiwarli, for example).

3.4 *Correspondence Principles*

Central to LFG is the concept of correspondence or *mapping principles* that relate the several parallel and independent representations of sentences. We can identify two of these as crucially important: argument structure to f-structure mapping and c-structure to f-structure mapping.

3.4.1 *A-structure to f-structure mapping*

As we saw, a-structures are specifications of predicates and their associated argument and complement semantic functions. *Lexical mapping theory* establishes correspondences between these functions and grammatical functions like SUBJect and OBJect. LFG assumes that there are four core grammatical functions: SUBJECT, OBJECT, OBJECT$_{\theta}$ (the thematically restricted 'second object' in sentences like 'I gave John *money'*) and OBLIQUE$_{\theta}$ (oblique elements with a range of thematic roles). These are decomposable into two features:

- [± r] for *semantically restricted* functions; OBJECT$_{\theta}$ and OBLIQUE$_{\theta}$ will be [+ r]
- [± o] for *object-like* functions, OBJECT and OBJECT$_{\theta}$ will be [+ o]

This gives the following classification:

SUBJ	=	[-r, -o]	OBJ	=	[-r, +o]
OBL$_{\theta}$	=	[+r, -o]	OBJ$_{\theta}$	=	[+r, +o]

Lexical mapping assumes the existence of a Universal Thematic Hierarchy (see Bresnan and Kanerva 1989, and references therein) which reflects a scale of thematic prominence. The Universal Hierarchy identifies the most prominent (highest) argument which can be selected as the logical subject, and is as follows:

Agent > Beneficiary > Goal > Instrument > Patient/Theme > Locative

Lexical arguments are underspecified with respect to syntactic functions. Internal arguments (patient, theme, and applied arguments) selected by the verb can be underspecified in one of two ways:

(a) as [-r], which results in mapping to SUBJ or OBJ

(b) as [+o], but only if they are **low** on the Universal Thematic Hierarchy, i.e. below Goal. Such arguments will surface as OBJ_{θ}. This gives:

Goal >	Instrument > Patient/Theme > Locative
[-r]	[-r] or [+o]

The external argument or default subject role is assigned to the highest logical argument (signified θ^) which is not internal, i.e. specified as either [-r] or [+o] as noted above.

Mapping from these underspecified representations to full specification of grammatical functions is achieved by two principles:

(I) the SUBJECT PRINCIPLE: assign the features [-r, -o] to:

(a) the external argument; otherwise, to

(b) an internal argument

This assignment is monotonic, that is, feature values can be added but **not** changed.

(II) the DEFAULT PRINCIPLE: complete partially specified functions by assigning a positive value to the unspecified syntactic feature [r] or [o].

The following is an example of how these principles would apply to a predicate such as ‘give’ in English:

	give	<	Agent		Goal	Theme	>
internal					-r	+o	
subject			-r				
			-o				
default					+o	+r	
functions			SUBJ		OBJ	OBJ_θ	
eg.			John	gave	the girl	a book	

Argument mappings can be affected by the operation of *lexical rules* which apply to change the argument structures of predicates. Thus, the *passive* is a lexical rule which removes the highest argument and makes it unavailable for mapping, i.e. $\hat{\theta} \mapsto Ø$ (the result will be that the Agent will no longer be able to be SUBJECT but Patient or Recipient can be). The anti-passive in ergative languages on the other hand removes the logical object argument and makes it unavailable for mapping, that is, OBJ $\mapsto Ø$.

Using these mechanisms it is possible to map from argument structures to the set of grammatical functions associated with the predicate syntactically.

3.4.2 *C-structure to f-structure mapping*

In configurational languages mapping from c-structure to f-structure is mediated by phrase structure annotations (see 10 above) which pass information up the tree from lexical items and from structural positions. This information is mapped from the phrase structure nodes to a *variable*, one for each node in the tree (which then corresponds to a pair of brackets in the f-structure matrix). The variables are resolved by an *f-description*, a set of *functional equations* for all the nodes in the tree. So for example, consider just the subject DP in the tree in (7) above with variables indicated in (13).

Figure 2 goes here

This gives us the following functional equations:

(14) $(f_1\ \text{SUBJ}) = f_2$

$(f_2\ \text{DEF}) = f_3$

$(f_2\ \text{DIST}) = f_3$

$(f_2\ \text{PRED}) = f_4$

These are resolved as:

(15)

$$\left[\begin{array}{ll} \text{SUBJ}\ f_2\ f_3\ f_4 & \left[\begin{array}{ll} \text{DEF} & + \\ \text{DIST} & \text{proximate} \\ \text{PRED} & \text{`child'} \end{array}\right] \end{array}\right]$$

and so on for the rest of the nodes on the tree. That is, we map from c-structure configurations plus lexical information to f-structure matrices.

In non-configurational languages structural position does not define grammatical functions so this information must be derived from other sources. One suggestion (see Nordlinger 1998) is that morphologically inflected forms are annotated to construct their local f-structure environment. So, for example, in a language like Jiwarli, the ergative case form would construct a partial f-structure indicating that the nominal bearing this case is a SUBJECT (regardless of its c-structure position), similarly the accusative constructs an OBJECT function and so on. In head-marking languages the agreement morphology would be involved in resolution of functional equations to construct f-structures directly from morphological (lexical) information, without mapping from c-structure (annotated) positions.

4. Functions in complex sentence constructions – raising and control

Linguists have long recognized that in non-finite complement clause constructions such as the following, the subordinate verb is missing a subject:

(15) Harry doesn’t like to speak Sumbawan

(16) Harry doesn’t seem to speak Sumbawan

Here we understand the main clause subject ‘Harry’ to be also the subject of the subordinate predicate ‘speak Sumbawan’ – the difference between these sentence is:

- in (15) (often called a *control construction*) two thematic roles are involved: Harry bears a thematic relation to ‘like’ and to ‘speak’;
- in (16) (often called a *raising construction*) only one thematic role is involved: Harry bears a thematic relation to ‘speak’, but not to ‘seem’.

In transformational grammars these two sentences are analyzed as having an *empty category* (or *gap*) as the subordinate subject. In the case of control this is PRO, which obtains its reference from the main clause subject, while in the case of raising this is a DP trace that remains after the subordinate subject has been moved up to the main clause:

Referential control

(17) Harry doesn’t like [PRO to speak Sumbawan]

(18) Harry doesn’t seem [t to speak Sumbawan]

Movement

Since LFG does not permit empty categories in c-structure, the constituency of these two sentences types is the same; both involve a main clause verb taking a VP’ (to VP) as its complement without any subject DP in the c-structure:

Figure 3 goes here

The grammatical function of the VP’ element is given as XCOMP, an *open function* complement, i.e. one missing a subject function in c-structure. The identity of this subject function in f-structure is resolved by *functional control.* The lexical entries for *like* and *seem* are annotated to indicate that their XCOMP’s SUBJect is informationally equivalent to their own SUBJect. The difference between the constructions is that for *like* this main clause subject is semantically selected by the verb (inside the < > brackets in the verbs a-structure) while for *seem* it is not (outside the < > brackets in the lexical entry). This is specified as follows (we are assuming a relational specification here, one derived from a-structures by mapping):

(20) *like* V (↑ PRED) = ‘like < (↑ SUBJ) (↑ XCOMP)>’

(↑ SUBJ) = (↑ XCOMP SUBJ)

(21) *seem* V (↑ PRED) = ‘seem < (↑ XCOMP)> (↑ SUBJ)’

(↑ SUBJ) = (↑ XCOMP SUBJ)

The functional control equation is mapped to f-structure as a link between the main subject’s f-structure and the subordinate subject’s f-structure (the controller *f-commands* the controllee, i.e. is part of a less embedded f-structure). This gives us an f-structure like the following (the structure for *seem* will be identical):

(22)

$$\left[\begin{array}{ll} \text{SUBJ} & \left[\text{PRED 'Harry'}\right] \\ \text{PRED} & \text{'like < (}\uparrow\text{ SUBJ) (}\uparrow\text{ XCOMP)>'} \\ \text{TENSE} & \text{present} \\ \text{NEG} & + \\ \text{XCOMP} & \left[\begin{array}{ll} \text{SUBJ} & \left[\quad\right] \\ \text{OBJ} & \left[\text{PRED 'Sumbawan'}\right] \\ \text{PRED} & \text{< (}\uparrow\text{ SUBJ) (}\uparrow\text{ OBJ)>'} \end{array}\right] \end{array}\right]$$

Thus the LFG analysis does not have any empty categories in c-structure and the difference between the raising and control constructions reduces to a difference in lexical entries and a-structure.

5. Long distance dependencies

In sentences such as the following there is a dependency relationship between an element in sentence-initial position and a grammatical function later in the clause:

(23) What doesn't Harry like to speak?

(24) Which analysis does Joan seem to prefer to write about?

(25) That person, I've never seen you talk to.

Since the distance between initial position and the site of the grammatical function assignment can be potentially unlimited, these are referred to as *long distance dependencies* or *unbounded dependencies*. In transformational accounts such sentences are analyzed as involving movement (called wh-movement or A-bar movement) with the sentence-initial *filler* being *extracted* and leaving a *gap* (or wh-trace empty category). Since LFG has no movement operations, long distance dependencies must be captured in another way.

There are several accounts of long distance dependencies in the LFG literature, the most recent of which use the concept of *functional uncertainty* (see Kaplan and Zaenan 1989, Bresnan 2000). Essentially we set up a functional equation which identifies the initial element bearing a *discourse function* (DF) such as TOPIC or FOCUS with a *grammatical function* (GF) such as SUBJECT or OBJECT later in the sentence. The path of this identification can be indefinitely long and passes through (in English) any number of COMPlement clauses – function assignment is thus *uncertain*. Two approaches to the functional equation have been taken: *outside-in functional uncertainty* (Kaplan and Zaenan 1989) or *inside-out functional uncertainty* (Bresnan 2000). We illustrate only the former here. Basically, the initial topic or focus position in a language such as English has associated with it a functional equation like:

(26) $(\uparrow \text{DF}) = (\uparrow \text{COMP* GF})$

This says: in f-structure identify the f-structure of an element bearing a discourse function with the f-structure of a grammatical function, passing through indefinitely many COMPlements (as indicated by the Kleene star operator *). This identification will be signaled by a link in the f-structure, and any restrictions on the long distance dependency (so-called *island constraints*) can be stated in terms of the path from filler to gap through the f-structure. (Note that we used a link in dealing with raising and control structures above, however this is strictly local and is lexically determined.) LFG thus claims that c-structure information has nothing to do with the analysis of long distance dependencies, contrary to transformational accounts. Some evidence that this is correct comes from our observation above (see example 3) that there can be *category mismatches* between the filler and gap, and it is *functional* identification which is relevant.

6. Anaphora

A major concern of modern syntactic theories is the analysis of sentences such as the following which involve a *binding* relationship between an *antecedent* element and an *anaphor* which refers back to it:

(27) Harry talked about himself.

(28) Harry talked to Joan about herself.

(29) The children fought with each other all day.

Crucially, *pronouns* in such sentences cannot refer back to the antecedent, but must be *disjoint* in reference, as in:

(30) Harry talked about him. (*not* Harry)

(31) Harry talked to Joan about her. (*not* Joan)

(32) The children fought with them all day. (*not* each other)

Also, the anaphor cannot generally occur before the antecedent:

(33) *Himself talked about Harry.

(34) *Each other fought with the children all day.

The correct analysis of sentences such as these has been a major challenge for syntactic theory, and is usually stated in terms of constituency (c-structure) relationships such as precedence (linear order) and c-command (informally a constituent c-commands another if it is higher in the tree structure).

Work within LFG (Dalrymple 1993, Bresnan 2000, Falk 2001) has shown that cross-linguistically anaphors are of several types and the factors which determine their distribution can be c-structure relationships, as well as a-structure ones (the antecedent must be *thematically higher* than the anaphor, according to the hierarchy discussed in 3.4.1) and f-structure ones (the antecedent must be *relationally higher* than the anaphor on a hierarchy where SUBJECT precedes OBJECT which precedes OBLIQUE$_{\theta}$, for example), with f-structure prominence or rank being crucially important (see Bresnan 2000 for discussion and arguments). This is entirely to be expected in a linguistic theory such as LFG which has parallel representations, each of which is independent and each of which captures different aspects of linguistic structure. The power of a model of anaphor that incorporates functional information is demonstrated by Dalrymple's (1993) close attention to the distribution of different types of anaphors and restrictions on their use in a wide range of languages.

7. Current issues

Like most linguistic theories, Lexical Functional Grammar is constantly being tested and revised by its practitioners to deal with empirical facts about individual language structures and to capture cross-linguistic generalizations in the most economical and insightful manner. Among the areas currently of concern in LFG are:

- the role of morphology in language structure, including the contribution of morphology to functional structure independently of c-structure, the relationship between inflectional morphology and syntax (and the notions of functional category, competition and blocking), and the interface between morphology and syntax (including concepts of what it is to be a word, and how to analyze predicates that are morphologically complex, such as serial verb or main verb-classifying verb constructions);

- the nature of semantic representations and σ-structure, especially the development of a ‘glue language’ to relate semantics and f-structures;
- computational implementation of LFG grammar models on large data sets. Active research is underway in Europe and the United States on computer architectures for LFG (including grammar-translation) using a data-oriented probabilistic approach to LFG-parsed corpora (called DOP-LFG)
- a number of scholars are working on the relationship between Optimality Theory (OT) and LFG models, exploring how violable constraints can be added to LFG analyses to give a preference ranking between them, and how OT approaches to syntax can be informed by research on parallel constraint-based models such as LFG.

8. Conclusion

Lexical Functional Grammar is a powerful approach to linguistic analysis and theory building that has been practiced for over twenty-five years. It is characterized by the idea of parallel independent but related linguistic structures that each contribute information in their own constrained fashion. It has been applied to a wide range of languages and is being actively developed by an international community of scholars for linguistic analysis, theory construction, and computational modeling with the goal of analyzing and understanding the nature of human language.

9. Further reading

Several excellent introductions to LFG have appeared recently: Falk 2001 is a very readable introduction to the main concepts, as is Chapter 13 of Carnie 2001. Bresnan 2000 is a more advanced state-of-the-art account that introduces a number of new ideas, as well as presenting the theory. Earlier accessible articles are Neidle 1994, Sadler 1996, and Dalrymple 1999; Bresnan 1982 and Dalrymple et al 1995 contain classic articles that remain relevant, though now somewhat dated.

Much material is available on the internet through the Stanford and University of Essex sites:

`http://www-lfg.stanford.edu/lfg/`

`http://clwww.essex.ac.uk/LFG/`

and through the International Lexical Functional Grammar Association at:

`http://www-lfg.stanford.edu/lfg/ilfga/`

There is also an LFG e-mail and discussion list open to all interested persons: access to it is described on the web sites listed above.

10. Bibliography

Austin, Peter 2001 Word order in a free word order language: the case of Jiwarli. In J. Simpson, D. Nash, M. Laughren, P. Austin and B. Alpher, eds, *Forty years on*. Canberra: Pacific Linguistics.

Austin, Peter and Joan Bresnan 1995 Non-configurationality in Australian Aboriginal languages, *Natural Language and Linguistic Theory* 14: 215-268

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

Bresnan, Joan 2000 *Lexical Functional Syntax*. London: Blackwell.

Bresnan, Joan and Jonni Kanerva 1989 Locative inversion in Chichewa: a case study of factorization in grammar. *Linguistic Inquiry*, 20(1):1-50.

Carnie, Andrew 2001 *Syntax*. London: Blackwell.

Dalrymple, Mary 1993 *The Syntax of Anaphoric Binding*. CSLI Publications, Stanford, CA. CSLI Lecture Notes, number 36.

Dalrymple, Mary. 1999. Lexical-Functional Grammar. *MIT Encyclopedia of the Cognitive Sciences*, ed. Rob Wilson and Frank Keil. The MIT Press.

Dalrymple, Mary, Ronald M. Kaplan, John T. Maxwell, III, and Annie Zaenen, eds. 1995 *Formal Issues in Lexical-Functional Grammar*. Stanford University: Center for the Study of Language and Information.

Falk, Yehuda 2001 *Lexical-Functional Grammar: An introduction to parallel constraint-based syntax*. Stanford: Center for the Study of Language and Information.

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*. Chicago University Press, Chicago, pp. 17-42. Reprinted in Mary Dalrymple, Ronald M. Kaplan, John Maxwell, and Annie Zaenen, eds. 1995, 137-165.

Neidle, Carol 1994 Lexical-Functional Grammar. *Encyclopedia of Language and Linguistics*. New York: Pergamon Press.

Nordlinger, Rachel 1998 *Constructive Case: Dependent Marking Nonconfigurationality in Australia*. Stanford: CSLI Publications.

Sadler, Louisa 1996 New developments in LFG. In Keith Brown and Jim Miller, editors, *Concise Encyclopedia of Syntactic Theories.* Oxford: Elsevier Science

Figure 1

(10)

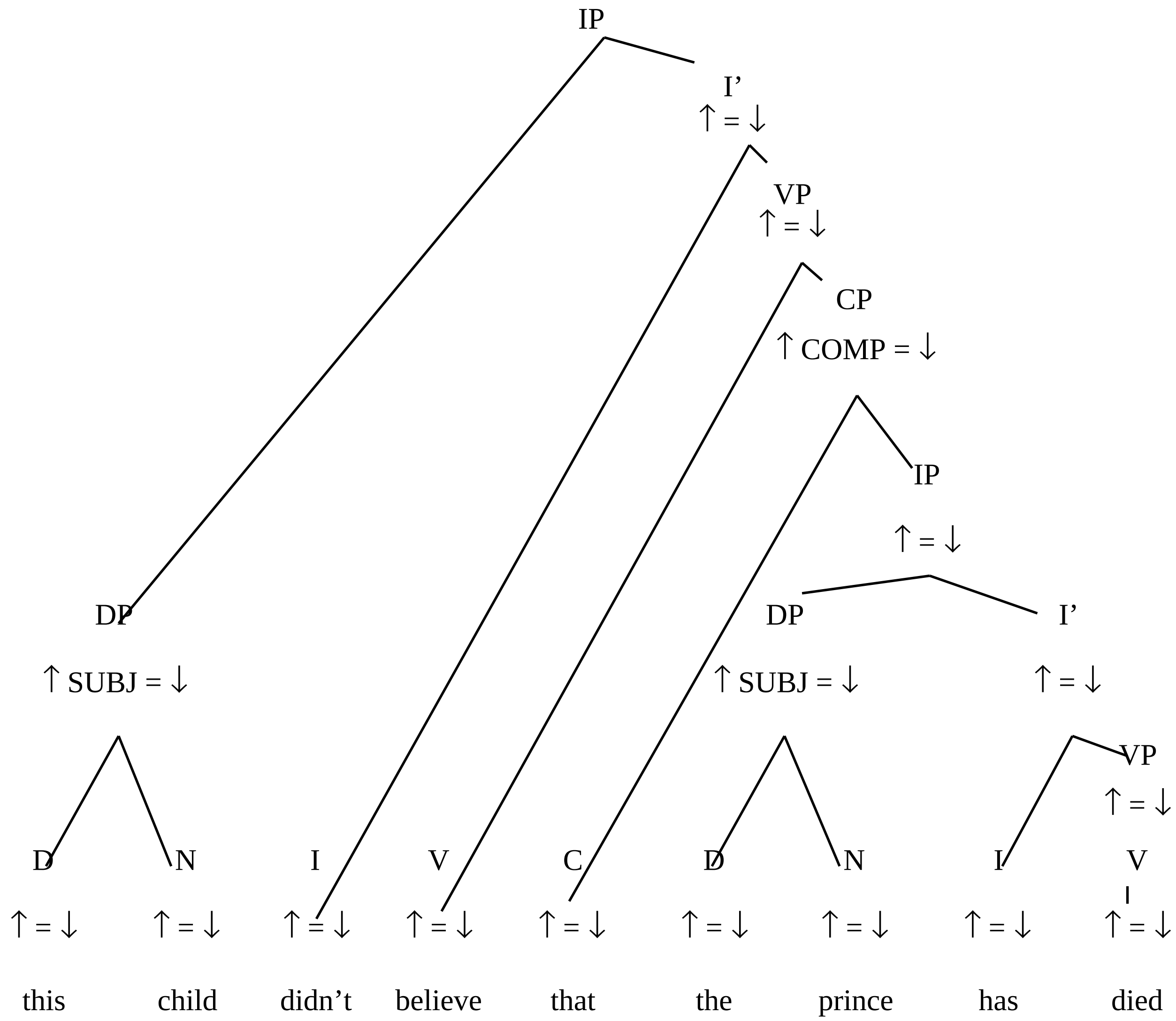

IP
I’
↑ = ↓
VP
↑ = ↓
CP
↑ COMP = ↓
IP
↑ = ↓
DP
↑ SUBJ = ↓
DP
↑ SUBJ = ↓
I’
↑ = ↓
VP
↑ = ↓
D N I V C D N I V
↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓ ↑ = ↓
this child didn’t believe that the prince has died

Figure 2

(13)

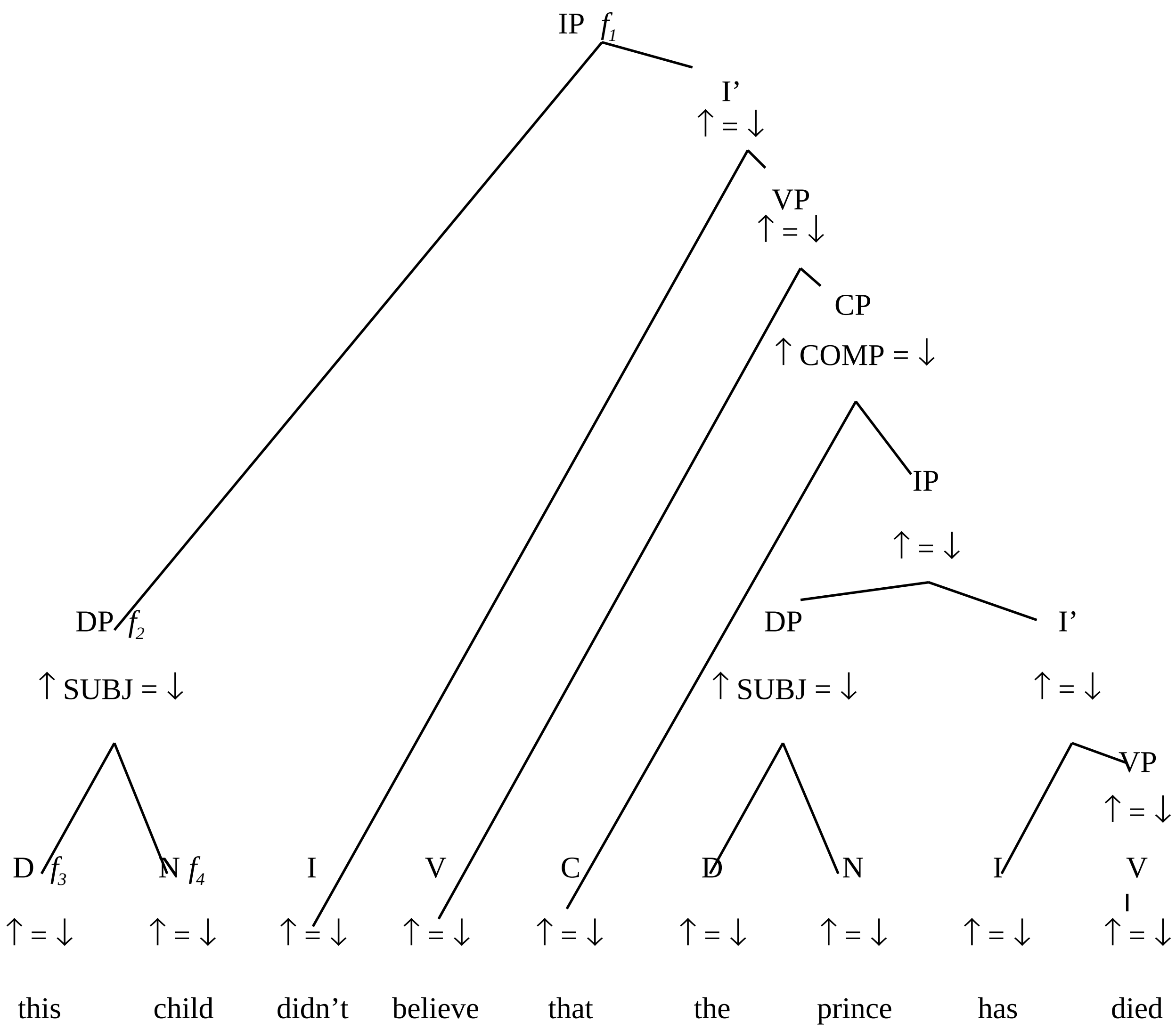
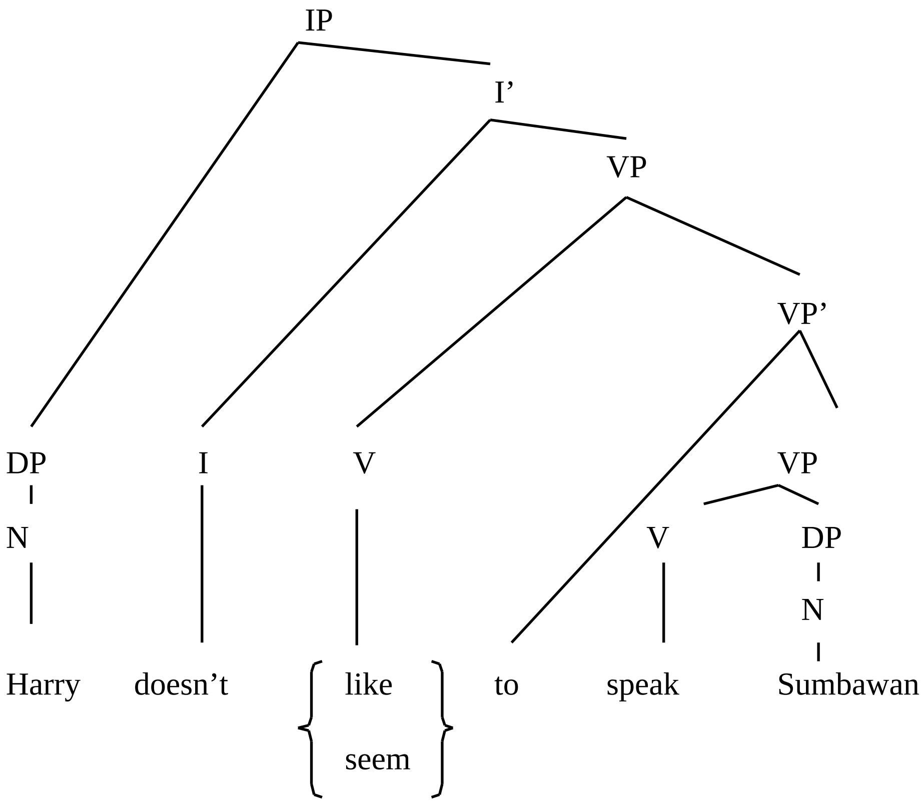

IP f_1
I’
↑ = ↓
VP
↑ = ↓
CP
↑ COMP = ↓
IP
↑ = ↓
DP f_2
↑ SUBJ = ↓
DP
↑ SUBJ = ↓
I’
↑ = ↓
VP
↑ = ↓
D f_3
N f_4
I
V
C
D
N
I
V
↑ = ↓
this
child
didn’t
believe
that
the
prince
has
died

Figure 3

(19)


IP
I’
VP
VP’
DP
I
V
VP
N
V
DP
N
Harry
doesn’t
like
seem
to
speak
Sumbawan