

**CAUSATIVES AS COMPLEX PREDICATES WITHOUT
THE RESTRICTION OPERATOR**

Petr Homola and Matt Coler
Codesign, s.r.o. INCAS3

Proceedings of the LFG13 Conference

Miriam Butt and Tracy Holloway King (Editors)

2013

CSLI Publications

<http://csli-publications.stanford.edu/>

Abstract

The paper explores an extension of the LFG framework at the level of semantics. The problem of so-called syntactically formed complex predicates is discussed and a solution is proposed that combines the elegance of the treatment of morphology and syntax within LFG with modern approaches to semantics.

1 Introduction

While LFG provides a well-studied framework for morphosyntactic analysis, semantics is less elaborated and several mutually incompatible approaches have been proposed within this formalism (we survey some of them briefly in Subsection 5.1). The aim of this paper is to sketch a formal approach to semantics and its integration with the apparatus of LFG.

The impulse for the reported research was a close examination of so-called syntactically formed complex predicates, especially causatives in Romance languages. Mohanan (1997) defines complex predicates as a “[...] construction [...] in which two semantically predicative elements jointly determine the structure of a single syntactic clause.” While complex predicates appear already in the work of Aissen and Perlmutter (1983), in the context of LFG they were investigated, for example, by Alsina (1996, 1997) and Butt (1997).¹ The latter approach was the point of departure for our investigation. Aside from Catalan causatives, we will also illustrate the interplay between syntax and (lexical) semantics with examples from Aymara, a polysynthetic language. We restrict ourselves to verb-verb compounds.

We use Kaplan’s (1995) mathematical notation² throughout the paper.

2 Complementation vs. Composition

Within LFG, Alsina’s (1996; 1997) concept of complex predicates has been applied to a number of phenomena and languages including, for example, Turkish causatives (Çetinoğlu et al., 2008). These approaches are comparable insofar as they use the so-called restriction operator (RO) introduced by Kaplan and Wedekind (1993). The reason for the use of the RO is the fact that standard LFG does not allow for modification of the PRED attribute in f-structures. The RO solves this problem by allowing the exclusion of attributes from an f-structure. We propose two solutions to this problem that circumvent the RO. One is the so-called equational unification, a generalization of the unificational mechanism used

¹See also (Alsina et al., 1997) for a general overview.

²More specifically, A denotes the set of atomic symbols, N denotes the set of nodes, F denotes the set of feature structures, S denotes the set of semantic forms. An f-structure is a function that takes atomic symbols to $A \cup F \cup S$. Functional correspondence is expressed by the function $\phi : N \rightarrow F$ that takes nodes in a c-structure to f-structures.

in “standard” LFG, and the other is the so-called event semantics. The latter approach represents a rather radical extension of LFG that goes far beyond the treatment of complex predicates.

Consider the following Catalan example. The PRED value of the main f-structure we use in the examples to explicate how E -unification works is given in (2).

- (1) *El vaig fer riure.*
 him go.1SG do.INF laugh.INF
 “I made him laugh.”
- (2) $\text{cause}\langle(\uparrow \text{SUBJ}), \text{riure}\langle(\uparrow \text{OBJ})\rangle\rangle$

Unlike in languages with morphologically formed causatives, such as Turkish or Aymara, Catalan causatives are formed syntactically, i.e. the complex PRED value is not created in the lexicon. We will show in the next section that the creation of complex predicates can be formalized as unification. First, however, let us illustrate Alsina’s formal treatment of complex predicates.

The operation that “merges” two semantic forms is called “composition” by Alsina as opposed to “complementation” (which is the term he uses for structural unification). His “composition” operation, which uses the RO, is defined thus (Alsina, 1997, p. 236):

Definition 1 *The operator $=_H$ is defined as follows:*

$$\begin{aligned} \uparrow =_H \downarrow &\stackrel{df}{=} (\uparrow \setminus \text{PRED}) = (\downarrow \setminus \text{PRED}) \\ &(\uparrow \text{PRED}) = \\ &= F((\downarrow \text{PRED}), (\rightarrow_H \text{PRED})) \end{aligned}$$

where the symbol \rightarrow_H refers to a sister node with the head equation and

1. $F(x, \emptyset) = x$,
2. $F(P^1\langle a \rangle, \dots P^*\langle b \rangle \dots) = \dots P^1\langle c \rangle \dots$ where P^* is an unspecified predictor and c is the unification of a and b ,
3. elsewhere, the result is vacuous.

We illustrate in the next section how two semantic forms, like (3), are composed to form a complex form in a single f-structure.

3 E -Unification in Term Algebras

The process of the composition of f-structures is based on unification. In standard LFG, PRED values are treated as atoms in the syntactic component, making it impossible to alter them. However, it is possible to use equational unification, a

concept used in logical programming for deduction and reasoning, to circumvent the RO and use only standard functional descriptions.

For Catalan causatives, we assume the following PRED values of the verbs in (1):

$$(3) \quad \begin{array}{ll} \textit{fer} & \text{cause}\langle(\uparrow \text{SUBJ}), f\langle(\uparrow \text{OBJ})\rangle\rangle \\ \textit{riure} & \text{laugh}\langle(\uparrow \text{SUBJ})\rangle \end{array}$$

The symbol f represents a higher-order variable that can be instantiated with a function symbol. To form a complex causative predicate in the syntax, we define an equational theory ($\dot{=}_E$) induced by the following term identity (equational theories E_i constitute a separate part of the grammar, aside from the lexicon and context-free rules with functional annotations):

$$(4) \quad E = \{\text{cause}\langle(\uparrow \text{SUBJ}), f\langle(\uparrow \text{OBJ})\rangle\rangle \approx f\langle(\uparrow \text{SUBJ})\rangle\}$$

If we unify the f-structures of *fer* and *riure* modulo $\dot{=}_E$, we get the complex PRED value given in (2).³

3.1 Formal Definition of E -Unification

Let $\mathcal{T}(\mathcal{F}, \mathcal{V})$ be a term algebra with a signature (set of function symbols) \mathcal{F} and a set of variables \mathcal{V} . Let E be a set of equations over $\mathcal{T}(\mathcal{F}, \mathcal{V})$ (called identities or axioms). We define equational theory $\dot{=}_E$ as the least congruence relation on $\mathcal{T}(\mathcal{F}, \mathcal{V})$ closed under substitution and containing E . More formally, $\dot{=}_E$ is the least binary relation on $\mathcal{T}(\mathcal{F}, \mathcal{V})$ with the following properties:

1. $E \subseteq \dot{=}_E$
2. $s \dot{=}_E s$ for all s
3. if $s \dot{=}_E t$ then $t \dot{=}_E s$ for all s, t
4. if $s \dot{=}_E t$ and $t \dot{=}_E r$ then $s \dot{=}_E r$ for all s, t, r
5. if $s_1 \dot{=}_E t_1, \dots, s_n \dot{=}_E t_n$ then $f(s_1, \dots, s_n) \dot{=}_E f(t_1, \dots, t_n)$ for all s, t, n, f
6. if $s \dot{=}_E t$ then $s\sigma \dot{=}_E t\sigma$ for all s, t, σ

An E -unification problem over \mathcal{F} is a finite set of equations $\Gamma = \{s_1 \dot{=}_E t_1, \dots, s_n \dot{=}_E t_n\}$ where $s_i, t_i \in \mathcal{T}(\mathcal{F}, \mathcal{V})$. An E -unifier of Γ is a substitution σ such that $s_1\sigma \dot{=}_E t_1\sigma, \dots, s_n\sigma \dot{=}_E t_n\sigma$. $\mathcal{U}_E(\Gamma)$ is the set of E -unifiers of Γ and Γ is E -unifiable iff $\mathcal{U}_E(\Gamma) \neq \emptyset$. Unlike syntactic unification, a most general unifier may not exist. Typically, one can compute a minimal complete set of E -unifiers (i.e., a set of unifiers that are not comparable to each other with respect to the relation of being more general modulo E).

³Since E -unification as defined in the next subsection operates on first-order terms, we reify all terms in order for the function variable to be an argument. For example, $f(x)$ becomes $\textit{pred}(f, x)$.

3.2 Discussion

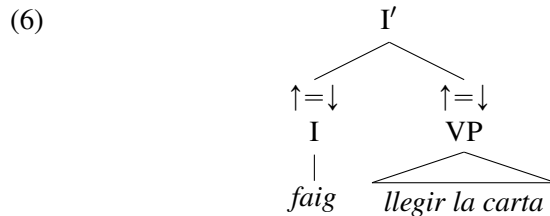
It follows from the definition given above that equational unification is a simple generalization of syntactic unification. In (1), for example, we unify the PRED values of *fer* and *riure* modulo E as defined in (4) using the substitution

$$\sigma = \{f \mapsto \text{riure}\}$$

We have seen how the f-structures of *fer* and *riure* can be combined to an f-structure with a complex PRED value using only unification, i.e. using $\uparrow=\downarrow$ for both nodes. By defining an equational theory $\dot{=}_E$ over a set of identities E ((4) is a simple example for causatives of intransitive verbs) E can be understood as a linguistically motivated transparent description of syntactically formed PRED altering constructions. E is, of course, language-specific while the unification mechanism is universal.

We conclude this section with an example for a transitive verb (in “standard” LFG, the unification of the PRED values in (6) would lead to a clash, thus ‘=’ is to be understood as E -unification):⁴

- (5) *Li faig llegir la carta.*
 him do.1SG read.INF the letter
 “I make him read the letter.”



The PRED value of the f-structure of the sentence and the corresponding E -theory are as follows:

$$(7) \text{ cause}\langle(\uparrow \text{ SUBJ}), \text{llegir}\langle(\uparrow \text{ OBJ}_\theta)(\uparrow \text{ OBJ})\rangle\rangle$$

$$(8) E = \{\text{cause}\langle(\uparrow \text{ SUBJ}), f\langle(\uparrow \text{ OBJ}_\theta)(\uparrow \text{ OBJ})\rangle\rangle \approx f\langle(\uparrow \text{ SUBJ})(\uparrow \text{ OBJ})\rangle\}$$

4 E -Unification of Feature Structures

The solution proposed in the previous section, while formally sound and correct, entails a serious problem: it uses higher-order logic since the variable f in the

⁴It is clear that verbs with different valency frames need to be treated by different equations. To make the declarations more transparent to traditional linguists, one could use equation templates to express more equations with merely one expression (much like $S \rightarrow C^+$ is a rule template which represents an infinite set of context-free rules).

equations in (4) and (8) stands for a function symbol. Thus, we adapt the method to a “reified” semantic representation and investigate E -unification of feature structures. We assume an intuitive interpretation of feature structures as directed acyclic graphs with edges labelled by atomic symbols (elements from A) and nodes labelled by elements from $F \cup A \cup S$. A formal treatment based on attribute-value logic will be described in the next subsection.

For clarity, we repeat here (1) as (9).

- (9) *El vaig fer riure*
 him go.1SG do.INF laugh.INF
 “I made him laugh.”

The corresponding lexico-semantic value of the complex predicate, expressed as a feature structure, is given in (10).⁵ The feature structure is a straightforward (recursive) conversion of a term where FN is its functor and ARG n its arguments.

$$(10) \left[\begin{array}{cc} \text{FN} & \text{'cause'} \\ \text{ARG1} & \text{SUBJ} \\ \text{ARG2} & \left[\begin{array}{cc} \text{FN} & \text{'riure'} \\ \text{ARG1} & \text{OBJ} \end{array} \right] \end{array} \right]$$

Note that there is no way to obtain (10) from the feature structures of the two verbs in (9) as formed in the lexicon (given in (11)⁶) by means of structural unification.

$$(11) \left[\begin{array}{cc} \text{FN} & \text{'cause'} \\ \text{ARG1} & \text{SUBJ} \\ \text{ARG2} & \left[\begin{array}{cc} \text{FN} & f \\ \text{ARG1} & \text{OBJ} \end{array} \right] \end{array} \right] \left[\begin{array}{cc} \text{FN} & \text{'riure'} \\ \text{ARG1} & \text{SUBJ} \end{array} \right]$$

The operation that merges the two values (“composition”) can be implemented by means of E -unification of feature structures based on the following E -theory⁷ (the curve denotes, according to a common LFG notation, value sharing).

$$(12) E = \left\{ \left[\begin{array}{cc} \text{FN} & \text{'cause'} \\ \text{ARG1} & \text{SUBJ} \\ \text{ARG2} & \left[\begin{array}{cc} \text{FN} & \text{---} \\ \text{ARG1} & \text{OBJ} \end{array} \right] \end{array} \right] \approx \left[\begin{array}{cc} \text{FN} & \\ \text{ARG1} & \text{SUBJ} \end{array} \right] \right\}$$

As with term algebras, E states that, for example, *He laughs* and *I made him laugh* are equal modulo the feature of causation.

⁵We use attribute symbols from (Kaplan and Maxwell, 1996).

⁶These PRED values are encoded as terms in the corresponding morpholexical entries of the verbs in the lexicon.

⁷Here again, ‘=’ in the corresponding rule annotation denotes E -unification.

4.1 Formal Representation of Feature Structures

This subsection explicates how the extended unification method, described in the previous subsection, is implemented in our system. Feature structures can be represented as formulae of a fragment of first order logic with equality. We use the attribute-value logic (henceforth AVL) proposed by Wedekind (1991) which is briefly sketched here in the context of E -unification.

In AVL, we have a set of constants \mathcal{C} and a set of unary function symbols \mathcal{F}_1 ($\mathcal{C} \cap \mathcal{F}_1 = \emptyset$). The class of terms \mathcal{T} contains all constants and if τ is a term and f a function symbol, $f\tau$ is also a term. The atomic formulae of AVL have the form $\tau_1 = \tau_2$ where $\tau_1, \tau_2 \in \mathcal{T}$ or \perp .⁸ The formulae are formed recursively by means of the logical connectives \neg and \vee .

Equivalence classes of certain AVL formulae can be seen as a commutative idempotent monoid. The atoms of the monoid are atomic formulae of the form $\sigma a = b$ where $\sigma \in \mathcal{F}_1^+$ and a and b are constants. The unit element is \top and the complementation operation is \wedge . Subsumption is defined as $\varphi \sqsubseteq \psi \equiv \psi \rightarrow \varphi$. It is easy to see that a thus defined monoid is commutative and idempotent (further it is atomistic and distributive). The operator I that states whether a feature structure is well-formed can be defined on the set of formulae as follows:

$$I(\varphi) = \begin{cases} \mathbf{1} & \text{if } \varphi \text{ is satisfiable} \\ \mathbf{0} & \text{otherwise} \end{cases}$$

Thus $I(\varphi \wedge \psi) = \mathbf{1}$ if the feature structures expressed by φ and ψ are unifiable. Since AVL is decidable, I is computable for all formulae.

Now we turn to E -theories. The equational axioms (the elements of E) have the form

$$\sigma_1 x \wedge \cdots \wedge \sigma_n x \approx \tau_1 x \wedge \cdots \wedge \tau_m x$$

where $\sigma_i, \tau_i \in \mathcal{F}_1^+$ and $x \in \mathcal{C}$ (i.e., x is a constant). Sometimes it may be useful to allow x to be a term, i.e., ηy where $\eta \in \mathcal{F}_1^*$ and $y \in \mathcal{C}$ (which gives us the possibility of replacing a substructure in a feature structure).

4.2 Computationally Tractable E -Unification via Rewriting

In this subsection we briefly describe an efficient implementation of E -unification as implemented in our experiments. Readers not interested in implementation details may skip to Section 5. While there is a universal E -unification procedure that is sound and complete (Baader and Snyder, 2001; Gallier and Snyder, 1989), it is in general very inefficient and yields redundant results. In this subsection, we briefly discuss techniques that allow for an efficient E -unification algorithm for a subclass of E -theories. The terminology in what follows as well as some theorems are taken from (Baader and Nipkow, 1998) (for this reason we omit the proofs).

⁸We use the symbol $=$ instead of \approx for equality in AVL as the latter symbol is already in use for equational axioms.

Definition 2 A rewriting system R is terminating if there is no infinite chain

$$a_1 \rightarrow_R a_2 \rightarrow_R \dots$$

Definition 3 b is a normal form of a if $a \xrightarrow{*}_R b$ and there is no c such that $b \rightarrow_R c$. A rewriting system R is called normalizing if every a has a normal form.

Definition 4 A rewriting system R is called locally confluent if for every a, b, c such that $a \rightarrow_R b, a \rightarrow_R c$ there exists a d and $b \xrightarrow{*}_R d, c \xrightarrow{*}_R d$.

Definition 5 A rewriting system R is called confluent if for every a, b, c such that $a \xrightarrow{*}_R b, a \xrightarrow{*}_R c$ there exists a d and $b \xrightarrow{*}_R d, c \xrightarrow{*}_R d$.

Lemma 1 (Newman’s lemma) A terminating rewriting system is confluent iff it is locally confluent.

E -equivalence on F can be interpreted as a confluent rewriting system if an ordering $<$ defined on F can be found such that $a < b \rightarrow a \cdot c < b \cdot c$ for all $a, b, c \in F$ (i.e. $<$ is monotone) and the Knuth-Bendix completion procedure (Knuth and Bendix, 1970) succeeds. In such a case, we obtain a rewriting system R and $\dot{=}_R$ (defined by $a \dot{=}_R b \equiv a \xleftrightarrow{*}_R b$) is equivalent to $\dot{=}_E$.

Note that if there is an oriented rewriting system \rightarrow_R equivalent to $\dot{=}_E$ that is not confluent, we can check E -equivalence of a and b by computing all normal forms $a \xrightarrow{*}_R \hat{a}$ and $b \xrightarrow{*}_R \hat{b}$ and checking whether $\hat{a} = \hat{b}$.

5 Towards More Flexible Semantic Forms

In the previous section, we have seen how values of the PRED attribute (elements of S) can be represented using (a fragment of first-order) logic. Now we put semantic forms in context of the well-studied so-called “Davidsonian” semantics. Davidson (1967) analyzes sentences using events that stand for actions treated as individuals in a first-order language. We use Hobbs’ (1985; 2003) notation⁹ (such as primed predicates that denote “eventualities”) in the examples.

After surveying two most elaborated approaches to semantics within LFG, we show how simple and complex predicates are represented and how semantic representations can be incrementally created in LFG via codescription.¹⁰

⁹A reviewer has raised the question whether the so-called “donkey sentences” can be build up in this formalism. The answer is “yes” and the issue is discussed in detail by Hobbs (1983).

¹⁰While we used LFG in the experiments, the method is flexible enough to be used in any rule-based grammar formalism based on context-free or categorial grammars such as (Uszkoreit, 1986) or (Kay, 1979, 1984).

5.1 Related Work

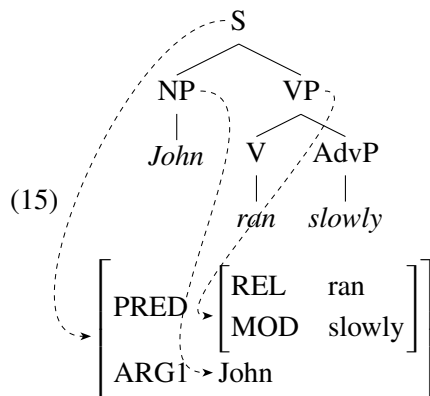
Virtually all approaches to formal semantics assume the Principle of Compositionality, formally formulated by Partee (1995) as follows: “The meaning of a whole is a function of the meanings of the parts and of the way they are syntactically combined.” This means that semantic representation can be incrementally built up when constituents are put together during parsing. Since c(onstituent)-structure expresses sentence topology rather than grammatical relations, the rules that combine the meanings of subphrases frequently refer to the underlying syntactic structure, that is, f(unctional)-structure in LFG. Indeed, Halvorsen and Kaplan (1995) in their account of semantics within LFG define the s(emantic)-structure as a projection of the c-structure (through the correspondence function σ) but they refer to grammatical functions (GFs) by means of the compound function $\sigma\phi^{-1}$ (ϕ is the correspondence function from c-structures to f-structures), as in the following example:

(13) John ran slowly.

The corresponding lexical entry for the verb is

$$(14) \quad \begin{aligned} (\sigma\mathcal{M} * \text{REL}) &= \text{ran} \\ (\sigma\mathcal{M} * \text{ARG1}) &= \sigma\phi^{-1}(\uparrow \text{SUBJ}) \\ (\sigma\mathcal{M} * \text{ARG2}) &= \sigma\phi^{-1}(\uparrow \text{OBJ}) \end{aligned}$$

and the resulting correspondence between the c-structure and the s-structure is



Note that Halvorsen and Kaplan (1995) represent s-structures as feature structures (since they use functional annotations to construct them). Example (15) can be more conventionally expressed as $\text{slowly}(\text{ran})(\text{John})$.

Recent approaches to semantics in LFG are based on the so-called “glue semantics” (Dalrymple et al., 1993, 1995; Dalrymple, 2001). Consider the sentence

(16) Bill obviously kissed Hillary

Its semantic form is, according to Dalrymple et al. (1993),

$$\text{obviously}(\text{kiss}(\text{Bill}, \text{Hillary}))$$

Glue semantics uses linear logic; lexical entries are assigned “meaning constructors” that consist of a logical expression and instructions for how the meaning is put together. For *to kiss*, for example, we have

$$(17) \quad \forall X, Y. (f \text{ SUBJ})_\sigma \rightsquigarrow X \otimes (f \text{ OBJ})_\sigma \rightsquigarrow Y \multimap f_\sigma \rightsquigarrow \textit{kiss}(X, Y)$$

In words, (17) means that if the meaning of $(f \text{ SUBJ})_\sigma$ is X and the meaning of $(f \text{ OBJ})_\sigma$ is Y , then the meaning of f_σ is $\textit{kiss}(X, Y)$. For brevity, meaning constructors are sometimes written as $\llbracket \text{word} \rrbracket$. For (16), then, we get

$$(18) \quad \llbracket \text{Bill} \rrbracket \otimes \llbracket \text{obviously} \rrbracket \otimes \llbracket \text{kissed} \rrbracket \otimes \llbracket \text{Hillary} \rrbracket \\ \vdash f_\sigma \rightsquigarrow \textit{obviously}(\textit{kiss}(\textit{Bill}, \textit{Hillary}))$$

The idea behind glue semantics is that the lexicon and the rules for syntactic assembly provide meaning constructors that are interpreted as soon as all expressions on the left-hand side of the linear implication (\multimap) are available.¹¹

Note that both Halvorsen and Kaplan (1995) and glue semantics use higher-order logic. Furthermore, both approaches are “functionist” In the next section we go on to outline an account of semantics that, while using codescription, relies on pure first-order logic (FOL) for representation and on conjunction of existentially quantified positive literals as the means of meaning assembly, as advocated by Hobbs (1985) and on Minimalist grounds by Pietroski (2005).

5.2 Davidsonian Logical Representation

The sentence *John loves Mary* can be logically expressed (disregarding tense for the sake of simplicity) using a binary predicate for the verb and constants for its arguments:

$$(19) \quad \textit{love}(\textit{John}, \textit{Mary})$$

Davidson (1967) has introduced “events” into the description of logical forms of sentences to be able to refer to “actions” by means of FOL (i.e. events are treated as individuals). We use the notation and terminology of Hobbs (1985; 2003) who introduced the “nominalization operator” and the term “eventuality” to refer to “possible events”. The predicate *love* in (19) can be “nominalized” and defined as follows:

$$(20) \quad \textit{love}(x, y) \equiv \exists e. \textit{love}'(e, x, y) \wedge \textit{Reexists}(e)$$

The newly introduced variable e is the eventuality of John’s loving Mary and Hobbs’ predicate *Reexists* states that the eventuality is realized (this predicate is

¹¹More recent work on glue semantics uses a slightly different notation. (17) would be written as

$$\lambda X. \lambda Y. \textit{kiss}(X, Y) : (\uparrow \text{SUBJ})_\sigma \multimap [(\uparrow \text{OBJ})_\sigma \multimap \uparrow_\sigma]$$

discussed at length in (Hobbs, 1985, 2003), we will not need it in the remainder of the paper).

In the rest of the paper, we refer to “Davidsonian” formulae (with actions denoted by primed predicates) as *conjunctive logical forms* (CLF).¹²

5.3 Conjunctive Logical Forms and Parsing

As Bresnan (2001) puts it, “the formal model of LFG is *not* a syntactic theory [...] Rather, it is an architecture for syntactic theory”. In light of this fact, we show in this section how CLFs can be integrated with the context-free backbone of LFG regardless of the concrete theory used (such as X' theory).

Recall that in the formal architecture of LFG, N is the set of nodes and F is the set of f-structures. By Φ we denote the set of formulae (CLFs) and V denotes the set of variables that may occur in CLFs. In standard LFG, the mapping $M : N \rightarrow N$ takes nodes to their mother node and $\phi : N \rightarrow F$ takes nodes to f-structures. We introduce $\xi : N \rightarrow \Phi$ that takes nodes to formulae and $\tau : N \rightarrow V$ that takes nodes to variables.

For terminal nodes, ξ and τ are defined in the lexicon. The conversion of standard LFG semantic forms (PRED values) to CLFs is almost straightforward. A few examples are given in the following table:

| | $(\phi M * \text{PRED})$ | $\xi M*$ | $\tau M*$ |
|------|---|---|-----------|
| (22) | ‘John’ | $x = \text{John}$ | x |
| | ‘dog’ | $\text{dog}(x)$ | x |
| | ‘see $\langle(\uparrow\text{SUBJ}),(\uparrow\text{OBJ})\rangle$ ’ | $\text{see}'(e, \tau\phi^{-1}(\phi M * \text{SUBJ}), \tau\phi^{-1}(\phi M * \text{OBJ}))$ | e |

The variables in the lexicon (i.e., x and e in (22)) are instantiated independently for every morpholexical entry. The derivation is considered invalid if the corresponding values of ϕ^{-1} are not defined.

Since the variables used in morpholexical entries are distinct, they are instantiated as unique in the same way as PRED values in standard LFG. An illustration (in prenex normal form) is given in (23).

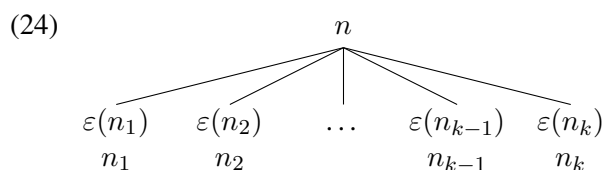
$$(23) \quad \begin{array}{c} \text{John sees a dog.} \\ \text{---} \\ \exists e, x_1, x_2. \text{see}'(e, x_1, x_2) \wedge x_1 = \text{John} \wedge \text{dog}(x_2) \end{array}$$

¹²An alternative representation has been suggested by Parsons (1990). He, too, uses events but proposes unary predicates for actions and special predicates for their arguments. In this spirit, we use the following notation:

$$(21) \quad \text{love}'(e, x, y) \equiv \text{love}''(e) \wedge \text{actor}(e, x) \wedge \text{patient}(e, y)$$

The formula of a nonterminal node is composed from the formulae of its daughter nodes. In LFG, the context-free rules are enriched with functional (morphosyntactic) annotations. Likewise, we enrich them with logical annotations. Since Hobbs' (1985; 2003) "ontologically promiscuous" formulae are conjunctions of positive literals, we combine the formulae of the daughter nodes using the logical connective \wedge .

In (24), n_i are the daughter nodes of n and $\varepsilon(n_i)$ are the corresponding logical annotations in rules (such as $actor(\Delta, \nabla)$ for a NP node). Note that $\{n_1, \dots, n_k\} = M^{-1}(n)$.



Since n is a nonterminal node, $M^{-1}(n) \neq \emptyset$. We define $\xi(n)$ for nonterminal nodes by

$$(25) \quad \xi(n) = \exists \tau(n). \bigwedge_{m \in M^{-1}(n)} \xi(m) \wedge \varepsilon(m)$$

$\tau(n)$ is a newly introduced variable. For ease of exposition, we give all formulae in an equivalent prenex normal form, i.e. $Q_1 x_1 \dots Q_n x_n. \varphi$ where Q_i are quantifiers and φ contains no quantifiers.

In the logical annotations, we use two metavariables defined as follows in the context of $\varepsilon(n_i)$:

$$(26) \quad \nabla = \tau(n_i), \Delta = \tau(M(n_i))$$

Thus Δ and ∇ are to logical annotations what \uparrow and \downarrow are to functional annotations in standard LFG.¹³

¹³If we wanted the Neo-Davidsonian semantic representation, we would enrich context-free rules with logical annotations instead of using the ϕ function in the lexicon. For example:

$$(27) \quad \begin{array}{lcl}
 \text{S} & \rightarrow & \begin{array}{cc} \text{NP} & \text{VP} \\ (\uparrow \text{SUBJ}) = \downarrow & \uparrow = \downarrow \\ actor(\Delta, \nabla) & \Delta = \nabla \end{array} \\
 \text{VP} & \rightarrow & \begin{array}{cc} \text{V} & \text{NP} \\ \uparrow = \downarrow & (\uparrow \text{OBJ}) = \downarrow \\ \Delta = \nabla & patient(\Delta, \nabla) \end{array}
 \end{array}$$

Rather than θ -roles, we use what is called "protoroles" (Dowty, 1991), "tectogrammatical roles" (Sgall et al., 1986) or "roles on the action tier" (Jackendoff, 1990) in the literature on semantic analysis of natural languages. Thus $actor(e, x)$ means that x has the role "actor" in the eventuality e .

5.4 Complex Eventualities

In this subsection we apply the formal machinery explicated in the previous subsection to complex eventualities¹⁴ (of which complex predicates are a special case). Consider the sentence *John made Mary cry* with a syntactically formed causative. The sentence is represented by one f-structure (i.e. the f-structures of *made* and *cry* are unified and the corresponding nodes are coheads) with a complex semantic form (with two predicators as can be seen in (29)):

$$(28) \text{ cause}\langle(\uparrow\text{SUBJ}),\text{cry}\langle(\uparrow\text{OBJ})\rangle\rangle$$

The CLF of the sentence *John made Mary cry* is given in (29).

$$(29) \exists e_1, e_2. \text{cause}'(e_1, \text{John}, e_2) \wedge \text{cry}'(e_2, \text{Mary})$$

To create a CLF given in (29), the rule that combines the causative verb (in English *make*) with the main verb is enriched with a logical annotation (that is used instead of the default formula (25)). The morpholexical entries of the verbs and the corresponding rule are given in (30) (φ is the logical form associated with the c-structure node of the main verb; the slash symbolizes substitution):¹⁵

| | ξ M* | τ M* |
|------|---|-----------|
| make | $\text{cause}''(e) \wedge \text{actor}(e, \tau\phi^{-1}(\phi\text{M} * \text{SUBJ}))$ | e |
| cry | $\text{cry}''(e) \wedge \text{actor}(e, \tau\phi^{-1}(\phi\text{M} * \text{SUBJ}))$ | e |

$$(30) \begin{array}{l} \text{VP} \rightarrow \text{VP} \quad \text{V} \\ \quad \quad \uparrow=\downarrow \quad \uparrow=\downarrow \\ \xi : \quad \quad \quad \psi \\ \varepsilon : \quad \quad \Delta = \nabla \text{ patient}(\Delta, \nabla) \end{array}$$

where $\psi = \varphi[\text{actor}(\nabla, \tau\phi^{-1}(\phi\text{M} * \text{SUBJ}))/\text{actor}(\nabla, \tau\phi^{-1}(\phi\text{M} * \text{OBJ}))]$. The morpholexical entries in conjunction with the rule in (30) generate the semantic form in (29).

5.4.1 Complex Eventualities in a Polysynthetic Language

The logical annotation that adjusts the alignment between semantic forms and grammatical functions may get more complicated in polysynthetic languages, such as Aymara (Hardman et al., 2001; Briggs, 1976; Adelaar, 2007; Cerrón-Palomino and Carvajal, 2009; Yapita and Van der Noordaa, 2008). Like many languages with polysynthesis, Aymara has polypersonal agreement, but object marking is forbidden from occurring on nominalized verb forms. The corresponding suffix is attached to the inflected verb instead, as in (31).¹⁶ (In Aymara, clauses are individ-

¹⁴Complex eventualities can be roughly conceived of as a flat representation of Jackendoff's (1990) conceptual structures.

¹⁵While Parson's (1990) notation is equivalent to Davidson's (1967) from the standpoint of representation and reasoning, in case of complex eventualities the former is clearly easier to manipulate within LFG, thus we use it in the examples that follow.

¹⁶This feature is discussed in depth in (Homola and Coler, 2013) whence we take the examples.

uated by morphological marking (Hardman et al., 2001); affirmative clauses, for example, contain exactly one $-w(a)$ suffix, i.e., (31) is a syntactic unit that constitutes a single clause.)

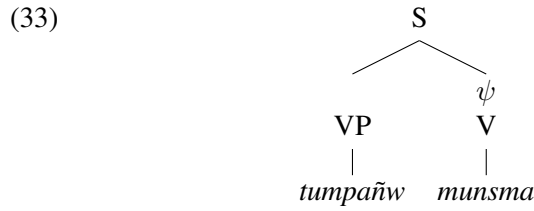
- (31) *Tumpa-ñ-w mun-sma*
 visit-INF-FOC want-SMPL_{1→2}
 “I want to visit you.”

The suffix *-sma* (which is a combined subject/object marker) has the annotation

$$(32) \left[\begin{array}{l} \text{SUBJ} \left[\begin{array}{l} (\text{PRED} \quad \text{'pro'}) \\ \text{PERSON} \quad 1 \end{array} \right] \\ \text{OBJ} \left[\begin{array}{l} (\text{PRED} \quad \text{'pro'}) \\ \text{PERSON} \quad 2 \end{array} \right] \end{array} \right]$$

$$\text{actor}(\Delta, \tau\phi^{-1}(\phi M * \text{SUBJ})) \wedge \text{patient}(\Delta, \tau\phi^{-1}(\phi M * \text{OBJ}))$$

The c-structure with the logical rule annotation is



where $\psi = \varphi[\text{patient}(\triangleright, \tau\phi^{-1}(\phi M * \text{OBJ}))/\text{patient}(\nabla, \tau\phi^{-1}(\phi M * \text{OBJ})]$ and \triangleright refers to the term associated with the sister node (this notation is analogous to Alsina’s (1997) \rightarrow_H). Described informally, the object marked on the finite verb is semantically transferred to the main (infinite) verb.

A very similar example follows:¹⁷

- (34) *Tump-iri-w jut-sma*
 visit-AG-FOC come-SMPL_{1→2}
 “I came to visit you.”

The difference is that the verb *to come* is intransitive, thus it cannot have a (syntactic) object at all. This puzzling example needs a detailed examination at the syntactic level. At the level of semantics, however, the compositional representation above is adequate.

¹⁷Note that the sentence contains exactly one $-w(a)$ suffix, thus it is a single clause.

5.5 An Example

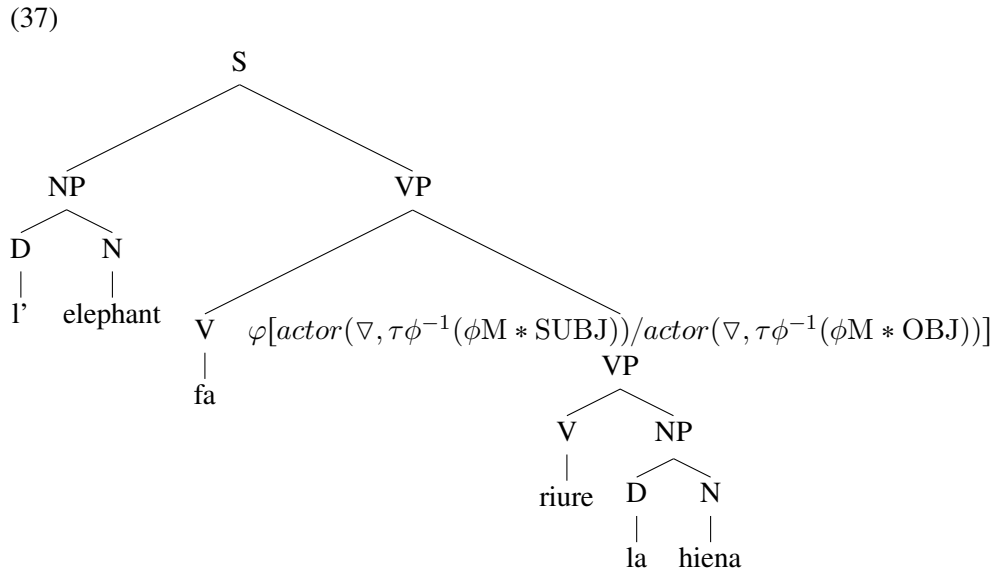
For illustration, we give here an analysis of a sentence taken (in a slightly modified form) from (Alsina, 1997) in the notation of CLFs.

- (35) *L' elephant fa riure la hiena*
 the elephant-MASC make-PRES,3SG laugh-INF the-FEM hyena
 “The elephant makes the hyena laugh.”

The morphological entries are:

| | $(\phi M * \text{PRED})$ | $\xi M*$ | $\tau M*$ |
|------|--------------------------|--|-----------|
| | ‘elefant’ | $elephant(x)$ | x |
| | ‘hiena’ | $hyena(y)$ | y |
| (36) | fer | $cause''(e_1) \wedge actor(e_1, \tau\phi^{-1}(\phi M * \text{SUBJ}))$ $\wedge patient(e_1, \triangleright)$ | e_1 |
| | riure | $laugh''(e_2) \wedge actor(e_2, \tau\phi^{-1}(\phi M * \text{SUBJ}))$ | e_2 |

The analysis yields the following c-structure



The CLF is as follows:

$$\exists x, y, e_1, e_2. elephant(x) \wedge cause''(e_1) \wedge actor(e_1, x) \wedge patient(e_1, e_2) \\ \wedge laugh''(e_2) \wedge actor(e_2, y) \wedge hyena(y)$$

Note that in conjunction with a commonsense theory, such as that of Hobbs (2005), we can directly conclude that

$$\exists y, e_2. laugh''(e_2) \wedge actor(e_2, y) \wedge hyena(y)$$

That is, *The elephant makes the hyena laugh* implies *The hyena laughs*.

6 Conclusions and Further Research

We have discussed the representation and syntactic formation of complex predicates, such as causatives, in the formal framework of LFG. The more conservative solution we have suggested is the use of *E*-unification that operates on semantic forms represented as expressions in term algebras, and as formulae in Wedekind's (1991) attribute-value logic.

As a more radical solution, we showed how a new, more flexible semantic representation (conjunctive logical forms) based on the (Neo-)Davidsonian approach to semantics can supplant the PRED attribute. Although we focused primarily on complex predicates, the latter account has far reaching consequences as it provides a full-fledged semantic framework that can be used to capture entire sentences and even discourse in a well-studied logical formalism.

A few ideas for further research include the following:

- Explore in detail *E*-unification and identify other areas where it could prove useful.
- Investigate the use of conjunctive logical forms in other types of compound sentences within LFG.
- Compare different predicate logical approaches to semantics, such as Hobbs' (1985; 2003) "ontological promiscuity", and identify the best one for the purposes of LFG.

References

- Willem Adelaar. *The Languages of the Andes*. Cambridge University Press, 2007.
- Judith Aissen and David M. Perlmutter. Clause reduction in Spanish. In David M. Perlmutter, editor, *Studies in Relational Grammar 1*, pages 360–403. The University of Chicago Press, 1983.
- Alex Alsina. *The Role of Argument Structure in Grammar: Evidence from Romance*. CSLI Publications, 1996.
- Alex Alsina. A Theory of Complex Predicates: Evidence from Causatives in Bantu and Romance. In Alex Alsina, Joan Bresnan, and Peter Sells, editors, *Complex Predicates*, pages 203–246. 1997.
- Alex Alsina, Joan Bresnan, and Peter Sells. Complex Predicates: Structure and Theory. In Alex Alsina, Joan Bresnan, and Peter Sells, editors, *Complex Predicates*, pages 1–12. 1997.
- Franz Baader and Tobias Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.

- Franz Baader and Wayne Snyder. Unification Theory. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, pages 439–526. 2001.
- Joan Bresnan. *Lexical-Functional Syntax*. Blackwell Textbooks in Linguistics, New York, 2001.
- Lucy Therina Briggs. *Dialectal Variation in the Aymara Language of Bolivia and Peru*. PhD thesis, University of Florida, 1976.
- Miriam Butt. Complex Predicates in Urdu. In Alex Alsina, Joan Bresnan, and Peter Sells, editors, *Complex Predicates*, pages 107–150. 1997.
- R. Cerrón-Palomino and J. Carvajal Carvajal. Aymara. In M. Crevels and P. Muysken, editors, *Lenguas de Bolivia*. Plural Editores, La Paz, Bolivia, 2009.
- Özlem Çetinoğlu, Miriam Butt, and Kemal Oflazer. Mono/bi-clausality of Turkish Causatives. In *Proceedings of the 14th International Conference on Turkish Linguistics*, 2008.
- Mary Dalrymple. *Lexical Functional Grammar*, volume 34 of *Syntax and Semantics*. Academic Press, 2001.
- Mary Dalrymple, John Lamping, and Vijat Saraswat. LFG Semantics via Constraints. In *Proceedings of Sixth Meeting of the European ACL*, 1993.
- Mary Dalrymple, John Lamping, Fernando Pereira, and Vijat Saraswat. Linear logic for meaning assembly. In *Proceedings of Computational Logic for Natural Language Processing*, 1995.
- Donald Davidson. The logical form of action sentences. In *The logic of decision and action*, pages 81–95. University of Pittsburg Press, 1967.
- David Dowty. Thematic Proto-Roles and Arguments Selection. *Language*, 67: 547–619, 1991.
- Jean H. Gallier and Wayne Snyder. Complete Sets of Transformations for General E-Unification. *Theoretical Computer Science*, 67:203–260, 1989.
- Per-Kristian Halvorsen and Ronald M. Kaplan. Projections and Semantic Description in Lexical-Functional Grammar. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell, and Annie Zaenen, editors, *Formal Issues in Lexical-Functional Grammar*, pages 279–292. CSLI, 1995.
- Martha Hardman, J. Vásquez, and J. Yapita de Dios. *Aymara. Compendio de estructura fonológica y gramatical*. Instituto de Lengua y Cultura Aymara, 2001.
- Jerry R. Hobbs. An Improper Treatment of Quantification in Ordinary English. In *Proceedings of the 21rd Annual Meeting of the Association for Computational Linguistics*, pages 57–63, 1983.

- Jerry R. Hobbs. Ontological Promiscuity. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, pages 61–69, 1985.
- Jerry R. Hobbs. Discourse and Inference. MS, Information Sciences Institute, University of Southern California, Marina del Rey, 2003.
- Jerry R. Hobbs. Encoding Commonsense Knowledge. MS, Information Sciences Institute, University of Southern California, Marina del Rey, 2005.
- Petr Homola and Matt Coler. Pragmatic Structures in Aymara. In *Proceedings of the Dependency Linguistics Conference*, 2013.
- Ray Jackendoff. *Semantic Structures*. MIT Press, Cambridge, Massachusetts, 1990.
- Ronald M. Kaplan. The Formal Architecture of Lexical-Functional Grammar. In *Formal Issues in Lexical-Functional Grammar*, pages 7–28. CSLI Publications, 1995.
- Ronald M. Kaplan and John T. Maxwell. LFG Grammar Writer’s Workbench. Technical report, Xerox Corporation, 1996.
- Ronald M. Kaplan and Jürgen Wedekind. Restriction and Correspondence-based Translation. In *Proceedings of the 6th EACL Conference*, pages 193–202, 1993.
- Martin Kay. Functional Grammar. In *Proceedings of the 5th meeting of the Berkeley Linguistics Society*, 1979.
- Martin Kay. Functional Unification Grammar: a formalism for machine translation. In *Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics*, 1984.
- Donald Knuth and Peter Bendix. Simple Word Problems in Universal Algebras. pages 263–297, 1970.
- Tara Mohanan. Multidimensionality of Representation: NV Complex Predicates in Hindi. In Alex Alsina, Joan Bresnan, and Peter Sells, editors, *Complex Predicates*, pages 431–472. 1997.
- Terence Parsons. *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press, 1990.
- Barbara H. Partee. Lexical Semantics and Compositionality. In Lila R. Gleitman and Mark Liberman, editors, *Invitation to Cognitive Science. Language*, volume 1, pages 311–360. MIT Press, 1995.
- Paul M. Pietroski. *Events and Semantic Architecture*. Oxford University Press, 2005.

Petr Sgall, Eva Hajičová, and Jarmila Panevová. *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. D. Reider Publishing Company, 1986.

Hans Uszkoreit. Categorical Unification Grammars. In *Proceedings of COLING*, 1986.

Jürgen Wedekind. Classical Logics for Attribute-Value Languages. In *Proceedings of the 5th EACL conference*, pages 204–209, 1991.

Juan de Dios Yapita and J.T. Van der Noordaa. *La dinámica aymara. Conjugación de verbos*. Instituto de Lengua y Cultura Aymara, 2008.