Complex predicates: an LFG+glue analysis

John J. Lowe
University of Oxford

ABSTRACT

In this paper I discuss weaknesses in the traditional LFG account of complex predicates and in the XLE implementation of the same. I argue that the concept of predicate composition in general, and the mechanisms required to achieve it, are problematic, but that the most problematic element is the concept of argument fusion. I show that a semantically-integrated account of complex predicate formation is possible within LFG + glue, an account which provides a simple and effective formalization of argument fusion, and which does not suffer from the weaknesses of traditional approaches.[1]

Keywords: complex predicates, Lexical Functional Grammar, argument structure, glue semantics

1 INTRODUCTION

Complex predicates present a challenge to any lexicalist theory of syntax since, in at least some languages, there is clear evidence that a single clausal predicate can result from a *syntactic* process involving two or more distinct lexical elements (usually a lexical verbal or nominal element, and one or more 'light' verbal elements). The resulting

[1] I am very grateful to the attendees of the Oxford Glue Group, 8 May 2015, and the audience at the 18th SE-LFG meeting, SOAS, 31 October 2015, where versions of this work were presented, for their attention and valuable comments, in particular to Mary Dalrymple, Avery Andrews, Jamie Findlay, Louisa Sadler, Kersti Börjars, Andrew Spencer, and Peter Austin. I am also grateful to Anna Kibort, whose paper on causatives at the 16th SE-LFG meeting, SOAS, 21 February 2015, first got me thinking about complex predicates and glue. Naturally all errors are my own. This work was undertaken while I was in receipt of an Early Career Research Fellowship funded by the Leverhulme Trust.

predicate functions as if it were a single lexical element, but its formation within the syntax belies this. In this paper I discuss a number of approaches to complex predicate formation within the strict lexicalist theory of Lexical Functional Grammar (LFG; Kaplan and Bresnan 1982; Bresnan 2001; Falk 2001), and show that all suffer from theoretical, and in some cases even empirical, weaknesses. I then present an analysis within LFG augmented with glue semantics (LFG + glue; e.g. Dalrymple 2001; Asudeh 2012), which overcomes the weaknesses in previous approaches and even has the potential to account for data which is problematic for previous accounts.

Early work on complex predication within LFG proposed either a multiclausal syntactic analysis similar to raising (e.g. Ishikawa 1985), or an essentially lexical analysis, whereby complex predicates are formed from their constituent parts inside the lexicon (e.g. Kaplan and Wedekind 1993; Ackerman and Webelhuth 1996, 1998). However, authors such as Mohanan (1994), Butt (1995) and Alsina (1996) demonstrated beyond reasonable doubt that some languages attest complex predicates which are syntactically monoclausal, yet must be analysed as formed in the syntax. From an LFG perspective, the challenge in modelling such a phenomenon lies in the process of predicate formation, in particular in the merger of distinct semantic forms, since semantic forms are in principle not manipulable in syntax, and in the fusion and linking of the arguments of merged predicates. Since the early work of Butt (1995) and Alsina (1996), there has been a wealth of research on complex predicate formation as a syntactic phenomenon within LFG, in particular by Miriam Butt and her colleagues.[2] Two main formal approaches have developed: one that now might reasonably be called the 'traditional' LFG approach, which seeks to integrate the analysis of complex predicates with work on argument structure and 'linking theory', and a somewhat different approach which is utilized in the computational implementation of LFG, XLE (Crouch *et al.* 2011). Relatively little work has been done, however, on how semantics interacts with the syntax and argument structure of complex predicate formation; the exceptions are Kaplan and Wedekind (1993), Dal-

[2] See e.g. Butt (1997), Butt and Geuder (2001), Butt *et al.* (2003), Butt and Ramchand (2005), Butt and King (2006), Butt *et al.* (2010), Ahmed and Butt (2011), Raza (2011), Ahmed *et al.* (2012), Butt *et al.* (2012), Sulger (2012), and Butt (2014).

rymple *et al.* (1993a), Andrews and Manning (1999), Andrews (2007), and Homola and Coler (2013).[3] In particular, there exists no account of complex predicates within standard architectural assumptions and in the current standard 'new' glue format. Recent work in LFG + glue, e.g. by Asudeh and Giorgolo (2012), has shown that glue semantics is able to do a lot of the work traditionally attributed to argument structure; one aim of this paper is to show that this holds also for complex predication.

In the next section I show that neither of the main approaches to complex predicate formation in LFG provides an entirely satisfactory analysis of predicate composition or argument merger. In §3 I argue that a semantically integrated account is more satisfactory; in §4 I show that my proposal can not only deal with some of the most complex phenomena that previous accounts can, but that it even has the potential to deal with phenomena that are problematic for previous accounts. In §5 my proposal is compared with previous proposals for a semantic account of complex predicates in LFG. In §6 I draw my conclusions.

2 THE STANDARD ACCOUNTS

As mentioned in the previous section, there are two approaches that might be considered the standard approaches to complex predicates in LFG. This is not to say that there are two *competing* approaches, or that it is a case of some authors advocating one approach over the other. Rather, the two approaches are used in different contexts, even by the same authors. For example, Butt (2014) provides one of the most elegant and fully formulated accounts of what I will refer to as the 'linking' approach, which builds on much of her previous work, but at the same time Butt has been at the forefront of developing

[3] Current work in XLE does not attempt to integrate glue semantics, or any theory of the syntax-semantics interface, into the implementation. Functional means of dealing with semantic representations are available, by means of the f-structure LEX-SEM feature or by means of f-structure rewriting (Crouch and King 2006), but these permit no active role for semantics in the grammar. The absence of a semantically integrated account of complex predicates within XLE does not therefore have anything specifically to do with complex predicates but is merely a feature of the XLE implementation at the present time.

the XLE treatment of complex predicates within the context of the Urdu PARGRAM grammar (Butt *et al.* 1999, 2002; Butt and King 2007; Sulger *et al.* 2013).

The very fact that the computational implementation of LFG does not include a full formalization of the linking approach raises something of a question mark over both approaches, in particular over the lack of formalization of the linking approach, and over the analytical accuracy of the XLE approach.[4] In this section both approaches are described, focusing initially on those aspects that both approaches share, and then drawing out the ways in which they differ. The description of the linking theory approach is based on the recent account of Butt (2014).

The phenomenon in question is exemplified in (1) and (2):[5] (1) shows a simple transitive sentence in Urdu with the verb *likh* 'write', while (2) shows a sentence involving a complex predicate formed of the verb *likh* 'write' and the 'permissive' light verb *de*.[6]

(1) *saddaf-ne* *ciṭṭhii* *likh-ii*
Saddaf-ERG note.NOM.F.SG write-PERF.F.SG
'Saddaf wrote a note.' (Urdu)

(2) *anjum-ne* *saddaf-ko* *ciṭṭhii* *likh-ne*
Anjum-ERG Saddaf-DAT note.NOM.F.SG write-INF.OBL
d-ii
let-PERF.F.SG
'Anjum let Saddaf write a note.' (Urdu)

As Butt and other authors have demonstrated, Urdu complex predicates such as that in (2) are monoclausal at f-structure but consist of two predicating elements, each with their own argument structures. Light verbs can combine productively and recursively with most verbal, and many nominal, forms, such that their combination must be treated syntactically, not lexically.

[4] The reasons for the differences between the two approaches are discussed by Butt *et al.* (2010, 249–250); they boil down to the desire for computational efficiency within XLE.

[5] The examples are from Butt (2014).

[6] The following abbreviations are used in the glosses: CAUS 'causative', DAT 'dative', ERG 'ergative', F 'feminine', INF 'infinitive', INSTR 'instrumental', M 'masculine', NOM 'nominative', OBL 'oblique', PERF 'perfect', SG 'singular'.

Under the linking approach to complex predicates, the lexical entry for the verb *likh* 'write' is assumed to contain the semantic form specification in (3), while the lexical entry for the light verb *de* 'let' is assumed to contain the semantic form specification in (4).

(3) $(\uparrow \textsc{pred}) = \text{`write} \langle \underset{[-\textsc{o}]}{\textsc{agent}}, \underset{[-\textsc{r}]}{\textsc{theme}} \rangle\text{'}$

(4) $(\uparrow \textsc{pred}) = \text{`let} \langle \underset{[-\textsc{o}]}{\textsc{agent}}, \underset{[+\textsc{r}]}{\textsc{goal}}, \%\textsc{pred} \rangle\text{'}$

In these semantic forms, the verb forms concerned subcategorize for arguments which are defined by reference to the semantic role of the argument and by reference to one of the features ±O or ±R, which constrain the mapping between semantic roles and grammatical functions according to the principles of Mapping Theory (Bresnan and Kanerva 1989). The specifics of the argument structure model assumed, and the details of Mapping Theory, are not important for the present purposes; the representations of Butt (2014) are adopted here, but e.g. all the semantic forms and argument structure representations presented in this paper could easily be rewritten in the model of Kibort (2001, 2004, 2006, 2007, 2008), and no significant differences would result.

What is important is that these semantic forms must fuse in the formation of the f-structure, with the semantic form of the lexical verb supplying the value of the %PRED variable in the argument structure of the light verb. This process of fusion is discussed in more detail in the rest of this section; at this point it suffices to say that the selected semantic roles are associated with grammatical functions, and that a single predicate, with a single subcategorization frame, results. This can be seen in the PRED value in (5), which shows the resulting f-structure for the clause in (2).

(5) $$\begin{bmatrix} \textsc{pred} & \text{`let-write}\langle \textsc{subj}, \textsc{obj}_{\text{goal}}, \textsc{obj}\rangle\text{'} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{`Anjum'} \end{bmatrix} \\ \textsc{obj}_{\text{goal}} & \begin{bmatrix} \textsc{pred} & \text{`Saddaf'} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{`note'} \end{bmatrix} \end{bmatrix}$$

The first theoretical weakness of the linking theory approach, a feature shared with the XLE approach, is in the mechanism of predicate fusion. A fundamental assumption of early lexicalist syntax was the principle of Direct Syntactic Encoding, i.e. the principle that lexical properties such as argument structure should not be manipulable in the syntax. This plays out in LFG in the fact that, at least originally, semantic forms are not manipulable in the syntax. As clearly demonstrated by Mohanan (1994), Butt (1995) and Alsina (1996), however, complex predicates require a syntactic explanation, and in this respect, at least, the principle of Direct Syntactic Encoding cannot be maintained. Under the linking and XLE approaches to complex predication an exception to the non-manipulability of semantic forms must be made, since there is no other way for predicates to compose, and the variable %PRED utilized in the semantic forms for light verbs (as in (4)) was adopted as a means of manipulating semantic forms outside the lexicon. The variable %PRED is therefore an augmentation of the original LFG system which, though apparently necessary, significantly increases its power, and is required purely to account for complex predicates. If %PRED, and manipulable semantic forms in general, could be eliminated, this would be theoretically advantageous in restricting the power of the LFG formalism and reducing the number of construction-specific devices required.

A further problem with predicate fusion is the mechanism required to actually get the information supplied by the embedded semantic form inside the semantic form of the light verb, i.e. precisely how a semantic form such as that in (6) gets instantiated as (7).[7]

(6) 'let ⟨ AGENT, GOAL, %PRED ⟩'

(7) 'let ⟨ AGENT, GOAL, 'write ⟨ AGENT, THEME ⟩' ⟩'

Most recent discussions of complex predication that are based within the linking approach brush over the explicit formalization of this process. In early work, Butt (1995) and Alsina (1996, 1997) do provide formalized accounts of the process. Butt's (1995) account ne-

[7] An instantiation as in (7) is usually represented in an f-structure in resolved form, that is with a single 'fused' predicate with a single subcategorization frame, and with subcategorization for semantic roles replaced by subcategorization for grammatical functions, as shown in (5).

cessitates assuming a distinction between two types of semantic form, one type (found with light verbs) which is incomplete on its own and requires that it be unified with a standard, complete, semantic form. In addition, the usual $\uparrow = \downarrow$ f-description must be reinterpreted such that it licenses the composition of semantic forms where necessary. Alsina's (1997) proposal is similar, except that the alternative interpretation of $\uparrow = \downarrow$ is associated with a new function $\uparrow =_H \downarrow$, and the precise formulation of the composition is stated in somewhat different terms. Both accounts involve augmentations of the standard LFG model, thereby increasing its power and, as argued by Andrews and Manning (1999), the proposals are either under-formalized in certain respects, or else there are difficulties with the formalizations involved. In any case, neither proposal appears to have been widely adopted, at least explicitly, in recent work within the linking approach.

Besides these early proposals, the only remaining available formalization is that proposed by Butt *et al.* (2003) for XLE, and in the following I assume that this formalization holds also for the linking approach.[8] For the XLE approach, it is necessary to assume that semantic forms can be decomposed into their constituent parts. In particular, the feature $\textsc{arg}_x$ can be used to refer to argument positions inside the PRED feature. That is, for example, the constituent parts of the semantic form in (6) can be referred to by the schema:

(8) $\text{‘}\textsc{fn}\ \langle\ \textsc{arg}_1,\ \textsc{arg}_2,\ \textsc{arg}_3\ \rangle\text{’}$

such that for any f-structure for which (6) provides the PRED, the %PRED variable can be referred to by the path PRED $\textsc{arg}_3$. Then, via a phrase structure rule such as that in (9), the PRED of a lexical verb can be identified with the %PRED slot in the PRED value of a light verb.[9]

[8] Butt *et al.* (2010) discuss the following details, in particular the use of the restriction operator, as specifically part of the XLE approach to complex predicates and not as part of the linking approach. However, as stated, no standard formalization exists for the linking approach (Butt *et al.* make no mention of what they assume) such that, to the extent that one wants to be able to formalize predicate fusion in the linking approach, one is essentially constrained to make use of the XLE mechanisms.

[9] This rule has been simplified for the purposes of exposition; a more detailed version is given in (11).

(9) $\mathrm{V} \rightarrow \quad \underset{\substack{\downarrow \backslash \mathrm{PRED} = \uparrow \backslash \mathrm{PRED} \\ (\uparrow \mathrm{PRED\ ARG_3}) = (\downarrow \mathrm{PRED})}}{\mathrm{V_{lex}}} \quad \underset{\uparrow = \downarrow}{\mathrm{V_{light}}}$

This works, but it suffers from the same problem that we have seen already with regard to the %PRED variable: the $\mathrm{ARG_x}$ feature is required specifically to account for predicate composition, and its purpose is to enable the manipulation of an otherwise non-manipulable element of f-structure, the semantic form.[10] Furthermore, this analysis must make use of the restriction operator \, as seen in (9). The restriction operator was introduced by Kaplan and Wedekind (1993), who provide the following definition:

(10) If f is an f-structure and a is an attribute:
$f \backslash a = f|_{\mathrm{Dom}(f)-\{a\}} = \{< s, v > \in f | s \neq a\}$

Informally, the f-structure $f \backslash a$ is identical to the f-structure that results from removing the attribute a from the f-structure f. This operation is a fundamental part of both the linking theory and XLE analyses of complex predication, since they seek to represent the fact that both lexical and light verb elements are co-heads of the clausal f-structure, even though some attributes of the clausal f-structure have different values from those required by the lexical verb. One of these attributes is PRED, as seen in (9): the PRED of the lexical verb's f-structure serves as an argument inside the PRED of the light verb (and thereby the clause), so the two are necessarily not the same. In the simplified phrase-structure rule given in (9), only one restriction is stated, but full treatments require considerable use of restriction. For example, Butt *et al.* (2003, 99) provide the following rule for complex predication in Urdu (explicitly for the XLE approach):

(11) $\mathrm{V} \rightarrow \quad \underset{\substack{\downarrow \backslash \mathrm{PRED} \backslash \mathrm{SUBJ} \backslash \mathrm{VTYPE} \backslash \mathrm{LEX\text{-}SEM} = \\ \uparrow \backslash \mathrm{PRED} \backslash \mathrm{SUBJ} \backslash \mathrm{OBJ_{goal}} \backslash \mathrm{VTYPE} \backslash \mathrm{LEX\text{-}SEM} \\ (\uparrow \mathrm{PRED\ ARG_3}) = (\downarrow \mathrm{PRED}) \\ (\uparrow \mathrm{OBJ_{goal}}) = (\downarrow \mathrm{SUBJ})}}{\mathrm{V}} \quad \underset{\uparrow = \downarrow}{\mathrm{V_{light}}}$

[10] The FN feature has found more widespread use, but both are rendered unnecessary for any phenomenon under the proposals made in §3.

Restriction is a well-defined set-theoretic operation, and is not in principle to be avoided. Bresnan apud Butt *et al.* (2010, 253) questions the use of the restriction operator on theoretical grounds, since it potentially endangers the Principle of Direct Syntactic Encoding by permitting grammatical functions to be changed in the syntax; this is really the same problem we have seen already with the other aspects of the formalization of predicate composition. A more specific problem is that it may cause inside-out functional uncertainty to fail (Andrews 2001, and p.c.).[11] In any case, an analysis that can dispense with restriction is perhaps to be preferred over one that cannot do so purely on grounds of simplicity.

In fact, the use of restriction has some not entirely desirable consequences. The intuition behind the use of the restriction operator here is, as mentioned, that both the lexical verb and the light verb are co-heads of the clausal f-structure. This is a key part of the important observation that such complex predicates are monoclausal at f-structure. Nevertheless, while equations of the type $\downarrow \backslash \textsc{pred} = \uparrow \backslash \textsc{pred}$ do in some sense permit the lexical verb to function as a co-head, they also specify the existence of a separate f-structure, of which the lexical verb alone is the head. That is, e.g., for the sentence in (2), alongside the f-structure in (5) there must also exist that in (12), which represents the f-structure for the lexical verb alone.

(12) $$\begin{bmatrix} \textsc{pred} & \text{‘write}\langle\textsc{subj}, \textsc{obj}\rangle\text{’} \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{‘Saddaf’} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{‘note’} \end{bmatrix} \end{bmatrix}$$

Butt *et al.* (2003, 101) refer to the full clausal f-structure of a complex predicate, such as that in (5), as representing “the final analysis”, implying that the separate f-structure for the lexical verb is somehow preliminary and not independently part of the final analysis. However, by the phrase-structure rules and f-descriptions that specify both clausal and lexical verb f-structures, there is no sense in which one f-structure is in any sense subordinate to, or subsumed or rendered

[11] Recent work by Homola and Coler (2013) sets out explicitly to eliminate the need for restriction in the analysis of complex predicates; I will discuss this in more detail below.

superfluous by, the other. Both exist, side by side, sharing all features not restricted out, but potentially differing in respect of the restricted features. This means that, although it is a fundamental assumption of the linking and XLE approaches to complex predicates that the lexical verb – light verb complex is monoclausal at f-structure, the only widely utilized and fully formalized analysis of this in LFG requires that there are in fact two f-structures (contrary to the original analyses of Butt 1995 and Alsina 1996, 1997). It is worth remarking that the only real value of the restriction operator here is to permit these two f-structures to exist side by side, rather than one embedded inside the other. That is, if one were prepared to permit the f-structure for the lexical verb to be embedded inside the f-structure for the clause, it would in principle be possible to do away with the restriction operator. For example, a phrase-structure rule such as that in (13) would produce an f-structure such as (14) for the sentence in (2).[12]

(13)
$$\begin{array}{lccc} \mathrm{V} & \rightarrow & \begin{array}{c} \mathrm{V} \\ (\uparrow \textsc{ep}) = \downarrow \\ (\uparrow \textsc{pred arg}_3) = (\downarrow \textsc{pred}) \\ (\uparrow \textsc{obj}_{\text{goal}}) = (\downarrow \textsc{subj}) \\ (\uparrow \textsc{obj}) = (\downarrow \textsc{obj}) \end{array} & \begin{array}{c} \mathrm{V}_{\text{light}} \\ \uparrow = \downarrow \\ \\ \\ \\ \end{array} \end{array}$$

(14)
$$\left[\begin{array}{ll} \textsc{pred} & \text{‘let-write}\langle \textsc{subj}, \textsc{obj}_{\text{goal}}, \textsc{obj} \rangle\text{’} \\ \textsc{subj} & \left[\begin{array}{ll} \textsc{pred} & \text{‘Anjum’} \end{array} \right] \\ \textsc{obj}_{\text{goal}} & \left[\begin{array}{ll} \textsc{pred} & \text{‘Saddaf’} \end{array} \right] \\ \textsc{obj} & \left[\begin{array}{ll} \textsc{pred} & \text{‘note’} \end{array} \right] \\ \textsc{ep} & \left[\begin{array}{ll} \textsc{pred} & \text{‘write}\langle \textsc{subj}, \textsc{obj} \rangle\text{’} \\ \textsc{subj} & [\;] \\ \textsc{obj} & [\;] \end{array} \right] \end{array} \right]$$

The similarity with a raising analysis of complex predication is obvious. But, as stated, the fundamental assumption of these approaches is that a raising-like analysis involving a multiclausal f-structure is not appropriate, since there is very good evidence for monoclausality.

[12] As in (5), with the PRED value shown in resolved, i.e. ‘fused’, form.

However, as long as the lexical verb's f-structure is not directly subcategorized for by the light verb (hence the use of the ad hoc EP in (14), standing for 'embedded predicate', rather than e.g. COMP), and as long as all the arguments of the complex predicate appear in the clausal f-structure by virtue of the identification of the lexical verb's PRED with an argument of the light verb's PRED, it could be argued that the evidence for monoclausality does not in principle exclude the embedding of an f-structure for the lexical verb inside the clausal f-structure. That is, if the outer f-structure in (5) is f, the outer f-structure in (14) is g, and the attribute EP is e, then $f = g \backslash e$, and any evidence for monoclausality can be explained by reference to $g \backslash e$ just as easily as it can by reference specifically to f. In other words, if the lexical verb must head its own f-structure, it does not really matter whether this f-structure appears inside the clausal f-structure, as in (14), or alongside it, as the linking/XLE analyses assume. Details aside, embedding is essentially the approach taken by Andrews and Manning (1999), whose proposals involve the embedded predicate appearing as the value of an f-structure feature ARG.

Given the evidence for monoclausality, it would be preferable if the analysis could eliminate the need for a separate f-structure headed by the lexical verb altogether. As discussed, the f-structure for the lexical verb in a complex predicate is not treated as part of the "final analysis". The assumption of such an f-structure is, in terms of the syntax, at least, little more than a technical necessity for the linking/XLE approaches to be able to account for predicate composition. On the other hand, there may be *semantic* difficulties with assuming only a single f-structure. This is discussed in more detail in §4, but at this point one may draw the conclusion that if multiple f-structures are necessary, there seems to be little gained by using restriction when all it achieves is disconnecting those f-structures from one another.

Thus far, I have avoided detailed discussion of the arguments of complex predicates. Beside the process of predicate, and f-structure, composition, this is the second major question mark over the linking/XLE analyses of complex predication in LFG. It is also apparently the most problematic, since while there do exist formal accounts of predicate composition within LFG (however problematic), there exists no comparable formalization of argument fusion. Although it is an aim of this paper to provide a general treatment of complex predicates, the primary aim is to show that the hitherto unformalized process of argument fusion receives a formally elegant account when treated within LFG + glue.

The issue is how in (7), for example, the AGENT argument of the lexical verb *likh* 'write' is fused with the GOAL argument of the light verb, such that only the single resulting fused argument undergoes mapping to a grammatical function (i.e. with the result that there are only three arguments of the complex predicate 'let-write', rather than four; cf. (2) and (5)). At this point, the linking and XLE approaches go their separate ways. As for the linking approach, there has been considerable work on the argument structure relations involved, and how the arguments resulting from the fusion of two predicates map correctly to their respective grammatical functions. Generalizations have also been stated on which arguments may fuse: e.g. Butt (1995, 1998) proposes that the lowest matrix argument must be identified with the highest embedded argument. In terms of the actual process of argument fusion, however, I am aware of no explicit account within the linking approach, even in the most recent work by Butt (e.g. 2014). As for XLE, argument fusion is simply avoided.

In (1), *saddaf-ne* is the agent and the subject (or [−O] argument in linking theory terms). However, in (2), the equivalent argument is still the agent of the event of writing, but it now surfaces as OBJ_{goal} in the f-structure. That is, the argument structures for the predicates of the two examples are respectively:

(15) 'write' ⟨ AGENT THEME ⟩

	AGENT	THEME
	[−O]	[−R]
	SUBJ	OBJ

(16) 'let' ⟨ AGENT GOAL 'write' ⟨ AGENT THEME ⟩⟩

	AGENT	GOAL	AGENT	THEME
	[−O]	[+O]	([−O])	[−R]
	SUBJ	OBJ_θ		OBJ

The problem is how the AGENT of the lexical verb is fused with the GOAL argument of the light verb, resulting in an argument that maps to OBJ_θ. In the case in question the fused argument adopts the properties of the light verb's argument: it adopts the [+O] of the light verb's

GOAL argument, and not the [−O] of the lexical verb's AGENT, meaning that it can map to an object function (here OBJ_θ). As stated, in linking approaches to complex predication, there is no explicit account of how this happens, even though it is a fundamental element of the approach.

That the argument fusion assumed in the linking approach is a badly underformalized notion is evident from the fact that, as mentioned, the XLE approach is rather different. In XLE there is no such thing as argument fusion. While linking accounts of complex predication consistently assume that a light verb such as Urdu *de* 'let' (and similar light verbs, such as causatives) is a three-place predicate, subcategorizing for two thematic arguments (for *de* an AGENT and a GOAL) and one predicate argument, in XLE such light verbs are two-place, subcategorizing for only one thematic argument and one predicate argument. For example, in XLE the lexical entry for Urdu permissive *de* will include the following (Butt *et al.* 2003, 99):

(17) $(\uparrow \text{PRED}) = \text{'let}\langle(\uparrow \text{SUBJ}), \%\text{PRED2}\rangle\text{'}$

Since the light verb here introduces only one thematic argument, for a complex predicate such as 'let-write' there is no need for argument fusion, since only three thematic arguments are introduced by the separate verbs: one by the light verb, two by the lexical verb. All that is needed is for the grammatical function of the lexical verb's SUBJ to be changed as appropriate when it appears in the clausal f-structure; this is achieved by f-descriptions such as $(\uparrow \text{OBJ}_{\text{goal}}) = (\downarrow \text{SUBJ})$, as seen in (11).[13] That this is very different from the linking approach to complex predicates is noted by Butt and King (2006), who point out that the XLE approach is closer to some Minimalist analyses of complex predication.

A further feature of both the linking and XLE approaches to complex predication is that neither involves an explicit account of the se-

[13] It is a further weakness of the XLE approach that this f-description has to appear as an annotation in the phrase-structure rule under the lexical verb's V, rather than under the light verb's V. In principle, one would expect the specification to be associated with the light verb; indeed, the grammatical function of the argument depends on the light verb, since while e.g. Urdu *de* 'let' requires the lexical verb's SUBJ to appear as OBJ_θ, another light verb might require it to appear as an OBL_θ.

mantic aspect.[14] In the following section, I develop an alternative approach to complex predication, which makes use of glue semantics not only to provide a proper semantic analysis of complex predication, but also to overcome the weaknesses of the linking/XLE approaches.

3 PROPOSAL

It has long been recognized that the resource sensitivity of glue semantics has the potential to capture a number of constraints that must otherwise be dealt with at other levels of structure.[15] In particular, the resource sensitivity of glue means that the principles of COMPLETENESS and COHERENCE, traditionally treated as well-formedness constraints on f-structure, are captured at the level of semantics, rendering them superfluous as f-structure constraints. This means the subcategorization frame traditionally assumed as part of an f-structure PRED feature is unnecessary: subcategorization can be dealt with almost entirely within the semantics (Kuhn 2001; Asudeh 2004, 2012; Asudeh and Giorgolo 2012).[16]

Therefore, the subcategorization requirements of a complex predicate, and the process of argument fusion, however understood, can be dealt with in the semantic representation. This permits an immediate simplification of the syntactic representation: it is no longer necessary to assume predicate composition in the f-structure, since the purpose of predicate composition is essentially to enable the combination of the subcategorization requirements of both the lexical verb and the light verb in the same PRED feature. At a stroke, manipulable PRED features, the %PRED variable, and the restriction operator are all rendered unnecessary. So in place of the phrase structure rule in (9), it is sufficient for the present purposes to assume the phrase structure rule in (18) for complex predicates in Urdu.

(18) $$V_{(\text{lex})} \quad \rightarrow \quad \underset{\uparrow = \downarrow}{V_{\text{lex}}} \quad \underset{\uparrow = \downarrow}{V_{\text{light}}}$$

[14] Cf. fn. 3.

[15] The earliest recognition of this may be by Kaplan apud Dalrymple *et al.* (1993a, 14); see also Kuhn (2001) and Andrews (2008).

[16] Non-semantic arguments cannot be dealt with in the semantics, but they can still be handled without recourse to subcategorization in semantic forms.

That is, the lexical verb and light verb are genuine co-heads, reflecting the original intuition regarding the construction. Under this analysis, if we wish to avoid the difficulties of predicate composition, only one verb can supply a PRED value. Since complex predicates can be recursively embedded under light verbs to form new predicates, the PRED must be supplied by the lexical verb. Light verbs may then contribute only features.[17] For example, rather than the f-structure in (5), I assume for the present an f-structure as in (19), based on lexical specifications as in (20) and (21).[18]

(19) $$\begin{bmatrix} \text{PRED} & \text{'write'} \\ \text{PERMISSIVE} & + \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'Anjum'} \end{bmatrix} \\ \text{OBJ}_{\text{goal}} & \begin{bmatrix} \text{PRED} & \text{'Saddaf'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'note'} \end{bmatrix} \end{bmatrix}$$

(20) *likh*: $(\uparrow \text{PRED}) = \text{'write'}$

(21) *de*: $(\uparrow \text{PERMISSIVE}) = +$

The point is that once subcategorization is removed from semantic forms, and given that the existence of a separate semantic representation eliminates any requirement for semantic forms to reflect semantic content, a light verb need contribute no more than (and perhaps not even as much as) an f-structure feature specifying permission, or

[17] In fact, they need not even contribute features, if there are no syntactic operations that would require reference to such features. In the analysis proposed here it is assumed that Urdu light verbs do contribute features, but this is assumed largely to make the f-structure representations clearer, and I make no firm claims as to whether such features are strictly necessary.

[18] The PRED value and subcategorized grammatical functions are not the only features that necessarily show different values for the lexical and light verb. In (11), for example, one of the features restricted out is VTYPE, since the light verb is finite and the lexical verb an infinitive. The solution for any such feature will depend on the function that it has in the wider grammar, but none should be impossible to deal with. In the case of VTYPE, for example, it would be possible to deal with this at 'morphological structure' (Butt *et al.* 1996, 1999), i.e. in just the same way as monoclausal auxiliary sequences in English.

causation, etc.: it no longer needs to contribute anything to the clausal PRED itself.[19]

This observation is not, in fact, new. Dalrymple *et al.* (1993a) point out that:

> If the only remaining function of the PRED is to ensure predicate uniqueness, it would do as well to assume that the PRED value for a sentence with a complex predicate is contributed by the main verb..., and that the function of [a light verb such as] LET is to modify the argument structure but not to contribute to or change the PRED value of the construction.

Dalrymple *et al.* (1993a) still assume complex PRED features of the form 'PERMIT⟨WRITE⟩', but they make no claims as to how they would be formed, and they assume such features perhaps only for the sake of greater consistency with existing accounts. Nevertheless, Dalrymple *et al.*'s important insight has been essentially ignored in work in both the linking and XLE approaches (presumably because these approaches tend to lack an explicit semantic angle), and it is well worth re-emphasizing.

At least superficially, the problematic concept of 'argument fusion' is more difficult to address, and it is here that the value of a glue-based approach becomes apparent. The problem essentially boils down to the question of how arguments are recategorized when they appear inside a complex predicate. Assuming that subcategorization is not dealt with in the f-structure, but only in the semantics, let us consider how a very simple glue semantic account might fare. A standard glue treatment of verbal meaning might assume the following meaning constructor for *likh* 'write' (assuming a very simple event semantics, making use of an event variable

[19] That is, following the Dalrymple *et al.* quote provided, I assume that the only important property of PRED features is their unique instantiation, which serves to distinguish any two f-structures that have PRED features; the value itself is unimportant. Thus it does not matter that the PRED value in (19) does not reflect the meaning of the complex verb (since the value is 'write' but the meaning of the full predicate is 'let write'). This was relevant only in pre-glue LFG, but is superfluous in LFG + glue, since semantic content is represented separately from the f-structure. What function PRED values do serve in LFG + glue is a matter for debate; see Andrews (2008) for discussion.

but ignoring temporal variables usually assumed in more elaborate treatments of event semantics in glue, e.g. Fry 2005, Haug 2008, Lowe 2015):

(22) $\lambda y.\lambda x.\lambda e.write(e) \wedge agent(e,x) \wedge theme(e,y) : (\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

Meaning constructors such as this render subcategorization in the f-structure, and thereby also completeness and coherence as f-structure well-formedness constraints, superfluous, since the glue expression ensures that only a SUBJ and an OBJ, and no other feature, can and must appear as governable grammatical functions in the f-structure headed by the verb, else an incoherent semantics would result. But what this meaning constructor also does is effectively tie the agent of the event of writing to the grammatical function SUBJ, and the theme of the event of writing to the grammatical function OBJ. This is fine for the simplex verb, but when embedded under a complex predicate the SUBJ should not be the agent of the event of writing: it will either have no thematic relation to the event of writing, or a relation of 'permitter', depending on how we choose to model the semantics of the light verb.

Asudeh and Giorgolo (2012) and Asudeh *et al.* (2014), in their glue-based approach to argument structure and valency alternations, propose meaning constructors of slightly different form from the sort in (22), but the basic problem is the same. In their approach, f-structural grammatical functions such as SUBJ and OBJ are linked with s-structure features labelled ARG_1, ARG_2, etc., via f-descriptions in the lexical entries of verbs.[20] So, the equation in the third line of the lexical entry in (23) identifies the semantic structure projected from the verb's SUBJ with an s-structure labelled ARG_1 in the s-structure projected from the verb's f-structure. Then, the glue expressions in the meaning constructor for the verb make reference to the s-structure features ARG_1, etc., and do not make direct reference to grammatical functions like SUBJ.

[20] S-structure features ARG_1 etc. are not related to the f-structure ARG_x features discussed in §2.

(23) 'write' V

$(\uparrow \text{PRED}) = \text{`write'}$
$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBJ})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

$\lambda y.\lambda x.\lambda e.write(e) \wedge agent(e,x) \wedge theme(e,y)$:
$(\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

Nevertheless, from the current perspective, it remains the case that e.g. ARG_1 in the meaning constructor in (23) is tied to the agent of the act of writing (y on the meaning side), and by the equation $(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$ this is tied to the grammatical function SUBJ.

The problem remains also in Findlay's (2014) fusion of Asudeh and Giorgolo's (2012) proposals with Kibort's (2001; 2004; 2006; 2007; 2008) model of argument structure (briefly detailed in Asudeh *et al.* 2014, 75–77), even though Findlay's model provides for greater flexibility in the association between grammatical functions and s-structure ARG_x features. In Findlay's model, the equation $(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$ in (23) would be replaced by the following equation (adopting the notation of Asudeh *et al.* 2014, 76 and omitting the use of templates):

(24) $\{(\uparrow \{\text{SUBJ} \mid \text{OBL}_\theta\})_\sigma = (\uparrow_\sigma \text{ARG}_1) \mid (\uparrow_\sigma \text{ARG}_1)_{\sigma^{-1}} = \emptyset\}$

This in principle permits the s-structure feature ARG_1, and thereby the agent of the event of writing, to be associated with either the grammatical function SUBJ, or OBL_θ (e.g. in the passive), or indeed with no grammatical function (if, for example, the agent were unrealized syntactically). But the possibilities of complex predicates go beyond what is generally admitted for argument structure alternations in this (or any) argument structure model, at least with respect to simplex predicates. In the case of the complex predicate in (2), for example, the agent of the event of writing must be associated with the grammatical function OBJ_θ, which is not possible in the Findlay/Asudeh *et al.* model.[21]

Whichever approach one takes to the representation of the meaning of predicates (e.g. whether along the lines of (22) or (23)), the

[21] The problem is that, as mentioned already, in traditional argument structure terms the agent of 'write' is '[−O]', but when embedded under the light verb it must be realized as '[+O]'.

solution to the problem at hand is in fact readily available in the glue system, and relatively simple to implement. The present exposition adopts the model of Asudeh and Giorgolo (2012).[22] Findlay's (2014) augmentations of Asudeh and Giorgolo's model are not crucial to the point at hand, so they are not utilized in this section, in order to simplify the discussion.

There is no need to change any of the basic assumptions regarding ordinary verbs like 'write'. That is, the lexical entry for 'write' will include the information in (23), just as under the proposals of Asudeh and Giorgolo (2012). As stated, the information in this lexical entry means that the f-structure SUBJ of a clause headed by 'write' will be associated with the ARG_1 feature at s-structure, and thereby with the agent of the writing event in the meaning representation. Now let us assume that the Urdu light verb *de* 'let' has a lexical entry such as the following:[23]

(25) 'let' V

$$(\uparrow \text{PERMISSIVE}) = +$$
$$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$$
$$(\uparrow \text{OBJ}_\theta)_\sigma = (\uparrow_\sigma \text{ARG}_3)$$

$$\lambda P.\lambda y.\lambda x.\lambda e.let(x, y, P(y, e)) :$$
$$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$$
$$(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$$

$$\lambda P.\lambda y.\lambda x.\lambda e.P(x, y, e) :$$
$$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$$
$$(\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$$

Consider first only the f-descriptions in the third and fourth lines and the first meaning constructor. According to the f-descriptions, the light verb requires that the f-structure for its clause contain both a SUBJ and an OBJ_θ argument, associated with the s-structure features ARG_1 and ARG_3 respectively. This essentially corresponds to the subcategorization for [−O] AGENT and [+R] GOAL argu-

[22] See (30) for the demonstration that the proposal would also work under a more standard treatment of glue expressions (i.e. using $(\uparrow \text{SUBJ})_\sigma$ etc. rather than $(\uparrow_\sigma \text{ARG}_1)$).

[23] I follow Butt (1998) in assuming that this verb does not introduce a new event variable, but the point is not crucial.

ments in (4) (merely with the argument structure mapping process resolved for the example under discussion). When combined with an ordinary transitive (or indeed intransitive) verb the specification $(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$ merely replicates that of the lexical verb, but the specification $(\uparrow \textsc{obj}_\theta)_\sigma = (\uparrow_\sigma \textsc{arg}_3)$ specification is new. The first meaning constructor also introduces a new entity variable into the meaning representation, referring to the permitter. By default, one would expect that if a word introduces a new grammatical function, and also introduces a new variable in the semantics corresponding to a grammatical function, then the meaning constructor introducing that variable will link it with the grammatical function via the semantic structure referred to in the corresponding glue term. This is what the first meaning constructor does: it associates the $\textsc{obj}_\theta$, via $\textsc{arg}_3$, with the 'permitter' role, and leaves the SUBJ function associated, via $\textsc{arg}_1$ with an argument of the embedded predicate. That is, if we combine the meaning of 'write' from (23) with the first meaning constructor in (25), we get:

(26) $\lambda z.\lambda y.\lambda x.\lambda e.let(x, y, [write(e) \wedge agent(e, y) \wedge theme(e, z)]) :$
$(\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{arg}_3) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma$

This may be what by default we should expect, but as it is, it does not work: the subject of the complex predicate should be the permitter, not the agent of the event permitted, and the $\textsc{obj}_\theta$ should be the agent of the event permitted, not the permitter. That is, the glue term $(\uparrow_\sigma \textsc{arg}_1)$ in (26) is linked with y, the agent of the event of writing, while the term $(\uparrow_\sigma \textsc{arg}_3)$ in (26) is associated with x, the 'permitter'. But since $\textsc{arg}_1$ is linked with SUBJ, and $\textsc{arg}_3$ with $\textsc{obj}_\theta$, this means that, given the sentence in (2), with f-structure as in (19) = (27), Saddaf would be the permitter and Anjum the writer.

(27) $$\begin{bmatrix} \textsc{pred} & \text{`write'} \\ \textsc{permissive} & + \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{`Anjum'} \end{bmatrix} \\ \textsc{obj}_{\text{goal}} & \begin{bmatrix} \textsc{pred} & \text{`Saddaf'} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{`note'} \end{bmatrix} \end{bmatrix}$$

This is where the second meaning constructor comes in. This makes no contribution to the meaning: it has an identity function on the meaning side. On the glue side, however, it takes an ordered set of glue premises and returns the same set in a different order. This reordering functions to effectively swap the associations between the glue terms $(\uparrow_\sigma \text{ ARG}_1)$ and $(\uparrow_\sigma \text{ ARG}_3)$ and the entity variables in the meaning representation, such that $(\uparrow_\sigma \text{ ARG}_1)$ is now linked with x, and $(\uparrow_\sigma \text{ ARG}_3)$ with y. ARG_1 is still linked with SUBJ, and ARG_3 with OBJ_θ, since these specifications cannot be changed, once made (in some sense, therefore, preserving the principle of Direct Syntactic Encoding). But we now have the correct associations between grammatical functions and thematic roles: SUBJ is linked to the permitter, and OBJ_θ to the writer. That is, if we compose the meaning constructor in (26) with the second meaning constructor in (25), the result is as shown in (28).

(28) $\lambda z.\lambda y.\lambda x.\lambda e.let(x, y, [write(e) \wedge agent(e, y) \wedge theme(e, z)]) :$
$(\uparrow_\sigma \text{ ARG}_2) \multimap (\uparrow_\sigma \text{ ARG}_3) \multimap (\uparrow_\sigma \text{ ARG}_1) \multimap (\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma$

The meaning constructor in (28) differs from that in (26) only in that the glue terms $(\uparrow_\sigma \text{ ARG}_1)$ and $(\uparrow_\sigma \text{ ARG}_3)$ are reordered. Crucially, this means that $(\uparrow_\sigma \text{ ARG}_1)$ is now associated with x on the meaning side, and $(\uparrow_\sigma \text{ ARG}_3)$ with y, rather than the other way around. x represents the 'permitter'; by the f-descriptions in the lexical entries for both 'write' and 'let', $(\uparrow_\sigma \text{ ARG}_1)$ is projected from $(\uparrow \text{ SUBJ})$; therefore, the SUBJ is now associated with the 'permitter', as it should be. Likewise, $(\uparrow_\sigma \text{ ARG}_3)$ is projected from $(\uparrow \text{ OBJ}_\theta)$, so this is associated with y, the agent of 'write'.

In this exposition I have treated the light verb 'let' as introducing two separate meaning constructors, but I do this purely for expository purposes: it is of course simpler to treat them as a single meaning constructor, which serves both to introduce the relevant meaning for the light verb, and to reorder the glue terms in such a way as to produce the correct associations between grammatical functions and semantic roles. That is, the lexical entry for 'let' given in (25) can be simplified by composing the two glue terms into one:

(29) 'let' V

$(\uparrow \text{PERMISSIVE}) = +$
$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBJ}_\theta)_\sigma = (\uparrow_\sigma \text{ARG}_3)$

$\lambda P.\lambda y.\lambda x.\lambda e.let(x, y, P(y, e))$:
$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

According to the present proposal, the meaning constructor introduced by the light verb in the lexicon serves to control and constrain what is traditionally understood as 'argument fusion', in a rather more formally explicit way than is found in any other LFG literature. To summarize, the 'argument structure' associations between grammatical functions and s-structure ARG_x features, as specified in the lexical entries of lexical verbs by f-descriptions such as $(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$, are not altered in any way by the light verb, because once they have been specified it is impossible to change them. But what the light verb can do is introduce new arguments, and new 'argument structure' associations between grammatical functions and s-structures features, and, crucially, it can reassociate the grammatical function – s-structure feature pairs with different semantic arguments in the meaning representation, which suffices to account for the usually rather mysterious process of 'argument fusion'.

The present proposal differs very clearly from the standard linking/XLE accounts, not only in its integration of a semantic representation, but also in its assumptions regarding the contribution of the light verb. Under the present proposal, the light verb does not introduce a new SUBJ argument, and does not cause the SUBJ of the embedded predicate to be demoted to OBJ_θ, as in linking/XLE approaches.[24]

[24] According to the presentation in this section, the light verb does specify the existence of a SUBJ argument via the equation $(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$ in the lexical entry, but as mentioned this is not a new contribution since it is already specified by the lexical verb. At least for the examples discussed in this paper, the fact that it is already specified by the lexical verb means that it is superfluous in the lexical entry for the light verb. It could perhaps, therefore, be removed, but I leave it in since there may potentially be contexts in which its presence is necessary.

Rather, it introduces a new OBJ$_\theta$ argument (= ARG$_3$), and then associates that OBJ$_\theta$ with the embedded agent.[25] At the same time it co-opts the SUBJ argument introduced by the lexical verb, and associates it with the new semantic role that its meaning introduces (i.e. in the example under discussion, the 'permitter').[26]

An empirically important difference between the present proposal and approaches that make use of the restriction operator is that under the present proposal the 'subject' of the embedded predicate, i.e. the permittee of permissive 'let' or the causee of a causative predicate, is not in fact a subject at any level of representation. This aligns with the Romance evidence discussed by Alsina (1996, 213–217) and Andrews (2007), where it is very clear that causees of causative complex predicates are not subjects, since only subjects can launch floating quantifiers, while causees are unable to do this. In a restriction-based approach the 'subject' of the embedded verb is still a subject at f-structure, merely not in the 'full' f-structure for the clause, so this constraint does not fall out so naturally.

[25] Or, more precisely, it associates it with the semantic role that is associated with SUBJ in the meaning constructor of the lexical verb, since this need not be an agent, of course.

[26] A more subtle difference between the present proposal and the linking approach, at least, is that the present proposal depends on the combinatory possibilities being stipulated in the lexical entries of the light verbs. For example, the additional argument introduced by 'let' in (29) is necessarily an OBJ$_\theta$, such that this is the only possible grammatical function for the subject of the embedded predicate. As pointed out by an anonymous reviewer, in some respects the lack of formalization, and the resulting lack of constraints on argument fusion, in the linking approach could be considered advantageous; for example, Alsina and Joshi (1991) utilise the potential for variability in linking to account for differential case marking phenomena. In principle, of course, a fully formalized account with the same empirical coverage is to be preferred, and it does not seem in principle problematic to introduce optionality into the lexical entries of light verbs where necessary to simulate the variability that the linking approach affords. Further investigation is required to determine precisely what degree of freedom in linking is desired, and how well this could be formalized in the present approach. In this regard, a reviewer suggests that it may prove beneficial to introduce a more complex event structure representation into the semantics, e.g. as proposed by Butt and Ramchand (2005).

The proposal made here works just as well under a more traditional approach to verbal meaning constructors, i.e. that exemplified in (22). Under such an approach, the lexical entry for *de* ‘let’ would be:

(30) ‘let’ V

$(\uparrow \text{PERMISSIVE}) = +$

$\lambda P.\lambda y.\lambda x.\lambda e.let(x, y, P(y, e)) :$
$[(\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow \text{OBJ}_\theta)_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

The meaning constructor in (30) introduces a new entity variable on the meaning side, representing the permitter, and by the order of the glue terms on the glue side this variable is associated with the semantic structure $(\uparrow \text{SUBJ})_\sigma$. The variable that was associated with $(\uparrow \text{SUBJ})_\sigma$ by the embedded predication becomes associated with $(\uparrow \text{OBJ}_\theta)_\sigma$. Note also that it would be trivial to rework this proposal within the ‘First Order’ glue of Kokkonidis (2008), or the propositional glue of Andrews (2010).

4 EXTENDING THE ANALYSIS

In this section, I show that the present proposal works unproblematically for the most complicated complex predicates treated in the linking/XLE literature, and in addition that it is able even to go beyond these approaches in dealing easily with phenomena that they cannot capture. I also discuss one formal drawback of the present proposal, which however does not affect the account of argument fusion and does not make the analysis any less adequate than the standard LFG analyses of other much less problematic phenomena.

To begin with, the present proposal has no difficulty in dealing with recursively embedded complex predicates, as found e.g. in Urdu. Butt *et al.* (2010) provide the following example of a nominal predicate quadruply embedded in a complex predicate, with the ‘linking’ style argument structure shown in (32); this is the most complex complex predicate I am aware of having been treated in the literature.[27]

[27] Following Butt (2014), I make a minor change to the ±O/R features in (32) compared with those assumed by Butt *et al.* (2010). The change is not crucial.

(31) *taaraa-ne* *amu-ko* *(bacce-se)*
Tara-ERG Amu-DAT child.OBL-INSTR
haathii *pinc* *kar-vaa* *le-ne*
elephant.M.SG.NOM pinch do-CAUS take-INF.OBL
dii-yaa
give-PERF.M.SG
'Tara let Amu have the elephant pinched (by the child) (completely).'

(32) 'let' ⟨ AG GO 'take' ⟨ AG CAUS ⟨ AG PAT 'do' ⟨ AG 'pinch' ⟨ AG TH ⟩⟩⟩⟩⟩

AG	GO		AG	TH
[−O]	[+O]		([−O])	[−R]
SUBJ	OBJ_θ		OBL_θ	OBJ

The core element of this verb form is a Noun-Verb complex predicate consisting of the predicate noun *pinc* 'pinch', and the light verb *kar* 'do'. This is embedded under a causative predicate, which is realized morphologically on the light verb *kar* (but which has scope over the whole Noun-Verb predicate). This is further embedded under the 'completive' aspectual light verb *le* (the lexical meaning of which is 'take'). Finally, this four-part predicate is embedded under the permissive light verb *de* 'let', which we saw in (2). I assume the following lexical entries for the verb forms and morphemes involved, with the permissive unchanged from (29).[28]

(33) 'pinch' N

$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBJ})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

$\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x) \wedge patient(e,y)$:
$(\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

[28] It is not particularly important for the present purposes precisely how one divides the meaning of the Noun-Verb complex predicate 'do a pinch' between the N and the V, i.e. between (33) and (34). The analysis assumed here associates the whole meaning of the Noun-Verb complex with the noun, which corresponds most closely with what Butt *et al.* (2010) assume in their linking-based presentation. The XLE analysis would be somewhat different, however, with 'pinch' introducing only an object(/patient) argument, and the subject(/agent) argument being introduced only by the light verb *kar* 'do'.

(34) 'do' V

$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$

$\lambda P.\lambda x.\lambda e.P(x,e):$
$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

(35) CAUSE

$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBL}_\theta)_\sigma = (\uparrow_\sigma \text{ARG}_4)$
$(\uparrow \text{CAUSE}) = +$

$\lambda P.\lambda y.\lambda x.\lambda e.cause(x,y,P(y,e)):$
$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

(36) 'take' V

$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{COMPLETIVE}) = +$

$\lambda P.\lambda x.\lambda e.completely(P(x,e)):$
$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

(37) 'let' V

$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBJ})_\sigma = (\uparrow_\sigma \text{ARG}_3)$
$(\uparrow \text{PERMISSIVE}) = +$

$\lambda P.\lambda y.\lambda x.\lambda e.let(x,y,P(y,e)):$
$[(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

Essentially, the causative predicate associates ARG_4 with the agent of the pinching, and reassociates ARG_1 with the causer. The permissive reassociates the causer with ARG_3, and ARG_1 with the permitter. Composing all the relevant meanings together will produce the meaning constructor in (38); the glue proof for this derivation is shown in Figure 1 on p. 454.

(38) $\lambda z.\lambda y.\lambda x.\lambda w.\lambda e.let(w,x,completely(cause(x,y,(pinch(e) \wedge agent(e,y) \wedge patient(e,z))))) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma$

The s-structure feature ARG_1 is linked to SUBJ, meaning that the SUBJ is understood as the permitter; ARG_2 is linked to OBJ, meaning that OBJ is understood as the patient of the pinching event; ARG_3 is linked to OBJ_θ, meaning that OBJ_θ is understood as the causer of the pinching event; ARG_4 is linked to OBL_θ, meaning that OBL_θ is understood as the agent of the pinching event. So, the f-structure for (31) will be as in (39) which, in association with the meaning constructor in (38), will result in the correct interpretation.

(39)
$$\begin{bmatrix} \text{PRED} & \text{`pinch'} \\ \text{CAUSE} & + \\ \text{PERMISSIVE} & + \\ \text{COMPLETIVE} & + \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{`Tara'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{`elephant'} \end{bmatrix} \\ \text{OBJ}_\theta & \begin{bmatrix} \text{PRED} & \text{`Amu'} \end{bmatrix} \\ \text{OBL}_\theta & \begin{bmatrix} \text{PRED} & \text{`child'} \end{bmatrix} \end{bmatrix}$$

The present proposal is thus able to deal with even very complex complex predicates; there is no reason why it should not be able to deal with essentially the same range of phenomena that can be dealt with under the linking and XLE approaches. There is, however, one respect in which the present proposal may be at a disadvantage relative to the traditional approaches, and which is relevant to the analysis of (31). The monoclausality assumed for the f-structure in §3 has one unfortunate consequence: there are no necessary constraints on the order of composition of predicates. That is, while the desired meaning (in (38)) can be correctly derived from the premises (as shown in Figure 1), it is also possible to derive a number of incorrect interpretations, by applying the light verbs' meaning constructors in different orders. Essentially, this is a problem of scope. The advantage of the multiclausal analysis obtained by using the restriction operator is that the order of embedding of the predicates can be constrained. The XLE analysis of Urdu complex predicates implies that the order of embedding must reflect the tree structure (though at least with unembedded complex predicates, there is no necessary subordination of the matrix verb under the light verb, or vice versa, in c-structure terms). The same is true of Romance complex predicates, as discussed by Alsina (1997) and e.g. Andrews (2007). That is, for example, each recursively embedded complex predicate will form a subconstituent of the larger verbal constituent. Under the linking/XLE approaches, this will result in an f-structure semantic form that shows the correct embedding. Under the present proposal there is no embedding in semantic forms, and there is no immediately available means of enforcing the correct order of embedding in the semantics. However, there are two main reasons why this apparent disadvantage of the present approach is not fatal.

To begin with, although the linking/XLE approaches are capable of obtaining the correct order of embedding in the f-structure semantic form, it is not at all obvious that they could easily achieve the same in a glue-based semantic representation, if they were augmented with such. That is, the problem with the present proposal is no more a problem than it is for the traditional accounts, if only the semantics is considered (and part of the present proposal is that only the semantics is relevant, since there is no predicate composition in f-structure). Any proposal that assumes a monoclausal f-structure (such as Butt 1995 and Alsina 1996) would be unable to account for the order of composition in glue. A restriction-based account seems less problematic, because there are distinct f-structures for each level of embedding, but restriction leaves these distinct f-structures essentially dissociated. This means that there would be no easy way for the meaning constructor of the light verb to refer to the (s-structure projected from the) f-structure associated with the predicate embedded under it.[29] The only way to constrain the glue composition effectively by reference to f-structure is to assume an embedded f-structural representation, as proposed by Andrews and Manning (1999) and as exemplified in (14). However, no standard LFG analysis assumes this, and as discussed above it rather undermines the basic intuition of monoclausality.

Secondly, the difficulty with constraining semantic scope when the f-structure is flat is not unique to complex predicates. As discussed

[29] So it is not clear that a restriction-based account is even compatible with a glue-based semantic analysis. The problem may possibly be resolvable if the f-descriptions in the c-structure specified that the s-structures projected from the dissociated f-structures be embedded one inside the other, but the details of this remain to be explored.

e.g. by Andrews and Manning (1999), it is a long-term problem in the analysis of recursive modification. Recursive modification involving one or more intensional adjectives must be interpreted semantically with respect to the linear / hierarchical order, e.g.:

(40) a. The former trustworthy chairman.

b. The trustworthy former chairman.

In LFG, the ADJUNCT set in which such modifiers appear at f-structure is flat, such that there is no way for the interpretative constraint to be enforced in the semantics. This is already a problem for LFG, then, and whatever solution may be proposed can be easily extended to the analysis of complex predicates, such that this should not be considered a fatal flaw of the present proposal.[30]

This difficulty aside, there is one important respect in which the present proposal is descriptively superior to the linking and XLE approaches. Butt *et al.* (2010) note that the OBL$_{\theta}$ in sentences like (31) is optional, and should perhaps be treated as an adjunct, but that this is not done in their linking analysis because "argument suppression with respect to argument merger as part of complex predication is not predicted within Linking Theory." That is, the linking approach to complex predicates has no way to deal with the optionality of arguments. This is also impossible within XLE, since there is no way to remove an argument from the subcategorization list of a predicate.

[30] Besides the proposal of Andrews and Manning (1999), another proposed solution is under development by Andrews (2015). Both of these involve rather severe changes to the traditional LFG view of f-structure. Note that neither f-precedence, nor the notion of linear precedence discussed by Asudeh (2009), can handle the complex predicate data, since the crucial relation is c-structure hierarchy, and not necessarily linearity. In XLE, surface scope and surface adjunct scope can be captured at f-structure using the notations <s / >s and ∈<h<s / ∈<h>s respectively, but it is not immediately obvious how this information could be utilized formally to constrain a glue derivation. Perhaps the simplest alternative is simply to state a constraint to the effect that semantic composition should mirror the c-structure, equivalent to the constraint placed on predicate composition by Alsina (1997, 237–238), though it must be admitted that such a solution is rather informal, and not easily formalized.

Under the present proposal, optionality of arguments would be unproblematic. Butt *et al.* (2010) suggest a possible adjunct analysis for the optional element, presumably because this is the default interpretation for an optional phrase. However, work by Needham and Toivonen (2011), Christie (2013), and Toivonen (2013) show that the argument-adjunct distinction is not absolute, and that optionality may also be a feature of some arguments. At least for the present purposes, given that the standard approaches to complex predicates in LFG assume that the element in question is an argument, an analysis as an optional argument seems preferable to an analysis as an adjunct. Asudeh and Giorgolo (2012) and Asudeh *et al.* (2014) formalize a semantics-based account of optional arguments of simplex predicates, and this can easily be transferred to the present analysis. Specifically, the optional argument is the causee of the (morphological) CAUSE predicate. Cf. the following example, based on the relevant portion of the complex predicate in (31).

(41) *amu-ne* *(bacce-se)* *haathii* *pinc*
Amu-ERG child.OBL-INSTR elephant.M.SG.NOM pinch
kar-vaa-yaa
do-CAUS-PERF.M.SG
'Amu had the elephant pinched (by the child).'

Instead of the lexical contribution in (35) for the causative element, we can assume the contribution in (42). I slightly update Asudeh and Giorgolo's representations based on Findlay (2014) and Asudeh *et al.* (2014), but treat the variability in grammatical function assignment as already resolved, since it is not relevant to the point at hand and would only complicate the discussion.[31]

[31] That is, in the first line of (42) I assume $\{(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_1) \mid (\uparrow_\sigma \text{ARG}_1)_{\sigma^{-1}} = \emptyset\}$ rather than Asudeh *et al.*'s $\{(\uparrow \{\text{SUBJ} \mid \text{OBL}_\theta\})_\sigma = (\uparrow_\sigma \text{ARG}_1) \mid (\uparrow_\sigma \text{ARG}_1)_{\sigma^{-1}} = \emptyset\}$, and make the equivalent simplification in the second line. Since processes such as passivization, etc., are not at issue here, the option of either SUBJ or OBL_θ in the first line will necessarily resolve to SUBJ, and the same option in the second line will necessarily resolve to OBL_θ, in accordance with Kibort's (2007) Mapping Principle, so it is simpler here to ignore the optionality.

(42) CAUSE $\{(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_1) \mid (\uparrow_\sigma \textsc{arg}_1)_{\sigma^{-1}} = \emptyset\}$
$\{(\uparrow \textsc{obl}_\theta)_\sigma = (\uparrow_\sigma \textsc{arg}_4) \mid (\uparrow_\sigma \textsc{arg}_4)_{\sigma^{-1}} = \emptyset\}$

$$\lambda P.\lambda y.\lambda x.\lambda e.cause(x, y, P(y,e)) :$$
$$[(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap$$
$$(\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma$$

$$(\lambda P.\exists x.P(x) : ((\uparrow_\sigma \textsc{arg}_4) \multimap \uparrow_\sigma) \multimap \uparrow_\sigma)$$

The f-descriptions in the first two lines of the lexical entry introduce the two arguments of the CAUSE predicate. The first line states that either there will be an f-structure SUBJ which projects to the semantic structure $\textsc{arg}_1$, or else there is nothing in the f-structure which projects to $\textsc{arg}_1$. Likewise, the second line states that either there will be an f-structure $\textsc{obl}_\theta$ which projects to the semantic structure $\textsc{arg}_4$, or else there is nothing in the f-structure which projects to $\textsc{arg}_4$. In the present context there is nothing to license the absence of a SUBJ from the f-structure. However, the rest of the lexical entry does license the absence of the $\textsc{obl}_\theta$ argument from the f-structure. The first meaning constructor is unchanged from (35): it introduces a new entity variable, the 'causer', and rearranges the associations between s-structure $\textsc{arg}_x$ features and variables, such that $\textsc{arg}_1$ will be associated with the causer and $\textsc{arg}_4$ with the causee.

The crucial element is the second meaning constructor in (42). This optional meaning constructor existentially quantifies the variable associated with $\textsc{arg}_4$. If, then, the $\textsc{obl}_\theta$ argument is absent from the f-structure, i.e. if no causee is explicitly realized in the syntax, this meaning constructor can apply to quantify the variable that would otherwise be left hanging. If the causee is explicitly realized in the syntax, appearing as $\textsc{obl}_\theta$ at f-structure, then this will serve to quantify the variable associated with $\textsc{arg}_4$, and the optional meaning constructor in (42) will not be required. That is, there are two possible f-structures for the example in (41), depending on whether or not the causee is omitted:

(43)
$$\begin{bmatrix} \textsc{pred} & \text{'pinch'} \\ \textsc{cause} & + \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{'Amu'} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{'elephant'} \end{bmatrix} \\ \textsc{obl}_\theta & \begin{bmatrix} \textsc{pred} & \text{'child'} \end{bmatrix} \end{bmatrix}$$

(44) $\begin{bmatrix} \textsc{pred} & \text{'pinch'} \\ \textsc{cause} & + \\ \textsc{subj} & \begin{bmatrix} \textsc{pred} & \text{'Amu'} \end{bmatrix} \\ \textsc{obj} & \begin{bmatrix} \textsc{pred} & \text{'elephant'} \end{bmatrix} \end{bmatrix}$

Assuming the simplified noun meanings in (45), and assuming the simplified 'finiteness' meaning constructor in (46) to quantify the event variable, the resulting meaning constructors for (43) and (44) will be as in (47) and (48) respectively. The glue proofs for these derivations appear in Figures 2 and 3 respectively, on pp. 455 and 456.

(45) a. $Amu : \uparrow_\sigma$

b. $elephant : \uparrow_\sigma$

c. $child : \uparrow_\sigma$

(46) $\lambda P.\exists e.P(e) : ((\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma) \multimap \uparrow_\sigma$

(47) $\exists e.cause(Amu, child, (pinch(e) \wedge agent(e, child) \wedge$ $patient(e, elephant))) : \uparrow_\sigma$

(48) $\exists e.\exists y.cause(Amu, y, (pinch(e) \wedge agent(e, y) \wedge$ $patient(e, elephant))) : \uparrow_\sigma$

In this way, the present proposal for dealing with complex predicates can easily handle the optionality of arguments, in a way that neither the linking approach nor the XLE approach can.

5 COMPARISON WITH PREVIOUS GLUE APPROACHES

There exist a few previous treatments of complex predicate formation within LFG that make significant reference to semantics, though there are none within the current standard 'new' glue approach, and none that have been widely adopted. In this section, I briefly discuss each approach, and provide comparisons with my own proposals.

The earliest proposal was made by Kaplan and Wedekind (1993). They do not explicitly make use of glue, but Dalrymple *et al.* (1993a) briefly illustrate how their proposal would be represented in glue. As mentioned in §2, Kaplan and Wedekind (1993) introduced the restriction operator into LFG, and into the analysis of complex predicates.

They assume that the lexical entry of a verb like Urdu *likh* 'write' contains the following default specifications:

(49) a. $(\sigma \uparrow \text{ARG1}) = \sigma(\uparrow \text{SUBJ})$
b. $(\sigma \uparrow \text{ARG2}) = \sigma(\uparrow \text{OBJ})$

Difference of notation aside, this is identical to the third and fourth lines of (23). Kaplan and Wedekind (1993) further assume that a lexical redundancy rule exists that can systematically modify these specifications for any ordinary verb, such that they become:

(50) a. $(\sigma\ [\uparrow \backslash \text{SUBJ}]\ \text{ARG1}) = \sigma(\uparrow \text{OBJ2})$
b. $(\sigma\ [\uparrow \backslash \text{SUBJ}]\ \text{ARG2}) = \sigma(\uparrow \text{OBJ})$

As described by Dalrymple *et al.* (1993a, 16), this means that the meaning for *likh* 'write' will be as in (51) (using the original glue notation). This will combine with the meaning for the permissive light verb, *de* 'let', which is shown in (52). The 'new glue' (Dalrymple *et al.* 1999) versions of (51) and (52) are shown in (53) and (54) respectively.

(51) $(\uparrow \backslash \text{SUBJ})_\sigma \rightsquigarrow \mathit{write}(X,Y)$
where X is the meaning of the OBJ2, and Y is the meaning of the OBJ.

(52) $\uparrow_\sigma \rightsquigarrow \mathit{permit}(X,Y)$
where X is the meaning of the SUBJ, and Y is the meaning of $\uparrow \backslash$SUBJ.

(53) $\lambda y.\lambda x.\mathit{write}(x,y) : (\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{OBJ}_\theta)_\sigma \multimap (\uparrow \backslash \text{SUBJ})_\sigma$

(54) $\lambda P.\lambda x.\mathit{permit}(x,P) : (\uparrow \backslash \text{SUBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap \uparrow_\sigma$

As discussed in §2, it may be preferable to avoid the use of the restriction operator in any case, but this is particularly true when one starts using it to refer to semantic structures projected from f-structures. But the most serious problem with the proposal of Kaplan and Wedekind (1993), which was noted by e.g. Dalrymple *et al.* (1993a), Butt (1994), Andrews and Manning (1999), and Butt *et al.* (2003), is that it assumes a fundamentally lexical approach to complex predicate formation and argument fusion. Specifically, the operation that serves to reassign the ARG_1 of 'write' to OBJ2 ($=\text{OBJ}_\theta$)

applies in the lexicon. As pointed out by Dalrymple *et al.* (1993a, 16), Kaplan and Wedekind's proposal predicts that any ordinary lexical verb can combine with only a finite number of light verbs, and entails a considerable amount of lexical duplication: there must exist separate lexical entries for a verb that combines with one light verb, with two light verbs, etc., and for light verbs that appear as the only light verb in a sentence, or with one other light verb in the sentence, etc. To the extent that Kaplan and Wedekind's semantic proposals can be converted to apply within the framework of Butt *et al.* (2003), who show that the restriction operator can be used to permit predicate composition in the syntax, they would unavoidably be affected by the problems with the linking and (especially) XLE approaches described in §2.

An alternative proposal is made by Dalrymple *et al.* (1993a), followed by Zaenen and Dalrymple (1995, 1996). Their proposals are formalized in the original glue representation.[32] Their proposal is that the links, or mapping, between the syntactic arguments and semantic roles of verbs are not defined in the lexical entries of those verbs, but are derived from independent 'mapping rules', which are universally available in the analysis of any clause. For example, they propose the following mapping rule (p. 8), which can apply to any clause containing a simple transitive verb selecting for an agent and a theme argument:

(55) $!(\forall f, X, Y.((f\ \textsc{subj})_\sigma \rightsquigarrow X) \otimes ((f\ \textsc{obj})_\sigma \rightsquigarrow Y) \multimap$
$\mathit{agent}((f\ \textsc{pred})_\sigma, X) \otimes \mathit{theme}((f\ \textsc{pred})_\sigma, Y))$

They explain this rule as follows:

> This rule associates subjects with agents, and objects with themes. It states that for all f-structures f, if the SUBJ of f is X and the OBJ of f is Y, we can conclude that X is the f-structure's PRED's agent, and Y is the f-structure's PRED's theme. (p. 8)

[32] The original glue representation was introduced by Dalrymple *et al.* (1993b); Dalrymple *et al.* (1996) replaced this with a formally simpler system, the first to gain wide currency; the current 'new' glue representation was introduced by Dalrymple *et al.* (1999).

When it comes to complex predicates, there is no alteration or manipulation of the mappings between grammatical functions and semantic roles (since these are not defined in the lexicon). A lexical verb introduces one or more grammatical functions and one or more semantic roles, and a light verb can also introduce a grammatical function and a semantic role. Then the correct mapping rule is selected that can match up all the pairs in the clausal f-structure, both those introduced by the lexical verb and those introduced by the light verb. For example, Dalrymple *et al.* propose the following mapping rule for a sentence with a permissive light verb and a lexical verb with agent and theme:

(56) $!(\forall f, X, Y, Z.((f \text{ SUBJ})_\sigma \rightsquigarrow X) \otimes ((f \text{ OBJ})_\sigma \rightsquigarrow Y) \otimes ((f \text{ OBJ2})_\sigma \rightsquigarrow Z) \multimap \mathit{permitter}((f \text{ PRED})_\sigma, X) \otimes \mathit{agent}((f \text{ PRED})_\sigma, Z) \otimes \mathit{theme}((f \text{ PRED})_\sigma, Y))$

It is not possible to directly convert this proposal into the 'new' glue representation, because the mapping rules proposed mix the meaning language and linear implication in a way that is no longer possible. However, the spirit of the proposal can be implemented. Authors such as Asudeh *et al.* (2008, 2013, 2014), Haug (2008) and Lowe (2015) assume that the meaning of verbs can be broken down into a basic verbal meaning and a semantic role or argument structure template. So, for the Urdu verb *likh* 'write', in place of the lexical entry with a single meaning constructor (23), we can assume a lexical entry such as the following:

(57) 'write' V
$(\uparrow \text{ PRED}) = \text{'write'}$
$\lambda e.\mathit{write}(e) : (\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma$

$(\uparrow \text{ SUBJ})_\sigma = (\uparrow_\sigma \text{ ARG}_1)$
$(\uparrow \text{ OBJ})_\sigma = (\uparrow_\sigma \text{ ARG}_2)$
$\lambda P.\lambda y.\lambda x.\lambda e.P(e) \wedge \mathit{agent}(e, x) \wedge \mathit{theme}(e, y) :$
$[(\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ ARG}_2) \multimap (\uparrow_\sigma \text{ ARG}_1) \multimap (\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma$

The advantage of this is that it raises the possibility of generalizing over both the syntactic and semantic aspects of argument structure patterns using templates (Dalrymple *et al.* 2004; Asudeh *et al.* 2008, 2013). For the present purposes, however, the relevant point is that the second meaning constructor in (57) contains the associations between semantic roles and grammatical functions (via s-structure features), which is the component of sentential meaning that Dalrymple *et al.* (1993a) propose is not a part of lexical entries, but universally available. So, if we were to convert Dalrymple *et al.*'s proposal into a format that conforms with the proposals of Asudeh and Giorgolo (2012) in 'new' glue, we would require the following lexical entry for *likh* 'write':

(58) 'write' V
$(\uparrow \textsc{pred}) = \text{'write'}$
$\lambda e.write(e) : (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma$
$(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
$(\uparrow \textsc{obj})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

This requires that the verb appear in an f-structure with a SUBJ and an OBJ, and also requires that the SUBJ and the OBJ project s-structures $\textsc{arg}_1$ and $\textsc{arg}_2$ respectively. But it makes no statement about how those grammatical functions, or those s-structure features, relate to the semantically entailed participants of the event of writing. This proposal would then require that the meaning constructor in (59) be universally available in the analysis of any sentence, and that in the analysis of a sentence containing the verb 'write' it be used to provide the appropriate semantic relations for the verb based on the $\textsc{arg}_x$ features specified in the verb's lexical entry.[33]

(59) $\lambda P.\lambda y.\lambda x.\lambda e.P(e) \wedge agent(e,x) \wedge theme(e,y) : [(\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma$

The universally available meaning constructor that would be required in the case of a complex predicate such as that in (2) would be as follows (i.e. in place of (56)):

(60) $\lambda P.\lambda Q.\lambda z.\lambda y.\lambda x.P(x,y,[Q(e) \wedge agent(e,y) \wedge theme(e,z)]) : [(\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma$

[33] I.e. (59) replaces (55) in the original formulation.

The meaning constructor in the lexical entry of the permissive light verb *de* 'let' would then be:

(61) $\lambda P.\lambda e.permit(P(e)) : [(\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma] \multimap (\uparrow_\sigma \text{ EV}) \multimap \uparrow_\sigma$

This is slightly different from the original proposal of Dalrymple *et al.* (1993a), since for them verbs do contain specification of their thematic roles in the lexicon, and it is merely the links between those roles and grammatical functions that are specified by the mapping rules. In new glue these cannot be separated without entirely losing the link between semantic role and grammatical function. A full separation would be possible within the proposals of Lowe (2014), where the use of complex typed structures permits meaning constructors to be effectively partitioned in two, but the resulting analysis for complex predicates would be further from Dalrymple *et al.*'s original proposals than the suggestion just made. So, in Lowe's (2014) model, the meaning constructor for a verb like 'write' would be as in (62), while the meaning constructor that would be removed from the lexicon and made universally available would be that in (63).

(62) $\lambda y.\lambda x.\lambda e.write(e) \wedge agent(e,x) \wedge theme(e,y) : (\uparrow_\sigma \text{ REL})_{\langle e\to e\to e\to t\rangle}$

(63) $\lambda P.\lambda y.\lambda x.\lambda e.P(x,y,e) : (\uparrow_\sigma \text{ REL})_{\langle e\to e\to e\to t\rangle} \multimap (\uparrow_\sigma \text{ ARG}_2)_{\langle e\rangle} \multimap (\uparrow_\sigma \text{ ARG}_1)_{\langle e\rangle} \multimap (\uparrow_\sigma \text{ EV})_{\langle e\rangle} \multimap \uparrow_{\sigma\langle e\rangle}$

Whether in its original form, or in one way or another converted to new glue, perhaps the main disadvantage of Dalrymple *et al.*'s (1993a) proposal is that it requires a potentially large inventory of universally available meaning constructors to function as mapping tools, all of which are available in any one derivation.[34] So the first mapping rule discussed above (55) associates subjects with agents and objects with themes, but there must also be different meaning constructors for every potential combination of grammatical functions and thematic roles, including for complex predicates, and in principle all are available for any sentence (though only the correct one will work, of course).

The mapping rules that Dalrymple *et al.* (1993a, 18) propose are of a rather different nature from other meaning constructors: "Map-

[34] Besides the comments here, compare also the comments and criticism on Dalrymple *et al.*'s proposals by Andrews and Manning (1999, 136–141).

ping rules exist separate from the collections of formulas that contain meanings of sentences." That is, it would be necessary to assume something additional in the grammar, alongside the standardly assumed structures and projections, specifically in order to deal with complex predicates. In fact, the ability to generalize mapping possibilities across verbs is readily available by making use of templates to encode generalizations across lexical entries, as shown by Asudeh and Giorgolo (2012) and Asudeh *et al.* (2014).

A further problematic aspect of Dalrymple *et al.*'s proposal is that, at least in the original formulation, these mapping rules necessarily make use of the 'of course' operator !, since each one can be used zero or more times in any derivation. Asudeh and Crouch (2002, 28) and Asudeh (2012, 101) argue that ! can and should be kept out of the linear logic fragment used in glue, in order to protect the resource sensitivity of glue semantics. Whatever the formulation, it remains the case that the mapping rules or meaning constructors concerned must be allowed to apply as many times as necessary in any derivation, weakening the resource sensitivity of the semantic model.

Having said all that, the proposal of Dalrymple *et al.* (1993a) does appear to work: it is a fully formalized semantically integrated account of complex predicate formation that does not rely on manipulable PRED features and predicate composition in the f-structure, and that does not depend on a nebulous concept of argument fusion. These features are precisely what the present proposal aspires to.

Another early proposal for a semantic analysis of complex predicates was made by Andrews and Manning (1999, 119–128). Their proposal depends on a somewhat non-standard syntactic analysis of complex predicates, and the approach in general has not been widely adopted, even by the authors themselves; I will not therefore discuss the proposals of Andrews and Manning (1999, 119–128) here, but focus on the more recent proposal of Andrews (2007). Andrews' (2007) proposal for a semantic analysis of complex predication is in some respects the most similar existing account to the present proposal, but it is formalized in a non-standard approach to glue, and to the LFG projection architecture, developed by Andrews (2010). Like Dalrymple *et al.* (1993a) and the present proposal, Andrews (2007) is concerned with the question of argument fusion, and proposes the following meaning constructor for a causative light verb predicate:

(64) $\lambda P.\lambda y.\lambda x.Cause(x,y,P(y)) : ((\uparrow \text{?OBJ})_e \to \uparrow_p) \to (\uparrow \text{?OBJ})_e \to (\uparrow \text{SUBJ})_e \to \uparrow_p$

where $(\uparrow \text{SUBJ})_e$ and $\uparrow_p$ correspond to $(\uparrow \text{SUBJ})_\sigma$ and $\uparrow_\sigma$ respectively in the more standard approach to the LFG architecture assumed here. $(\uparrow \text{?OBJ})_e$ is essentially a place-holder for a more sophisticated statement governing grammatical function alternations, since in the Romance phenomena that Andrews addresses, the causee may surface as either a dative case OBJ_θ or an accusative case OBJ, depending on whether the embedded predicate is transitive or intransitive, respectively.[35] Andrews (2007) does suggest how a more sophisticated statement might be formulated, but the presentation is brief and the proposal is not explained or exemplified in full. Altogether, the proposal is hard to assess for this reason; it seems to be heading in a similar direction to the present proposal, but the presentation is elliptical and, as stated, it is formalized in a non-standard approach to semantics in LFG.

The most recent proposal regarding complex predicates in LFG is made by Homola and Coler (2013). This proposal is in certain respects reminiscent of that of Dalrymple *et al.* (1993a), but it is formally rather different. Homola and Coler (2013) propose a radically new approach to the syntax-semantics interface in LFG, the details of which are beyond the scope of the present discussion. They deal firstly with the question of predicate composition, proposing a means of permitting predicate composition in the f-structure without having to make use of the restriction operator. Their proposal in this respect is essentially parallel to Dalrymple *et al.*'s (1993a) proposal, but focused on the f-structure rather than semantics. They propose to use *equational unification,* a concept from logical programming, to model predicate composition in f-structure. A set of 'equational theories' E_i constitute a separate subcomponent of the grammar. Homola and Coler propose the semantic forms in (65), and the equational theory in (66), to model the predicate fusion of a causative predicate with an intransitive verb:

(65) a. CAUSE⟨ (↑ SUBJ), f⟨ (↑ OBJ)⟩⟩
b. laugh⟨ (↑ SUBJ)⟩

[35] In the present model, this alternation should fall out unproblematically with the addition of the Findlay-Asudeh *et al.* (2014) argument structure proposals, depending on precisely how the Mapping Principle is formulated.

(66) $E = \{\textsc{cause}\langle\ (\uparrow \textsc{subj}),\ f\langle\ (\uparrow \textsc{obj})\rangle\rangle \approx f\langle\ (\uparrow \textsc{subj})\rangle\}$'

The equational theory in (66) functions to produce the complex semantic form in (67) from those in (65).

(67) $\textsc{cause}\langle\ (\uparrow \textsc{subj}),\ \text{laugh}\langle\ (\uparrow \textsc{obj})\rangle\rangle$

It is evident that this works according to essentially the same principle as the proposal of Dalrymple *et al.* (1993a): a separate component of the grammar contains a set of formulae that specify the argument reassignment/fusion in complex predicates. It therefore suffers from the same problem. A large number of such formulae must be assumed to deal with the full variety of complex predicates and all may be available in any derivation. As for the semantics, Homola and Coler (2013) need a special formula to appear on the c-structure node dominating a lexical verb which, in relation to the causative example they discuss, permits either the SUBJ or the OBJ to function as the actor of the lexical verb (since by their defaults, the subcategorization of the lexical verb would require the SUBJ to fill this role).[36] In this case too, one would presumably need a whole set of different formulae, any of which could potentially apply in any given case. For example, a formula would be required that enabled the $\textsc{obj}_\theta$ to fill the actor role, to cover complex predicates such as the permissive with a transitive predicate. All in all, their proposal involves a thorough revision of the LFG architecture, the implications of which would have to be carefully analysed, yet from the present perspective it still suffers from some of the same problems that already affected earlier proposals made within a more standard model.

While there have been a number of earlier proposals for a semantically integrated account of predicate composition in LFG, and while one or two of these at least show the potential to provide a descriptively adequate account of complex predication (Dalrymple *et al.* 1993a; Andrews 2007), none have been developed in great detail beyond the initial proposal, none have been adopted more widely in the LFG community, and none are formulated (or could easily be reformulated) in the 'new' glue approach, which has been standard in LFG for

[36] It is not worth providing the formulae they propose, since they could only be understood in the wider context of their proposals regarding the syntax-semantics interface in LFG, which as stated is beyond the scope of this paper.

over fifteen years. If in no other respect, then, the present proposal advances on previous proposals simply because it is formulated within the standard approach to LFG + glue, and therefore its potential for wider adoption is correspondingly greater.

6 CONCLUSION

In this paper, I have proposed a new, semantically integrated account of complex predicate formation within LFG + glue. I have shown that the proposed approach to complex predicates can deal with all the data that the standard linking/XLE approaches can deal with, even recursive complex predicate structures. Moreover, the proposed approach improves upon the standard linking/XLE approaches because it is fully formalized (in contrast to the linking approach, at least), does not involve mysterious processes of 'predicate composition' and 'argument fusion', does not require the use of construction-specific mechanisms (such as the restriction operator, manipulable PREDs, etc.), and properly integrates glue semantics. Previous accounts of complex predicates in LFG that integrate semantics either suffer from certain problems, or are not fully developed, but the present proposal is fully formalized within recent approaches to argument structure in LFG + glue, shares none of the problems affecting previous proposals, and involves no construction-specific additions to the formal model.

The one apparent weakness of the proposal, relating to the scope of multiple light verbs in a doubly (or more) embedded complex predicate, is not a weakness on the semantic side but relates to the syntax, and its solution is not specific to the analysis of complex predication, since the problem already affects the analysis of other, considerably more basic, phenomena (like recursive modification). This weakness aside, the present proposal also has the potential to go beyond both the linking and XLE approaches to complex predication in its ability to deal with optionality of arguments. As the only proposal for a semantically integrated account of complex predicates within the current standard approach to LFG + glue and the current standard LFG architecture, its potential for dealing with a wider range of complex predicate phenomena, including phenomena that are problematic for earlier approaches, is a worthy subject for future research.

$$\frac{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x) \\ \wedge patient(e,y) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \\ \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\lambda x.\lambda e.P(x,e) : \\ [(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap \\ (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x) \\ \wedge patient(e,y) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \\ \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x) \\ \wedge patient(e,y) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \\ \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\lambda y.\lambda x.\lambda e.cause(x,y,P(y,e)) : \\ [(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap \\ (\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e) \\ \wedge agent(e,y) \wedge patient(e,z))) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \\ \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e) \\ \wedge agent(e,y) \wedge patient(e,z))) : (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \\ \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\lambda x.\lambda e.completely(P(x,e)) : \\ [(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap \\ (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.completely(cause(x,y, \\ (pinch(e) \wedge agent(e,y) \wedge patient(e,z)))) : \\ (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda P.\lambda y.\lambda x.\lambda e.let(x,y,P(y,e)) : \\ [(\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma] \multimap \\ (\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.completely(cause(x,y, \\ (pinch(e) \wedge agent(e,y) \wedge patient(e,z)))) : \\ (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{c}\lambda z.\lambda y.\lambda x.\lambda w.\lambda e.let(w,x,completely(cause(x,y,(pinch(e) \wedge agent(e,y) \wedge patient(e,z))))) : \\ (\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_4) \multimap (\uparrow_\sigma \text{ARG}_3) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EV}) \multimap \uparrow_\sigma\end{array}}$$

Figure 1: Glue proof for (38)

$$
\dfrac{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x)\\ \wedge patient(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\lambda x.\lambda e.P(x,e) :\\ [(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x)\\ \wedge patient(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}
$$

$$
\dfrac{\begin{array}{l}\lambda y.\lambda x.\lambda e.pinch(e) \wedge agent(e,x)\\ \wedge patient(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\lambda y.\lambda x.\lambda e.cause(x,y,P(y,e)) :\\ [(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e)\\ \wedge agent(e,y) \wedge patient(e,z))) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_4)\\ \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}
$$

$$
\dfrac{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e)\\ \wedge agent(e,y) \wedge patient(e,z))) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_4)\\ \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \qquad elephant : \uparrow_\sigma}{\begin{array}{l}\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e)\\ \wedge agent(e,y) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}
$$

$$
\dfrac{child : \uparrow_\sigma \qquad \begin{array}{l}\lambda y.\lambda x.\lambda e.cause(x,y,(pinch(e)\\ \wedge agent(e,y) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda x.\lambda e.cause(x,child,(pinch(e)\\ \wedge agent(e,child) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}
$$

$$
\dfrac{Amu : \uparrow_\sigma \qquad \begin{array}{l}\lambda x.\lambda e.cause(x,child,(pinch(e)\\ \wedge agent(e,child) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda e.cause(Amu,child,(pinch(e)\\ \wedge agent(e,child) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}
$$

$$
\dfrac{\begin{array}{l}\lambda e.cause(Amu,child,(pinch(e)\\ \wedge agent(e,child) \wedge patient(e,elephant))) :\\ (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \qquad \begin{array}{l}\lambda P.\exists e.P(e) :\\ ((\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\exists e.cause(Amu,child,(pinch(e)\\ \wedge agent(e,child) \wedge patient(e,elephant))) :\\ \uparrow_\sigma\end{array}}
$$

Figure 2: Glue proof for (47)

$$\frac{\begin{array}{l}\lambda y.\lambda x.\lambda e.\mathit{pinch}(e) \wedge \mathit{agent}(e,x)\\ \wedge \mathit{patient}(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \quad \begin{array}{l}\lambda P.\lambda x.\lambda e.P(x,e) :\\ [(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda y.\lambda x.\lambda e.\mathit{pinch}(e) \wedge \mathit{agent}(e,x)\\ \wedge \mathit{patient}(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda y.\lambda x.\lambda e.\mathit{pinch}(e) \wedge \mathit{agent}(e,x)\\ \wedge \mathit{patient}(e,y) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1)\\ \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \quad \begin{array}{l}\lambda P.\lambda y.\lambda x.\lambda e.\mathit{cause}(x,y,P(y,e)) :\\ [(\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma] \multimap\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,z))) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_4)\\ \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda z.\lambda y.\lambda x.\lambda e.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,z))) : (\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_4)\\ \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \quad \mathit{elephant} : \uparrow_\sigma}{\begin{array}{l}\lambda y.\lambda x.\lambda e.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda P.\exists x.P(x) :\\ ((\uparrow_\sigma \textsc{arg}_4) \multimap \uparrow_\sigma) \multimap \uparrow_\sigma\end{array} \quad \begin{array}{l}\lambda y.\lambda x.\lambda e.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{arg}_4) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda x.\lambda e.\exists y.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\mathit{Amu} : \uparrow_\sigma \quad \begin{array}{l}\lambda x.\lambda e.\exists y.\mathit{cause}(x,y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\lambda e.\exists y.\mathit{cause}(\mathit{Amu},y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array}}$$

$$\frac{\begin{array}{l}\lambda e.\exists y.\mathit{cause}(\mathit{Amu},y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ (\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma\end{array} \quad \begin{array}{l}\lambda P.\exists e.P(e) :\\ ((\uparrow_\sigma \textsc{ev}) \multimap \uparrow_\sigma) \multimap \uparrow_\sigma\end{array}}{\begin{array}{l}\exists e.\exists y.\mathit{cause}(\mathit{Amu},y,(\mathit{pinch}(e)\\ \wedge \mathit{agent}(e,y) \wedge \mathit{patient}(e,\mathit{elephant}))) :\\ \uparrow_\sigma\end{array}}$$

Figure 3: Glue proof for (48)

REFERENCES

Farrell ACKERMAN and Gert WEBELHUTH (1996), The Construct PREDICATE: Empirical Arguments and Theoretical Status, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG96 Conference*, CSLI Publications, Stanford, CA, http://cslipublications.stanford.edu/LFG/1/ackerman.html.

Farrell ACKERMAN and Gert WEBELHUTH (1998), *A Theory of Predicates*, CSLI Publications, Stanford, CA.

Tafseer AHMED and Miriam BUTT (2011), Discovering Semantic Classes for Urdu N-V Complex Predicates, in *Proceedings of the Ninth International Conference on Computational Semantics (IWCS 2011)*, pp. 305–309, Association for Computational Linguistics, Stroudsburg, PA.

Tafseer AHMED, Miriam BUTT, Annette HAUTLI, and Sebastian SULGER (2012), A Reference Dependency Bank for Analyzing Complex Predicates, in Nicoletta CALZOLARI, Khalid CHOUKRI, Thierry DECLERCK, Mehmet UĞUR DOĞAN, Bente MAEGAARD, Joseph MARIANI, Jan ODIJK, and Stelios PIPERIDIS, editors, *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, European Language Resources Association (ELRA).

Alex ALSINA (1996), *The Role of Argument Structure in Grammar*, CSLI Publications, Stanford, CA.

Alex ALSINA (1997), A theory of Complex Predicates: Evidence from Causatives in Bantu and Romance, in Alex ALSINA, Joan BRESNAN, and Peter SELLS, editors, *Complex Predicates*, pp. 203–246, CSLI Publications, Stanford, CA.

Alex ALSINA and Smita JOSHI (1991), Parameters in Causative Constructions, in L. DOBRIN, L. NICHOLS, and R. M. RODRIGUEZ, editors, *Proceedings from the 27^th^ Regional Meeting of the Chicago Linguistic Society (CLS)*, pp. 1–16, Chicago Linguistic Society, Chicago, IL.

Avery D. ANDREWS (2001), Iofu and Spreading Architecture in LFG, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG01 Conference*, CSLI Publications, Stanford, CA.

Avery D. ANDREWS (2007), Projections and Glue for Clause-Union Complex Predicates, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG07 Conference*, pp. 44–65, CSLI Publications, Stanford, CA.

Avery D. ANDREWS (2008), The Role of PRED in LFG + Glue, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG08 Conference*, pp. 47–67, CSLI Publications, Stanford, CA.

Avery D. ANDREWS (2010), Propositional Glue and the Projection Architecture of LFG, *Linguistics and Philosophy*, 33(3):141–170.

Avery D. ANDREWS (2015), Sets and Heads in LFG, MS, Australian National University.

Avery D. ANDREWS and Christopher D. MANNING (1999), *Complex Predicates and Information Spreading in LFG*, CSLI Publications, Stanford, CA.

Ash ASUDEH (2004), *Resumption as Resource Management*, Ph.D. thesis, Stanford University.

Ash ASUDEH (2009), Adjacency and Locality: a Constraint-Based Analysis of Complementizer-Adjacent Extraction, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG09 Conference*, pp. 106–126, CSLI Publications, Stanford, CA.

Ash ASUDEH (2012), *The Logic of Pronominal Resumption*, Oxford University Press, Oxford.

Ash ASUDEH and Richard CROUCH (2002), Coordination and Parallelism in Glue Semantics: Integrating Discourse Cohesion and the Element Constraint, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG02 Conference*, CSLI Publications, Stanford, CA.

Ash ASUDEH, Mary DALRYMPLE, and Ida TOIVONEN (2008), Constructions with Lexical Integrity: Templates as the Lexicon-Syntax Interface, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG08 Conference*, pp. 68–88, CSLI Publications, Stanford, CA.

Ash ASUDEH, Mary DALRYMPLE, and Ida TOIVONEN (2013), Constructions with Lexical Integrity, *Journal of Language Modelling*, 1(1):1–54.

Ash ASUDEH and Gianluca GIORGOLO (2012), Flexible Composition for Optional and Derived Arguments, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG12 Conference*, pp. 64–84, CSLI Publications, Stanford, CA.

Ash ASUDEH, Gianluca GIORGOLO, and Ida TOIVONEN (2014), Meaning and Valency, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG14 Conference*, pp. 68–88, CSLI Publications, Stanford, CA.

Joan BRESNAN (2001), *Lexical-Functional Syntax*, Blackwell, Oxford.

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

Miriam BUTT (1994), Machine Translation and Complex Predicates, in Harald TROST, editor, *Proceedings of KONVENS '94 "Verarbeitung natürlicher Sprache", Vienna, September 28–30, 1994*, pp. 62–71, Springer, Berlin.

Miriam BUTT (1995), *The Structure of Complex Predicates in Urdu*, CSLI Publications, Stanford, CA.

Miriam BUTT (1997), Complex Predicates in Urdu, in Alex ALSINA, Joan BRESNAN, and Peter SELLS, editors, *Complex Predicates*, CSLI Publications, Stanford, CA.

Miriam BUTT (1998), Constraining Argument Merger Through Aspect, in Erhard HINRICHS, Andreas KATHOL, and Tsuneko NAKAZAWA, editors, *Complex Predicates in Nonderivational Syntax*, pp. 73–113, Academic Press, San Diego, CA.

Miriam BUTT (2014), Control vs. Complex Predication: Identifying Non-Finite Complements, *Natural Language & Linguistic Theory*, 32(1):165–190.

Miriam BUTT, Tina BÖGEL, Annette HAUTLI, Sebastian SULGER, and Tafseer AHMED (2012), Identifying Urdu Complex Predication via Bigram Extraction, in *Proceedings of the 24th International Conference on Computational Linguistics (COLING12)*, pp. 409–424.

Miriam BUTT, Helge DYVIK, Tracy Holloway KING, Hiroshi MASUICHI, and Christian ROHRER (2002), The Parallel Grammar Project, in J. CARROLL, N. OOSTIJK, and R. SUTCLIFFE, editors, *Proceedings of COLING 2002: Workshop on Grammar Engineering and Evaluation*, pp. 1–7.

Miriam BUTT and Wilhelm GEUDER (2001), On the (Semi)lexical Status of Light Verbs, in Norbert CORVER and Henk VAN RIEMSDIJK, editors, *Semi-Lexical Categories: On the Content of Function Words and the Function of Content Words*, pp. 323–370, Mouton de Gruyter, Berlin.

Miriam BUTT and Tracy Holloway KING (2006), Restriction for Morphological Valency Alternations: The Urdu Causative, in Miriam BUTT, Mary DALRYMPLE, and Tracy Holloway KING, editors, *Intelligent Linguistic Architectures: Variations on themes by Ronald M. Kaplan*, pp. 235–358, CSLI Publications, Stanford, CA.

Miriam BUTT and Tracy Holloway KING (2007), Urdu in a Parallel Grammar Development Environment, *Language Resources and Evaluation (Special Issue on Asian Language Processing: State of the Art Resources and Processing)*, 41:191–207.

Miriam BUTT, Tracy Holloway KING, and John T. MAXWELL, III (2003), Complex Predicates via Restriction, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG03 Conference*, CSLI Publications, Stanford, CA.

Miriam BUTT, Tracy Holloway KING, María-Eugenia NIÑO, and Frédérique SEGOND (1999), *A Grammar Writer's Cookbook*, CSLI Publications, Stanford, CA.

Miriam BUTT, Tracy Holloway KING, and Gillian RAMCHAND (2010), Complex Predication: How Did the Child Pinch the Elephant?, in Linda UYECHI and Lian Hee WEE, editors, *Reality Exploration and Discovery: Pattern Interaction in Language & Life. A Festschrift for K. P. Mohanan*, pp. 231–256, CSLI Publications, Stanford, CA.

Miriam BUTT, María-Eugenia NIÑO, and Frédérique SEGOND (1996), Multilingual Processing of Auxiliaries within LFG, in Dafydd GIBBON, editor, *Natural Language Processing and Speech Technology: Results of the 3rd KONVENS Conference, Bielefeld, October 1996*, pp. 111–122, Mouton de Gruyter, Berlin.

Miriam BUTT and Gillian RAMCHAND (2005), Complex Aspectual Structure in Hindi/Urdu, in Nomi ERTESCHIK-SHIR and Tova RAPOPORT, editors, *The Syntax of Aspect*, pp. 117–153, Oxford University Press, Oxford.

Liz CHRISTIE (2013), Result XPs and the Argument-Adjunct Distinction, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, pp. 212–231, CSLI Publications, Stanford, CA.

Dick CROUCH, Mary DALRYMPLE, Ronald M. KAPLAN, Tracy Holloway KING, John T. MAXWELL III, and Paula NEWMAN (2011), *XLE Documentation*, Palo Alto Research Center, Palo Alto, CA, http://www2.parc.com/isl/groups/nltt/xle/doc/xle_toc.html.

Dick CROUCH and Tracy Holloway KING (2006), Semantics via F-structure Rewriting, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG06 Conference*, CSLI Publications, Stanford, CA.

Mary DALRYMPLE (2001), *Lexical Functional Grammar*, Academic Press, San Diego, CA.

Mary DALRYMPLE, Vineet GUPTA, John LAMPING, and Vijay SARASWAT (1999), Relating Resource-based Semantics to Categorial Semantics, in Mary DALRYMPLE, editor, *Semantics and Syntax in Lexical Functional Grammar*, pp. 261–280, MIT Press, Cambridge, Mass., also in *Proceedings of the Fifth Meeting on Mathematics of Language* (MOL5), Schloss Dagstuhl, Saarbrücken, Germany, August 1997.

Mary DALRYMPLE, Angie HINRICHS, John LAMPING, and Vijay SARASWAT (1993a), The Resource Logic of Complex Predicate Interpretation, in K.-J. CHEN and C.-R. HUANG, editors, *Proceedings of the 1993 Republic of China Computational Linguistics Conference (ROCLING)*, Computational Linguistics Society of the Republic of China, Hsitou National Park, Taiwan, also published as Xerox Technical Report ISTL-NLTT-1993-08-03.

Mary DALRYMPLE, Ronald M. KAPLAN, and Tracy Holloway KING (2004), Linguistic Generalizations over Descriptions, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG04 Conference*, pp. 199–208, CSLI Publications, Stanford, CA.

Mary DALRYMPLE, John LAMPING, Fernando PEREIRA, and Vijay SARASWAT (1996), A Deductive Account of Quantification in LFG, in Makoto KANAZAWA, Christopher J. PIÑÓN, and Henriette DE SWART, editors, *Quantifiers, Deduction and Context*, pp. 33–57, CSLI Publications, Stanford, CA.

Mary DALRYMPLE, John LAMPING, and Vijay SARASWAT (1993b), LFG Semantics via Constraints, in *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 97–105, Association for Computational Linguistics.

Yehuda N. FALK (2001), *Lexical-Functional Grammar: an Introduction to Parallel Constraint-Based Syntax*, CSLI Publications, Stanford, CA.

Jamie FINDLAY (2014), Mapping Theory without Argument Structure, MS, University of Oxford.

John FRY (2005), Resource-logical Event Semantics for LFG, draft MS, based on a paper originally presented at LFG99, University of Manchester. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.84.6503&rep=rep1&type=pdf.

Dag T. T. HAUG (2008), Tense and Aspect for Glue Semantics: The Case of Participial XADJ's, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG08 Conference*, pp. 291–311, CSLI Publications, Stanford, CA.

Petr HOMOLA and Matt COLER (2013), Causatives as Complex Predicates without the Restriction Operator, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, pp. 316–334, CSLI Publications, Stanford, CA.

Akira ISHIKAWA (1985), *Complex Predicates and Lexical Operations in Japanese*, Ph.D. thesis, Stanford University.

Ronald M. KAPLAN and Joan BRESNAN (1982), Lexical-Functional Grammar: A Formal System for Grammatical Representation, in Joan BRESNAN, editor, *The Mental Representation of Grammatical Relations*, pp. 173–281, MIT Press, Cambridge, MA.

Ronald M. KAPLAN and Jürgen WEDEKIND (1993), Restriction and Correspondence-Based Translation, in *Proceedings of the 6th Meeting of the EACL*, pp. 193–202, European Association for Computational Linguistics, Utrecht.

Anna KIBORT (2001), The Polish Passive and Impersonal in Lexical Mapping Theory, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG01 Conference*, pp. 163–183, CSLI Publications, Stanford, CA.

Anna KIBORT (2004), *Passive and Passive-Like Constructions in English and Polish*, Ph.D. thesis, University of Cambridge.

Anna KIBORT (2006), On Three Different Types of Subjectlessness and How to Model Them in LFG, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG06 Conference*, CSLI Publications, Stanford, CA.

Anna KIBORT (2007), Extending the Applicability of Lexical Mapping Theory, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG07 Conference*, pp. 250–270, CSLI Publications, Stanford, CA.

Anna KIBORT (2008), On the Syntax of Ditransitive Constructions, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG08 Conference*, pp. 312–332, CSLI Publications, Stanford, CA.

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

Jonas KUHN (2001), Resource Sensitivity in the Syntax-Semantics Interface and the German Split NP Construction, in W. Detmar MEURERS and Tibor KISS, editors, *Constraint-Based Approaches to Germanic Syntax*, pp. 177–216, CSLI Publications, Stanford, CA.

John J. LOWE (2014), Gluing Meanings and Semantic Structures, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG14 Conference*, pp. 387–407, CSLI Publications, Stanford, CA.

John J. LOWE (2015), *Participles in Rigvedic Sanskrit: The Syntax and Semantics of Adjectival Verb Forms*, Oxford University Press, Oxford.

Tara MOHANAN (1994), *Argument Structure in Hindi*, CSLI Publications, Stanford, CA.

Stephanie NEEDHAM and Ida TOIVONEN (2011), Derived Arguments, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG11 Conference*, pp. 401–421, CSLI Publications, Stanford, CA.

Ghulam RAZA (2011), *Subcategorization Acquisition and Classes of Predication in Urdu*, Ph.D. thesis, University of Konstanz.

Sebastian SULGER (2012), Nominal Argument Structure and the Stage-/Individual-Level Contrast in Hindi-Urdu, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG12 Conference*, pp. 582–602, CSLI Publications, Stanford, CA.

Sebastian SULGER, Miriam BUTT, Tracy Holloway KING, Paul MEURER, Tibor LACZKÓ, György RÁKOSI, Cheikh Bamba DIONE, Helge DYVIK, Victoria ROSÉN, Koenraad DE SMEDT, Agnieszka PATEJUK, Özlem ÇETINOĞLU, I Wayan ARKA, and Meladel MISTICA (2013), ParGramBank: The ParGram Parallel Treebank, in *Proceedings of 51st Annual Meeting of the Association for Computational Linguistics, Sofia, Bulgaria, August 4–9, 2013*, pp. 550–560, Association for Computational Linguistics.

Ida TOIVONEN (2013), English Benefactive NPs, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, pp. 503–523, CSLI Publications, Stanford, CA.

Annie ZAENEN and Mary DALRYMPLE (1995), “Polymorphic” Causatives: Complex Predicates in French, Technical report, Information Sciences and Technologies Laboratory, Xerox PARC, Palo Alto, CA.

Annie ZAENEN and Mary DALRYMPLE (1996), Les Verbes Causatifs “Polymorphiques”: Les Prédicats Complexes en Français, *Langages*, 122:79–95.