Mapping theory without argument structure*

Jamie Y. Findlay
University of Oxford, Oxford, UK

ABSTRACT

Asudeh and Giorgolo (2012) offer an analysis of optional and derived arguments that does away with argument structure as a separate level of representation within the architecture of Lexical Functional Grammar in favour of encoding much of this information in a connected semantic structure. This simplifies the architecture in many ways, but leaves open the question of the mapping between thematic roles, arguments, and grammatical functions (traditionally explored under the umbrella of Lexical Mapping Theory; LMT: Bresnan and Kanerva 1989). In this paper, I offer a formalisation of these mapping relations, drawing on a modern reanalysis of traditional LMT (Kibort 2007), while also continuing Asudeh and Giorgolo's (2012) quest to evacuate as much information as possible out of individual lexical entries and into cross-categorising templates (Dalrymple *et al.* 2004; Crouch *et al.* 2012).

Keywords: *argument structure, mapping theory, argument linking, LFG, Glue Semantics*

*This work has benefitted enormously from discussion and ideas which I was exposed to at the Glue Group at the University of Oxford and during Anna Kibort's illuminating lectures on argument structure in Hilary Term 2013. I would like to express my thanks to all involved, and especially to Ash Asudeh for his detailed comments on an earlier draft. Thank you also to the anonymous JLM reviewers, whose thoughtful criticism has made this a far better paper. Any remaining errors are, as ever, entirely my own. This work was completed while I was the recipient of a UK Arts and Humanities Research Council studentship (grant reference AH/K503198/1), which I gratefully acknowledge.

1 INTRODUCTION

This paper makes a contribution to the theoretical frameworks of Lexical Functional Grammar (LFG: Kaplan and Bresnan 1982; Bresnan 2001; Dalrymple 2001; Falk 2001; Bresnan *et al.* 2016; Asudeh and Toivonen 2015) and Glue Semantics (Glue: Dalrymple 1999, 2001; Asudeh 2012). Some relevant formalisms will be explained where possible, but constraints of space prevent a full introduction to the two theories here.

The main purpose of this paper will be to show that current work by Anna Kibort (Kibort 2001, 2007, 2008, 2014) on Lexical Mapping Theory (LMT) is compatible with a proposal by Asudeh and Giorgolo (2012) (hereafter A&G) to do away with argument structure as a separate level of representation in the formal architecure of LFG, and to demonstrate how the two theories can be integrated.

The paper is structured as follows. Section 2 discusses what we want from a mapping theory in general, and introduces LMT. Following this, the key points of Kibort's version of LMT are briefly sketched in Section 3, while Section 4 discusses the role of argument structure, and introduces A&G's suggestion to do without it. Section 6 contains the main proposal of the paper, namely a formalism which allows the insights of Kibort's LMT to be combined with A&G's abandonment of argument structure. This section ends with examples of how two argument alternations, the passive and the benefactive, can be treated in the new theory. Finally, Section 7 offers conclusions.

2 (LEXICAL) MAPPING THEORY

Mapping theories attempt to find general principles by which arguments and grammatical functions are related, thus avoiding repeated (and redundant) lexical stipulation. It is not a coincidence, so the theory goes, that the Agent arguments in verbs like *hit*, *select*, *put*, or many others are usually syntactically realised as subjects, while the Patient-like arguments are usually direct objects.[1]

The traditional work on this problem in LFG is Lexical Mapping Theory (LMT: Bresnan and Kanerva 1989; Bresnan 1990; Butt *et al.* 1997). However, this name may not be entirely apposite. As several

[1] At least in syntactically accusative languages.

authors have pointed out (e.g. Butt 1995; Alsina 1996), "the theory cannot apply exclusively to individual words" (Dalrymple 2001, 212), since various problems generally thought to fall under the umbrella of LMT can involve *multiple* lexemes which combine to form complex predicates in the syntax (for example, causatives are formed analytically in some languages, e.g. Romance, even if they are synthetic in others).[2] For this reason, I follow the recent trend in dropping the 'lexical' and referring to this theory simply as *mapping theory*. I will, though, continue to use the term 'LMT' when discussing researchers, like Kibort, who explicitly position their work as belonging to this tradition.

What do we expect of such a theory (whatever we call it)? If the relationship between grammatical functions and arguments were simple or straightforward, there would be nothing to a mapping theory other than a listing of the recorded correspondences for each language. However, there is no one-to-one mapping between particular roles and particular grammatical functions (GFs). There are many operations which alter the mapping between the two, such as locative inversion, the passive, the applicative, or the causative. Some, such as the passive or the applicative, are described as *morphosyntactic*, in that they do not involve a change in (truth-conditional) meaning – they merely realign participants and grammatical functions.[3] Others, such as the causative, are *morphosemantic* in that they add additional participants or change the roles of existing participants, and thus change the truth-conditional meaning of the predicate.

At the very least, mapping theory must explain the morphosyntactic alternations. Ideally, it should also offer a principled account of the morphosemantic ones: Kibort (2007), for example, suggests an extension to traditional LMT which allows it to account for morphosemantic as well as morphosyntactic alternations.

[2] Although see Ackerman *et al.* (2011) for a dissenting view on the role of syntax in predicate formation.

[3] Of course, they alter other aspects of 'meaning', in the broader sense of the word, such as information structure or pragmatics. This is not surprising, for it would indeed be strange to discover that there were truly 'gratuitous' alternations that merely added complexity to the grammar with no corresponding communicative payoff.

Let us consider an example, that of the passive, which is a morphosyntactic alternation. A transitive verb like *devour* takes two arguments: a devourer and a devourum (the thing devoured). In sentence (1), the devourer argument is associated with the SUBJect GF, and the devourum with the OBJect, while in (2), the passive, the devourum is now the SUBJ, and the devourer is either unexpressed, or realised as an OBLique *by*-phrase:

(1) Jeremy devoured the pizza.

(2) The pizza was devoured (by Jeremy).

But such alternations are not unrestricted: in English, there is no purely morphosyntactic operation which would make the devourer an object, as in (3), and none which would make the devourum an oblique, as in (4), for example:[4]

(3) a. * The pizza devoured Jeremy. [With the intended meaning.]

b. * It devoured Jeremy ((by/to/…) the pizza).

(4) * Jeremy devoured by/to/… the pizza.

Any theory of mapping must explain why the alternation in (1)–(2) is possible, while others are not. This means we need to be able to restrict the type of GF an argument can be associated with, but not simply by reducing it to one. The standard approach has been underspecification by features, to which we now turn.

[4] It may be that such alternations exist in other languages: for example, if the difference between actor voice and undergoer voice in some Western Austronesian languages is truly a voice alternation (Himmelmann 2002), then this might be an example of a morphosyntactic alternation which has the form exemplified in (3a).

As an anonymous reviewer points out, there may also be morpho*semantic* alternations which do involve such alignments. For example, (4) corresponds to the antipassive or deobjective in Slavic languages (Fehrmann *et al.* 2010, 207–208). What is more, if we consider lexical relationships, the correspondence between verb pairs like *fear* and *frighten* might be thought to realise the alternation between (1) and (3a), whereby the subject in one member of the pair corresponds to the object in the other. Such lexical relatedness goes beyond the scope of mapping theory, however.

2.1 *Grammatical functions decomposed*

In standard LMT, the four-way cross-classification of GFs given in (5) (after Bresnan and Kanerva 1989) is assumed:

(5)

	$-r$	$+r$
$-o$	SUBJ	OBL_θ
$+o$	OBJ	OBJ_θ

SUBJ, OBJ, and OBL_θ are the subject, (direct) object, and oblique functions more or less familiar from traditional grammars. OBJ_θ may be less familiar: this is the so-called secondary or restricted object, as in the second object of English dative-shifted *give*:

(6) Kim gave Colin **his book**.

The necessity of theorising such a GF has been contested, but it is still taken as standard in mainstream LFG, and so I will continue to use it here (see Kibort 2013 for a defence of the status of OBJ_θ).

The two features, $[o]$ and $[r]$, refer, respectively, to the ***object***-like properties of a GF, and to whether it is semantically ***restricted*** or not. Thus, there are two objective ($[+o]$) GFs, namely OBJ and OBJ_θ, and two non-objective ($[-o]$) ones, *viz.* SUBJ and OBL_θ. Similarly, there are two semantically restricted ($[+r]$) GFs, OBL_θ and OBJ_θ, and two non-restricted ($[-r]$) ones, SUBJ and OBJ.

With this in place, the solution to the *devour* question above becomes straightforward. In the standard theory, we simply associate each argument with a single feature, which then limits its choice of GF to two. We saw that the devourer argument could be realised as a SUBJ or as an OBL;[5] thus, in the mapping theory, it is linked with a $[-o]$ feature, and can therefore surface as a SUBJ or an OBL (but not an OBJ, for example), just as needed. Meanwhile, the devourum is marked as $[-r]$, and can thus be realised as an OBJ or a SUBJ (but not an OBL, for example), again just as observed. A separate mechanism is required to determine which argument gets priority in selecting a particular GF – this is usually explained by reference to a thematic hierarchy of some kind, although there is a lack of agreement over the

[5] For the sake of parsimony, and to avoid being drawn into a debate about exactly what information could be the realisation of θ in OBL_θ (see also fn. 18, below), I will use OBL as shorthand for OBL_θ when the exact nature of the subscript/index is unimportant.

exact form this should take (Newmeyer 2002, 65ff.; Levin and Rappaport Hovav 2005, ch. 6). In the analysis presented here, we will use a different mechanism.

2.2 *The status of the features [o] and [r]*

A natural question to raise at this stage is that of the status of these features. Certainly, they are intended to cross-classify the grammatical functions. But it would seem from the definitions that they are intended to *constitute* the GFs somehow, as well. That is, they actually contribute some information related to semantic restrictedness or objectivity – though of course these terms then raise their own definitional questions.

One possibility is that the familiar GF labels are really just abbreviations for feature structures incorporating these mapping features. This is the approach hinted at by Falk (2001, 109, fn. 12), for example. On this view, the label SUBJ is really just a shorthand way of writing the f-structure in (7), and the f-structure given in (8) is a shorthand way of writing the fully expanded f-structure in (9):

(7)
$$\begin{bmatrix} \text{R} & - \\ \text{O} & - \end{bmatrix}$$

(8)
$$\begin{bmatrix} \text{PRED} & \text{'love'} \\ \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'Trevor'} \end{bmatrix} \\ \text{OBJ} & \begin{bmatrix} \text{PRED} & \text{'Elliot'} \end{bmatrix} \end{bmatrix}$$

(9)
$$\begin{bmatrix} \text{PRED} & \text{'love'} \\ \begin{bmatrix} \text{R} & - \\ \text{O} & - \end{bmatrix} & \begin{bmatrix} \text{PRED} & \text{'Trevor'} \end{bmatrix} \\ \begin{bmatrix} \text{R} & - \\ \text{O} & + \end{bmatrix} & \begin{bmatrix} \text{PRED} & \text{'Elliot'} \end{bmatrix} \end{bmatrix}$$

Now, in the standard theory, attribute-value structures such as (7) are only permitted as *values* of attributes, not as attributes themselves. F-structures are defined as functions from their attributes to their values, and the domain of those functions does not include those functions themselves. Thus, to allow structures like (9) is to alter the mathematical properties of f-structures, so that their domains no longer include only simple atomic values, but also sets (specifically, functions). Perhaps this is what we need, but it is worth noting that it is not simply a notational variant.

Such a move also represents a departure from one of LFG's foundational theoretical principles, namely that grammatical functions are primitives in some sense. Now the features R and O are the primitives instead.[6]

In matter of fact, we do not need to answer the theoretical questions lurking behind the decompositional approach to GFs in order to take advantage of it. By appealing to these features we are making empirical claims: if it is true that there are mapping phenomena which are sensitive to the $[\pm o]/[\pm r]$ distinction, then we have determined that some pairings/alternations of GFs should be ruled out. For example, there is no way, at least not using a single feature, of describing just the pair SUBJ and OBJ_θ, or the pair OBJ and OBL, and so (purely morphosyntactic) alternations involving these pairs should be ruled out. They do not form a natural class. This is an empirical claim, and in order to describe it, it is enough to see the $[\pm o]/[\pm r]$ distinction as merely mnemonic, describing four sets of pairs which can be linked to arguments by whatever mechanism we choose to use. Thus, abstracting away from the theoretical questions, we can use disjunctions to define the following feature decompositions (suggested to me by Ron Kaplan, p.c.):[7]

(10) MINUSO $\equiv$ $\{\text{SUBJ}|\text{OBL}_\theta\}$

(11) PLUSO $\equiv$ $\{\text{OBJ}|\text{OBJ}_\theta\}$

(12) MINUSR $\equiv$ $\{\text{SUBJ}|\text{OBJ}\}$

(13) PLUSR $\equiv$ $\{\text{OBL}_\theta|\text{OBJ}_\theta\}$

[6] Butt (1995, 31) makes this claim explicitly, saying that "[w]hile it may appear that grammatical functions like SUBJ, OBJ, etc. exist as primitive notions within the theory, a given grammatical function, a SUBJ for example, is actually nothing more and nothing less than the features $[-r, -o]$. Grammatical functions thus are not independent of the features, but are instead defined and therefore also constrained by them".

[7] These are written in the regular language used in LFG *functional descriptions* (see Asudeh 2012, 64–65). The expression $\{A|B\}$ represents a disjunction between A and B.

In essence, this approach sidesteps the theoretical issues raised by the decompositional approach and simply co-opts its empirical claims.

2.3 *Optionality of grammatical functions*

One other assumption I will be making that is relevant in considering the theory of mapping presented here is that all GFs are optional: the syntactic constraints of Coherence and Completeness (see Kaplan and Bresnan 1982, 211–212, and Dalrymple 2001, 35–39, for formal definitions and discussion) are subsumed by considerations of *resource sensitivity* in a Glue-based semantics (see discussion in Dalrymple 1999; Kuhn 2001; Asudeh 2012, ch. 5). That is, the presence of all and only the arguments required by a predicate is constrained by the linear logic component of Glue: incoherence leads to resource surplus, while incompleteness leads to resource deficit. When writing f-structures, therefore, I will give PRED values as simple semantic forms in single quotation marks (e.g. 'select'), omitting the traditional GF-selection/subcategorisation information usually given inside and outside angled brackets (e.g. 'select ⟨SUBJ, OBJ⟩').[8]

3 KIBORT'S LMT

Kibort (2001, 2007, 2008, 2014) has argued for a number of modifications to LMT, most importantly for a return to the separation implied by earlier work (e.g. Bresnan 1982) between thematic roles and *argument positions*, intermediary objects standing between thematic roles and the grammatical functions which realise them. Later work collapsed this distinction, conflating thematic roles with argument positions, which then reduces the problem of mapping to that of linking thematic roles to GFs directly. If the focus of mapping theory is purely morphosyntactic operations, this is perhaps understandable, but Kibort (2007) argues for extending the scope of LMT to include

[8] The main obstacle to relegating Coherence and Completeness to the semantics is expletive arguments, i.e. those which are required by the syntax but not the semantics, and which therefore might be thought not to make any semantic contribution. Clearly, resource sensitivity will not help us if such arguments are not included in the resource accounting in the first place. This problem is not insurmountable, however: see Asudeh (2012, 113) for some suggestions about how to resolve the problem without resorting to subcategorisation via the PRED feature.

morphosemantic operations as well, and here it is important to allow participants to realign with respect to their thematic roles (more on this below).

Kibort therefore suggests that argument structure is made up of a list of argument positions, each of which has associated with it an intrinsic assignment of syntactic features (or, ultimately, a pair of GFs, as we are thinking about it), but which can be associated with different thematic roles. Predicates have open to them a universal subcategorisation frame, from which they select a certain number of arguments. The intrinsic assignments are as given in (14):[9]

(14)	$<$	arg_1	arg_2	arg_3	arg_4	…	arg_n	$>$
		$[-o]$	$[-r]$	$[+o]$	$[-o]$		$[-o]$	

These argument positions are ordered, and a predicate can select any combination of them – that is, not necessarily a contiguous subsection: a predicate could select an arg_1 and an arg_4, for example – but there can only be one of each: e.g. there cannot be two arg_2s. As the arg_n notation makes clear, there can be more than four arguments; however, all arguments above arg_4 will be of the same syntactic type as an arg_4 (namely, $[-o]$).[10,11]

[9] In the full theory, arg_1 is associated with $[-o]$ in unergative verbs and $[-r]$ in unaccusative ones; I simplify here, since the only verbs we will be looking at require $[-o]$.

[10] As a reviewer notes, this means it is, in a certain sense, possible to have 'more than one arg_4', in that there may be more than one argument position of the same syntactic type as arg_4. However, such additional arguments would be distinguished by their subscripts, so that if there are two 'arg_4's, one will in fact be an arg_5.

[11] Kibort's stance on the uniqueness of argument positions does not seem wholly consistent. In some works, argument positions are described as being "unique" (Kibort 2007, 259), while in others it is explicitly claimed that multiple arg_3s, for example, are permitted (Kibort 2008, 330). Assuming that s-structures share the same functional properties as f-structures, the proposal I give in Section 6 does not allow for multiple argument positions with the same name, which means it may not be able to handle the case of multiple applicatives discussed in Kibort (2008). However, I am concerned that Kibort's proposals to resolve this problem raise issues for the internal coherence of her own system: if there are multiple argument positions with the same name, it is not clear to me how the mapping principles are to distinguish them.

In addition to the argument positions being ordered, we can derive a partial ordering on grammatical functions from their decomposition into features, which ranks GFs from least to most marked, where being marked is equated with having more + features (Bresnan *et al.* 2016, 331):

(15) $\text{SUBJ} > \text{OBJ}, \text{OBL}_\theta > \text{OBJ}_\theta$

Mapping is then simply linking the highest arg position to the highest available GF (with appropriate restrictions such as Function-Argument Biuniqueness (Bresnan 1980) to prevent multiple arguments mapping to the same GF). Let us see a brief example of how this works.

A verb like *select* will have the following argument structure:

(16)
$$\begin{array}{llllll} \textit{select} & < & \text{arg}_1 & \text{arg}_2 & > \\ & & [-o] & [-r] & \end{array}$$

If there is no further specification, the highest argument position, arg_1, will then map to the highest available $[-o]$ GF, in this case the SUBJ. The next argument, arg_2, then maps to the highest available $[-r]$ GF, in this case the OBJ, which is exactly the pattern we want for an active voice transitive verb.

The passive alternation can now be easily explained as an operation which further restricts arg_1 to $[+r]$ (Kibort 2001), giving us the following argument structure:

(17)
$$\begin{array}{lllll} \textit{select}_{\text{PASS}} & < & \text{arg}_1 & \text{arg}_2 & > \\ & & [-o] & [-r] & \\ & & [+r] & & \end{array}$$

The mapping now follows straightforwardly, using the same procedure. The first argument, arg_1, maps to the highest available GF which satisfies its feature requirements: in the present case, this is uniquely described, since the only GF which is both $[-o]$ and $[+r]$ is OBL. The next argument, arg_2, then maps to the highest available $[-r]$ GF, which is now the SUBJ.

Obligatorily three-place predicates like *put* will have the argument structure below:

(18)
$$\begin{array}{llllll} \textit{put} & < & \text{arg}_1 & \text{arg}_2 & \text{arg}_4 & > \\ & & [-o] & [-r] & [-o] & \end{array}$$

In the active, this will correctly specify the three GFs as SUBJ, OBJ, and OBL. But importantly, it will also provide the correct analysis of the passive, whereby the direct object can be 'promoted' to subject, but not the (object within the) prepositional phrase, as exemplified in (19)–(20):

(19) The cup was put on the table.

(20) * On the table was put the cup./* The table was put the cup on.

The argument structure for passive *put* is as follows:

(21) put_{PASS} $<$ arg_1 arg_2 arg_4 $>$

put_{PASS}	$<$	arg_1	arg_2	arg_4	$>$
		$[-o]$	$[-r]$	$[-o]$	
		$[+r]$			

If we follow the same mapping procedure as before, we can see that we obtain the correct results: arg_1 once again maps to OBL; arg_2, the next highest argument, then maps to SUBJ, thus preventing arg_4 from doing so; arg_4 maps to the highest available $[-o]$ GF, which is OBL (it is not a problem that there are two OBL arguments, since they will be distinguished by their indices, whatever these may be: for example, arg_1 might correspond to an $\text{OBL}_{\text{AGENT}}$ and arg_4 to an OBL_{GOAL}).

Kibort's analysis offers a simple and general solution to many of the traditional mapping problems, but it is obviously based in a theory where argument structure has a fundamental role. In the next section, I present evidence that we should do away with argument structure as a separate level of representation. The challenge then is to retain the advantages of Kibort's LMT in a formalism without a-structure. This is the topic of Section 6.

4 THE PROBLEM WITH ARGUMENT STRUCTURE

In the LFG conception of the architecture of the grammar, a modularity is assumed such that different components of the grammar (morphology, phonology, syntax, etc.) are treated as separate levels of structure, related by what are called correspondence functions. Of particular interest are the two levels of syntactic representation, *c(onstituent)-structure* (phrase structure) and *f(unctional)-structure* (which represents grammatical relations such as *subject of* and *object of* in an attribute-value matrix), and the level of the syntax-semantics interface, *s(emantic)-structure*. (For more on these structures, see Dalrymple 2001, 45–68, 7–44, 230–240, respectively; on the correspondence architecture, see Dalrymple 2001, 180–182ff., and Asudeh 2012, 49–54.)

Generally, a level of *a(rgument)-structure* is also assumed, which encodes the lexical arguments of a predicate, and controls their linking to grammatical functions. However, there are questions over where exactly such a level should appear in the architecture of the grammar, or indeed whether such an independent level of representation is needed. A&G argue, based on problems caused by predicates taking optional arguments, that it is best to do away with a-structure, and relegate most of its functions to an augmented s-structure. In Section 4.1, I present their reasoning. However, we may come to the same conclusion independently, via considerations of a more abstract or meta-theoretical nature, and Section 4.2 explores these.[12]

4.1 *Optional arguments*

Certain verbs, such as *eat* or *drink*, express their Patient argument in the syntax only optionally:

(22) a. Pedro ate the cake earlier.
     b. Pedro ate earlier.

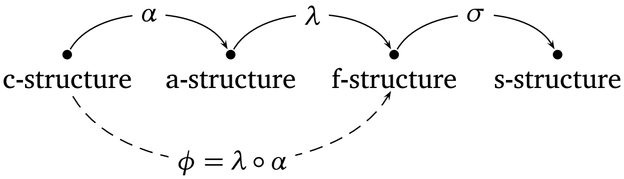
(23) a. Amanda drank her coffee quickly.
     b. Amanda drank quickly.

Nonetheless, this Patient argument must still be present in the verb's argument structure – it remains, after all, part of the core relation expressed by the verb – and must also be represented at semantic structure, since it is interpreted semantically – for *John ate* to be true, John must have eaten *something*. But, A&G argue, this means that the standard conception of the LFG correspondence architecture is inadequate.

Since Butt *et al.* (1997), the canonical view in LFG has been that a-structure should be treated as a separate level of representation in between c-structure and f-structure. This means that the traditional ϕ-function, which maps from c-structure to f-structure, is then seen as

[12] I thank an anonymous reviewer for their helpful observations on this point.

the composition of two new functions: the α-function from c-structure to a-structure, and the λ-function from a-structure to f-structure. The correspondence function from f-structure to s-structure remains the σ-function. This architecture is shown schematically in (24):

(24)


However, if 'optional' arguments appear at a-structure and s-structure, but not f-structure, we must posit a new correspondence function directly between a-structure and s-structure (which A&G call the θ-function) in order to bypass f-structure. This situation is shown in (25):

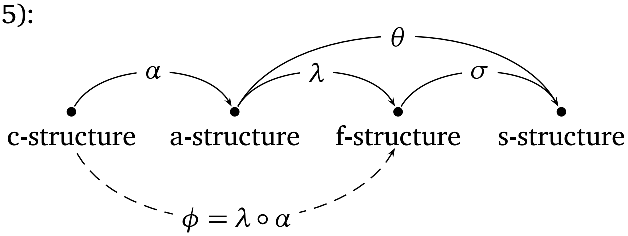
(25)


Figure 1 gives the relevant structures and correspondences for the sentence *Kim ate at noon* in this conception of the standard theory.

If we consider the PATIENT argument at a-structure, we see that it does not map to any grammatical function at f-structure. This means that we cannot reach its s-structure correspondent, p, by the normal means of composing the λ- and σ-functions, thereby passing through the f-structure – instead, we need a new, separate function, θ.

If the s-structures were not unconnected, as they are in standard LFG (wherein each of e, k and p are separate, unconnected entities, as in Figure 1), one alternative would be to pass along the outermost structures via the usual correspondence functions until one reached the semantic structure for the clause, then go from that structure, e, to the PATIENT's s-structure, p, via some internal path. However, since in the present setup there is no relation expressed at semantic structure between e and p, this is impossible.

Thus, given the standard architecture, there is no way to relate the PATIENT with its s-structure, p, except via the proposed new func-

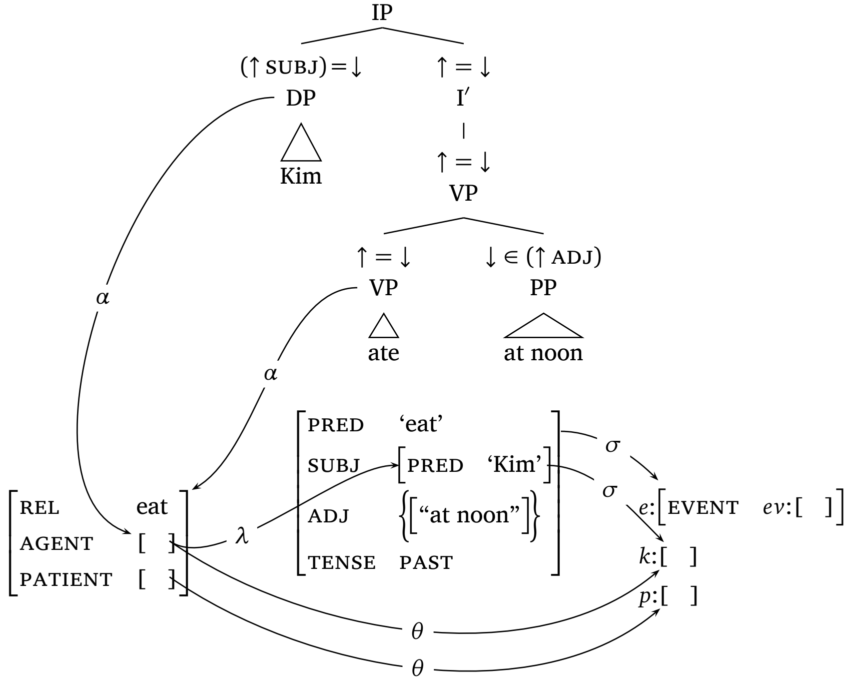


Figure 1: Relevant structures and correspondences for *Kim ate at noon* (after Asudeh and Giorgolo 2012, 70, Figure 1)

tion, θ. But, not only does making use of this new function add extra theoretical complexity, it also introduces a degree of indeterminacy into the grammar. There are now two correspondences between arguments which *are* realised syntactically (such as the AGENT argument in Figure 1) and their semantic structures, either via θ or via $\sigma \circ \lambda$. Therefore, instead of taking this option, A&G propose to make use of an architecture which does away with a-structure as a separate level of representation altogether, and with it the α-, λ-, and θ-functions (returning the ϕ-function to its former, underived, status). The information previously captured at a-structure is now encoded in a connected semantic structure. An analysis of the same sentence following this approach is given in Figure 2. A&G assume an event semantics for their meaning language, such that thematic roles are functions from events to individuals (Parsons 1990), and so avoid redundancy by using attributes like ARG_1 rather than AGENT in the semantic structure.[13]

[13] The framework suggested by A&G and elaborated on in this paper does not necessitate this treatment of thematic roles, and would be compatible with a

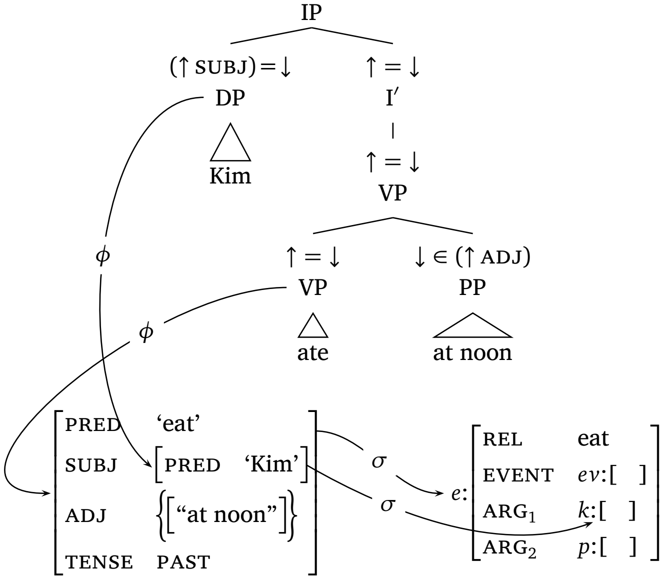


Figure 2:
Alternative analysis of *Kim ate at noon* (after Asudeh and Giorgolo 2012, 72, Figure 2)

A&G summarise the advantages that their approach brings as follows (p. 71):

1. We achieve a simplified architecture, which eliminates a separate a-structure projection, without losing information.
2. We do not lose linking relations and they are still post-constituent structure.[14]
3. We remove the non-determinacy that results from the presence of both the λ and θ correspondence functions.
4. Many of the meaning constructors for semantic composition are more elegant and simplified.

grammar that did without events in the semantics and instead treated thematic roles as e.g. attributes in s-structure (although of course appropriate modifications would be required). However, I consider it a strength of the present approach that it removes mention of thematic roles from the grammar; this is a view shared by Kibort (2007), and which I discuss further in Section 6.1.

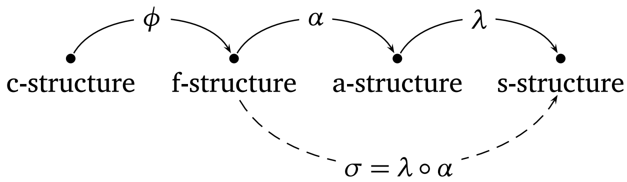
[14] Because complex predicates can correspond to more than one node at c-structure, but to a single, complex a- or s-structure, it is important that linking relations should be post-constituent structure so that they remain many-to-one (and still functional), rather than one-to-many (and so not; see Butt 1995 and Alsina 1996).

5. We regain the simple, traditional ϕ mapping from c-structure to f-structure.
6. We gain a connected semantic structure.

The form of A&G's argument is thus as follows. The location of a-structure in the correspondence architecture leads to theoretical complexity and redundancy when we consider optional arguments. One solution is to encode the information represented at a-structure somewhere post-f-structure. S-structure is post-f-structure, therefore one solution would be to encode it here. This also has the advantage of ontological parsimony: we have one less structure in our grammatical architecture.

4.2 *The role of a-structure*

Another, albeit less parsimonious, solution would be to relocate a-structure in between f- and s-structure, rather than collapsing it into the latter. That is, we might propose the architecture in (26) (here it is the σ-function which must be complexified, instead of the ϕ-function):

(26)


Aside from the problems posed by optional arguments, such a move has some independent motivation. Argument structure is generally seen as the interface between (lexical) meaning, including thematic roles, and syntax, in the form of the realisation of arguments as grammatical functions. But in the canonical architecture (in (24), above), a-structure stands between two levels of *syntax*, c-structure and f-structure, not between the syntax and the semantics. The modified architecture in (26) succeeds in remedying this situation.

However, by putting a- and s-structure in direct proximity like this, we draw attention to their potential similarities. S-structure is explicitly conceived of as the interface between syntax and semantics, acting as a syntactically-derived scaffold on which the linear logic of Glue can operate to control semantic composition. But a-structure is also an interface between syntax and semantics, relating GFs to the roles they play in the meaning. Thus, to avoid redundancy, we might well ask whether it is possible to collapse the two structures.

Butt (1995) argues that an independent level of a-structure is needed, but her conception of a-structure is highly semantic: adapted from Jackendoff's (1990) Lexical Conceptual Structures, it includes a large amount of lexical meaning, such as aspectual information. Butt's (1995) reliance on a-structure may be an artefact of the time of writing, when the semantic component of LFG was underdeveloped – she does not discuss s-structure at all, for example. If the two levels of representation are really doing the same work, or contributing different facets of the same information, then it makes sense to collapse them.

If we want to achieve such parsimony, however, we must ensure that we are not generating additional problems at the same time as we simplify our ontology. Since mapping theories are usually reliant on a separate level of argument structure, we must be able to provide a new theory which is instead based on s-structure. The purpose of the current paper is to do just this, and to give a mapping theory which is compatible with the architecture of the grammar proposed by A&G. Before we come to this, however, I wish to discuss another motivation of their paper.

5 LEXICAL GENERALISATIONS VIA TEMPLATES

Aside from the removal of argument structure as a separate level of representation, the other major theme in A&G's paper is an attempt to abstract as much information as possible away from individual lexical entries and into *templates* (Dalrymple *et al.* 2004; Crouch *et al.* 2012; Asudeh *et al.* 2013), which are shared by multiple lexical items.

Templates are shorthand ways of abbreviating functional descriptions and other information included in lexical entries. This means that a grammar which includes templates is extensionally equivalent to one which does not, since templates serve only as abbreviations. However, templates can be used to capture commonalities and to express linguistic generalisations, which means that, while a grammar with templates may be equivalent to one without them, the former may be able to capture generalisations which the latter cannot (A&G, p. 78).

Templates can be used to name functional descriptions. For example, we might define the templates SG-SUBJ and 1-SUBJ as in (27)–(28):

(27) SG-SUBJ :=
 (↑ SUBJ NUMBER) = SG

(28) 1-SUBJ :=
 (↑ SUBJ PERSON) = 1

We can then build up more complex templates from these:

(29) 1SG-SUBJ :=
 @1-SUBJ
 @SG-SUBJ

The '@' symbol represents a 'call' of the following template; i.e. that line is to be expanded into the contents of the template named in the call. Thus, (29) is equivalent to (30):

(30) 1SG-SUBJ :=
 (↑ SUBJ PERSON) = 1
 (↑ SUBJ NUMBER) = SG

Templates can be made a little more flexible by allowing them to take arguments. For example, we can define a template PERSON, such that (28) is equivalent to (32):

(31) PERSON(X) :=
 (↑ SUBJ PERSON) = X

(32) @PERSON(1)

We can do something similar for NUMBER, and then define a general SUBJECT template which takes two arguments, the person and the number of the predicate's subject:

(33) NUMBER(X) :=
 (↑ SUBJ NUMBER) = X

(34) SUBJECT(P, N) :=
 @PERSON(P)
 @NUMBER(N)

Now (29) is equivalent to (35):

(35) @SUBJECT(1, SG)

Templates can also contain meaning constructors, since these are included in the functional description:

(36) FUTURE :=

$(\uparrow \text{TENSE}) = \text{FUTURE}$

$\lambda P.\exists e[P(e) \wedge \textit{future}(e)] : [(\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$

This template would be called by a future tense verb, and provides the relevant f-structural information about tense, as well as a meaning constructor which existentially closes the predicate's event argument and specifies that it occurs in the future.

Combining all of the above, the lexical entry for the Latin verb *bibam*, 1st person singular future tense of 'drink', would be as follows (ignoring questions of mapping for the time being):[15]

(37) *bibam* V $(\uparrow \text{PRED}) = \text{'drink'}$

@SUBJECT(1, SG)

@FUTURE

$\lambda y \lambda x \lambda e.\textit{drink}(e) \wedge \textit{agent}(e) = x \wedge \textit{patient}(e) = y :$

$(\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

This is equivalent to the same lexical entry with all of the templates spelt out fully:

(38) *bibam* V $(\uparrow \text{PRED}) = \text{'drink'}$

$(\uparrow \text{SUBJ PERSON}) = 1$

$(\uparrow \text{SUBJ NUM}) = \text{SG}$

$(\uparrow \text{TENSE}) = \text{FUTURE}$

$\lambda P.\exists e[P(e) \wedge \textit{future}(e)] : [(\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$

$\lambda y \lambda x \lambda e.\textit{drink}(e) \wedge \textit{agent}(e) = x \wedge \textit{patient}(e) = y :$

$(\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

[15] Since Latin is pro-drop, this entry should also include (i):

(i) $((\uparrow \text{SUBJ PRED}) = \text{'PRO'})$

I omit this in the text for the sake of simplicity.

The use of templates allows us to streamline lexical entries, make them more readable, and talk about commonalities across lexical entries, in terms of named, shared f-descriptions. One area in which A&G put templates to work is in evacuating as much information as possible about semantic composition from individual lexical entries into cross-categorising patterns like AGENT-PATIENT-VERB, which describes all verbs that take an Agent and a Patient argument. An example is given in (39) (A&G, p. 78, their (37)):[16]

(39) AGENT-PATIENT-VERB :=

$$\lambda P \lambda y \lambda x \lambda e. P(e) \wedge \textit{agent}(e) = x \wedge \textit{patient}(e) = y :$$
$$[(\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma] \multimap$$
$$(\uparrow_\sigma \text{ ARG}_2) \multimap (\uparrow_\sigma \text{ ARG}_1) \multimap (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$$

This would be called by Agent-Patient verbs like *hit*, or *select*, which would have the following lexical entry:

(40) *select* V (↑ PRED) = ‘select’
@AGENT-PATIENT-VERB

$$\lambda e.\textit{select}(e) : (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$$

The only meaning that verbs contribute directly is the type of event they describe (the last line in (40)). The additional compositional work is done by the meaning constructor given in the AGENT-PATIENT-VERB template: it consumes the function contributed by the verb itself, predicates that of an event, and then provides thematic information on the meaning side, while on the linear logic side it returns a resource parallel in form to the familiar transitive verb resource (i.e. a dependency on the arguments of the verb which produces the meaning of the sentence – we now make use of the connected semantic structure positions rather than projections of grammatical functions). Asudeh *et al.* (2013) discuss this approach to composition in more detail.

In what follows, we will be able to augment these valency frame templates by including appropriate mapping information in them as well. We will also be able to describe various argument alternations

[16] A&G also stipulate various relations between grammatical functions and semantic arguments in such templates, but this is too limiting, and once we have established our theory of mapping, we can do better. As such, I omit reference to mapping from the present examples, since this is tangential to the main point under discussion.

by means of templatic material added to basic lexical entries, thus continuing the project of A&G and Asudeh *et al.* (2013) to reduce the idiosyncratic content of lexical entries as much as possible, and describe cross-categorising generalities using templates.

With all the pieces in place – Kibort's LMT, a connected semantic structure in lieu of argument structure, and the notion of lexical generalisation by template – we now turn to the main proposal of this paper.

6 MAPPING THEORY WITHOUT ARGUMENT STRUCTURE

6.1 *Preliminaries*

I want to suggest that Kibort's arg positions can be equated with the ARG attributes in A&G's connected semantic structures. This will purchase the explanatory power of Kibort's theory but without the cost of a fully fledged argument structure separate from semantic structure. One immediate advantage is that the uniqueness condition on arg positions comes for free, since the functional nature of semantic structures (assuming that they share this property with f-structures) means that there cannot be more than one attribute with the same name.

One implication of merging the proposals in this way, though, is that the subscript numbers on the ARG features at semantic structure now actually have some significance, contra, I suspect, the intention of A&G. In other words, alongside s-structures like (41) for *select*, where there are two arguments labelled ARG_1 and ARG_2, there will also be examples like (42) for *put*, where there are discontinuities in the numberings.

(41)
$$\begin{bmatrix} \text{REL} & \text{select} \\ \text{EVENT} & [\ ] \\ \text{ARG}_1 & [\ ] \\ \text{ARG}_2 & [\ ] \end{bmatrix}$$

(42)
$$\begin{bmatrix} \text{REL} & \text{put} \\ \text{EVENT} & [\ ] \\ \text{ARG}_1 & [\ ] \\ \text{ARG}_2 & [\ ] \\ \text{ARG}_4 & [\ ] \end{bmatrix}$$

Is this a problem? Let us consider A&G's position. In their paper, they evacuate information about thematic roles out of the grammatical architecture by relegating it to the meaning language, and having empty place-holder names for semantic arguments. But without further information, this situation makes a principled theory of mapping impossible: without knowledge of which argument corresponds to which thematic role, *or* which argument corresponds to which grammatical function, we cannot know that 'John loves Mark' means *love(john, mark)*, not *love(mark, john)*, for example. To provide for this, A&G simply stipulate the mappings between GFs and ARG positions. If we want something a little more general, we will need more information. While I share A&G's desire for theoretical parsimony, I think that if they also expect a theory of mapping to provide these mapping equations without something further, they ultimately ask too much. Therefore, one of the two reductions has to be abandoned: either we return thematic role information to the grammar, or we invest the argument names with some meaning.

The first of these reductions, the move to exclude thematic roles from the grammatical architecture, is, I believe, a worthwhile one. Thematic roles are "at best a pretty obscure lot" (as Quine (1956) once said of intensions), beset by multiple theoretical issues. As many have pointed out (e.g. Gawron 1983; Dowty 1991; Ackerman and Moore 2001; Davis 2011), a satisfactory list of roles has never been given. And even when a set of roles is agreed upon, it has not proved possible to find a coherent ranking or hierarchy among them that would apply equally well to all the phenomena for which such hierarchies are adduced (Newmeyer 2002, 65ff.; Levin and Rappaport Hovav 2005, ch. 6; Rappaport Hovav and Levin 2007).

What is more, thematic roles are sometimes thought of as sets of entailments, and it would then certainly seem to make more sense to categorise them as semantic predicates which can take part in such entailments, and which can stand as abbreviations for whatever complex of 'proto-role' properties actually instantiate them (Dowty 1991; Ackerman and Moore 2001). Thus, I believe that A&G's decision to rely on an event semantics which treats thematic roles simply as unanalysed predicates is a sensible one.

But this closes one avenue to a successful mapping theory. Obviously for a verb like *eat* we want, in some sense, to say that the Agent eats the Patient. In the syntax, this corresponds to the fact that whatever is the subject eats whatever is the object. But we cannot now say that the subject is the Agent, and that the object is the Patient, for example, since we would then be combining terms of the linear logic with terms of the meaning language.[17] Of course, the standard Glue formulation, e.g. (43), expresses the relation between thematic roles and GFs directly:

(43) $\lambda y \lambda x \lambda e.\mathit{eat}(e) \wedge \mathit{agent}(e) = x \wedge \mathit{patient}(e) = y :$
$(\uparrow \text{OBJ})_\sigma \multimap (\uparrow \text{SUBJ})_\sigma \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

But this is overly limiting in two ways. Firstly, it fails to account for morphosyntactic alternations such as passive, where the SUBJ corresponds to the Patient, not the Agent. In this case, we would have to have a different meaning constructor for passive *eat*, which seems wrong, since such alternations are supposed not to alter truth-conditional meaning and so should share the same meaning constructor. Secondly, we are faced once again with the problem of optional arguments, since there will not always *be* an OBJ, but there will always be a Patient argument. Both of these motivate understanding meaning composition in terms of semantic arguments rather than grammatical functions directly. This is where the ARG attributes at semantic structure come in.

Given the advantages of avoiding talk of thematic roles in the architecture of the grammar (a point also emphasised by Kibort as a strength of her approach), the alternative is to give up the assumption that the ARG names are devoid of significance. I do not see this as an inherent disadvantage, however. In A&G's approach, these ARG positions are the connection between syntax and semantics, inheriting this role from argument structure. It does not then seem unreasonable to me that they should in some way explain how they bear this connection. It is not enough, for example, that the two arguments of *eat* be *distinct*; we must also know which one is projected from which grammatical function. So now the question arises: What information do these argument positions encode?

[17] Interestingly, this kind of mixing *is* possible in the so-called 'Old Glue' formulation of Glue Semantics (e.g. in Dalrymple 1999), and Dalrymple *et al.* (1993) take advantage of this to implement a mapping theory very close in spirit to A&G's proposal.

For Kibort, the argument slots in her valency frame are *sui generis*; they are what argument structure consists of, and their function is simply to mediate between semantic and syntactic information. To this extent, they do not seem to *mean* anything. But by virtue of their intermediary role, they embody some information from each structure. For example, in traditional LMT, it is noted that Patient-like arguments tend to be $[-r]$. In Kibort's terms, this means that the thematic role of Patient tends to attach to arg_2 – in other words, arg_2 is in some sense associated with Patient-like properties. Similarly, Agents tend to be $[-o]$, which corresponds to arg_1. So while argument structure, under this approach, is itself technically devoid of semantic/thematic information, it still embodies certain relationships involving this information.

Of course, this is no criticism of Kibort; any theory of mapping will have to model such regularities (indeed, in many senses that is what a theory of mapping *is*). But it does suggest one way of seeing such argument roles (pointed out to me by Mary Dalrymple, p.c.): namely, that they can be thought of as embodying macro-level thematic properties. For example, $\text{arg}_2/\text{ARG}_2$ can be seen as grouping together some set of arguments which are 'Patient-like' in whatever way one chooses to elaborate on that concept; but that does not necessarily just mean 'Patients' *per se*. And this can be a source of cross-linguistic variation, much as Butt *et al.* (1997) propose different "intrinsic classifications" for thematic roles in different languages. For example, Goal arguments in English are often ARG_4s – in the unmarked case, they are realised by OBLiques – but in languages with morphological datives, they are often ARG_3s – being realised in the unmarked case by a restricted object, OBJ_{GOAL}.[18]

[18] It might be objected that we have not completely removed mention of thematic roles from the grammatical architecture, since we still have the semantically restricted GFs, which are indexed by thematic role (as illustrated here by OBJ_{GOAL}). However, these indices are really only for f-structure distinctness, and it doesn't especially matter what is used for that purpose. While it is true that 'mainstream' LFG has them indexed by thematic role, we can just as well use numbers, letters, or something else entirely. In fact, in the original formulation of LFG, Kaplan and Bresnan (1982) use the name of a preposition to index OBLiques (e.g. OBL_{WITH}), and we might well extend this to morphological case for restricted objects, so that the example in the text could be rewritten OBJ_{DAT} for dative case.

However, such tendencies must be just that, and nothing more concrete: the key advantage of Kibort's approach is that, like A&G's, it attempts to do without explicit thematic role information, and so any association between argument positions and thematic roles must not be too firm. This allows for what Kibort (2007, 2008) calls semantic participant re-alignment, whereby the same argument slot can have its semantic associations shifted by certain morphological processes (which allows for a better explanation of the patterns of argument-GF linking we observe in these cases). We will see an example of this in Section 6.3.2 below.

6.2 *Formalising Kibort's LMT*

The valency frames which make up Kibort's argument structure are quite esoteric objects. Let us try and formalise them a little more precisely using familiar LFG mechanisms. To clarify matters, we begin by simply rewriting Kibort's valency frame in our own terms as follows:

(44)	<	ARG$_1$	ARG$_2$	ARG$_3$	ARG$_4$	…	ARG$_n$	>
		MINUSO	MINUSR	PLUSO	MINUSO		MINUSO	

Kibort imposes no upper limit on the number of argument positions a verb can select, motivated by the fact that there are very many argument-adding operations such as the applicative, benefactive, causative, etc. However, we can draw a distinction, following Needham and Toivonen (2011), between *core* and *derived* arguments. Core arguments are those which are intrinsic to a verb's meaning, such as the two arguments of *devour*: a devouring event is inherently a binary relation, between the devourer and the devourum. This is in contrast to derived arguments, which can be optionally added to certain classes of verb. These include Instruments, Beneficiaries, and Experiencers, as in (45)–(47):

(45) Saint George slew the dragon **with a lance**.

(46) Kim drew a picture **for his sister**.

(47) It seems **to me** as if you don't know the answer.

Reasons of space preclude a detailed analysis of the differences between core and derived arguments here (see Needham and Toivonen 2011, especially pp. 408–413, for more), but what is interesting to note for our purposes is that, at least in English, derived arguments are often introduced by prepositions, and therefore surface as OBLs. Notably, this corresponds to the fact that all arg positions from arg_4 and above in Kibort's valency frame are marked $[-o]$, the feature which in the unmarked case will surface as an OBL (assuming that there is usually a higher arg position which will be realised as the SUBJ). With this in mind, I propose to associate all argument slots higher than arg_4 with derived arguments. The application of the mapping theory is then restricted to the core arguments of a predicate, specifically the first four, explicitly numbered slots in Kibort's valency frame. By contrast to core arguments, derived arguments will not participate in mapping theory proper, but rather will be introduced lexically/syntactically (see Section 6.3.2 for an example). The new, compact, valency frame is given in (48):

(48) $\left\langle \begin{array}{cccc} \text{ARG}_1 & \text{ARG}_2 & \text{ARG}_3 & \text{ARG}_4 \\ \text{MINUSO} & \text{MINUSR} & \text{PLUSO} & \text{MINUSO} \end{array} \right\rangle$

We now turn to the question of how to represent the default mapping principles in terms of the formal apparatus of LFG. Firstly, we need to associate each ARG value with its respective pair of GFs; secondly, we need to ensure that this mapping is optional, since it is always possible not to represent an argument syntactically (if it is encoded in some other way, as in e.g. the short passive, or the optional Patient arguments of *eat* and *drink*). The first task we can accomplish using a defining equation like the one in (49), for ARG_2:

(49) $(\uparrow \text{MINUSR})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

Using the feature decomposition/disjunction introduced earlier, this states that the σ-projection of either the SUBJ or the OBJ maps to ARG_2. Translating all of Kibort's intrinsic assignments into this format, we have the following:

(50) a. $(\uparrow \text{MINUSO})_\sigma = (\uparrow_\sigma \text{ARG}_1)$
b. $(\uparrow \text{MINUSR})_\sigma = (\uparrow_\sigma \text{ARG}_2)$
c. $(\uparrow \text{PLUSO})_\sigma = (\uparrow_\sigma \text{ARG}_3)$
d. $(\uparrow \text{MINUSO})_\sigma = (\uparrow_\sigma \text{ARG}_4)$

For the sake of brevity/clarity, mapping information like this can be captured in a template, MAP (cf. Asudeh *et al.* 2014, 76):

(51) $\textsc{Map}(D, A) :=$
$(\uparrow D)_\sigma = (\uparrow_\sigma A)$

Map(D, A) generates the appropriate functional description to map the feature decomposition D to the argument A. So, for example, a call of Map(minusr, $\textsc{arg}_2$) means that one of the gfs in minusr will map to $\textsc{arg}_2$. Thus, the generalisations in (50) can be captured more perspicuously as follows:

(52) a. Map(minuso, $\textsc{arg}_1$)
b. Map(minusr, $\textsc{arg}_2$)
c. Map(pluso, $\textsc{arg}_3$)
d. Map(minuso, $\textsc{arg}_4$)

This format also allows for lexical items to contain additional mapping entries, which augment the defaults in some way. For example, the passive rule discussed in Section 3 can be represented as (53):

(53) Map(plusr, $\textsc{arg}_1$)

This is equivalent to adding $[+r]$ to the specification of arg_1 in Kibort's theory. We will return to how the passive is implemented in Section 6.3.1 below.

The second desideratum, optionality, is a little more complicated. One suggestion might be to simply make use of the regular language of LFG's functional descriptions and indicate optionality by surrounding the expression in parentheses:

(54) $((\uparrow \textsc{minusr})_\sigma = (\uparrow_\sigma \textsc{arg}_2))$

If we only required one such mapping equation per argument, this would be perfectly acceptable: the resource sensitivity of Glue Semantics would ensure that, unless something else provided the requisite mapping information or alleviated the need for a particular resource to be syntactically realised (such as the predicate being in the passive), these mapping equations would be selected.

However, one of the strengths of Kibort's approach is that morphosyntactic argument alternations can be explained in terms of additional constraints placed on particular argument positions (such as the passive, as discussed above). This means we need to be able to add extra mapping equations. But we do not want the optionality of each mapping equation to be independent: when we say that the highest argument of a verb is $[-o]$ *and* $[+r]$ we do not mean that it can be just $[-o]$ *or* $[+r]$; if the argument is realised syntactically, it must meet *both* feature restrictions. In other words, for a particular argument, a verb must call *all* or *none* of the relevant mapping equations, not something in between. One way to enforce this is to use a disjunction:

(55) $\{@\text{MAP}(\text{MINUSO}, \text{ARG}_1) | (\uparrow_\sigma \text{ ARG}_1)_{\sigma^{-1}} = \varnothing\}$

The second disjunct says that nothing maps to $(\uparrow_\sigma \text{ ARG}_1)$. It does this by stating that the inverse of the σ-function applied to $(\uparrow_\sigma \text{ ARG}_1)$, which names the f-structure(s) which map(s), via σ, to $(\uparrow_\sigma \text{ ARG}_1)$, returns the empty set. In other words, there is no f-structure which maps to ARG_1. Thus, the whole expression in (55) says that *either* a MINUSO GF maps to ARG_1, *or* nothing does.[19]

Now, consider the situation where we have two expressions of this form:

(56) $\{@\text{MAP}(\text{MINUSO}, \text{ARG}_1) | (\uparrow_\sigma \text{ ARG}_1)_{\sigma^{-1}} = \varnothing\}$
$\{@\text{MAP}(\text{PLUSR}, \text{ARG}_1) | (\uparrow_\sigma \text{ ARG}_1)_{\sigma^{-1}} = \varnothing\}$

In this situation, if one of these disjunctions resolves to the MAP template, then the other must as well: any call of MAP which mentions

[19] I am assuming that being mentioned in a meaning constructor is sufficient for an attribute to appear at semantic structure, even in the cases where nothing explicitly maps to it. This seems to be the implication of e.g. Dalrymple's (2001, 250–253) analysis of common nouns, where the attributes VAR and RESTR appear in the semantic structure of the noun, even though nothing explicitly introduces them. If this is not the case, it may be necessary to add an extra equation to the right-hand disjunct of (55) to state that ARG_1 has *some* value, even if nothing provides it lexically. The expression in (i) might be one way of doing this (suggested to me by Mary Dalrymple, p.c.):

(i) $(\uparrow_\sigma \text{ ARG}_1) = \%\text{A}$

This introduces a local variable (Crouch *et al.* 2012) but gives no further information about it. It is intended to be interpreted as meaning "my ARG_1 has *some* value, but it doesn't matter what". The question of exactly when material appears at s-structure would seem to be an open one, to which further attention must be paid.

ARG_1 is incompatible with a constraint which states that nothing maps to ARG_1. This means that if we select the first disjunct of any of these expressions, we cannot select the second disjunct for any other expression mentioning the same ARG position. This describes exactly the situation we wanted to model: either all the mapping equations relating to a certain argument are chosen, or none are.

We are now in a position to fully encode the default mapping information for each argument position. We abbreviate them in templates, as below; for readability, we also abbreviate the second disjunct which prohibits mapping in a further template:

(57) NoMap(A) :=
$$(\uparrow_\sigma \text{ A})_{\sigma^{-1}} = \varnothing$$

(58) a. Arg1 :=
$$\{@\text{Map}(\text{minuso}, \text{arg}_1) | @\text{NoMap}(\text{arg}_1)\}$$

b. Arg2 :=
$$\{@\text{Map}(\text{minusr}, \text{arg}_2) | @\text{NoMap}(\text{arg}_2)\}$$

c. Arg3 :=
$$\{@\text{Map}(\text{pluso}, \text{arg}_3) | @\text{NoMap}(\text{arg}_3)\}$$

d. Arg4 :=
$$\{@\text{Map}(\text{minuso}, \text{arg}_4) | @\text{NoMap}(\text{arg}_4)\}$$

With this in place, we can now augment any valency templates, such as Agent-Patient-Verb, with the appropriate argument selection templates:

(59) Agent-Patient-Verb :=
@Arg1
@Arg2

$$\lambda P \lambda y \lambda x \lambda e. P(e) \wedge \textit{agent}(e) = x \wedge \textit{patient}(e) = y :$$
$$[(\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma] \multimap$$
$$(\uparrow_\sigma \text{ ARG}_2) \multimap (\uparrow_\sigma \text{ ARG}_1) \multimap (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$$

Which arguments a verb selects is determined by what valency template it calls, which, in turn, is constrained by the lexical semantics of the verb. Recall that the core component of a verbal lexical entry includes a predicate which characterises the event it describes; this specification can impose restrictions on what kinds of thematic roles make sense. For example, an intransitive verb like *yawn* could not call the AGENT-PATIENT-VERB template because the nature of a yawning event is such that there can only be one entity involved.[20] We will not discuss the exact nature of such entailments here, since this would take us too far afield into the realm of lexical semantics, but see Dowty (1991), Primus (1999), and Ackerman and Moore (2001) for some discussion.

One final piece of the puzzle is missing. Each call of the MAP template introduces a disjunction: it specifies that one of a pair of GFs is mapped to the relevant ARG position. The question now facing us is how to resolve these disjunctions.

The final instantiation of the mapping equations with particular grammatical functions will be achieved based on the ranking of the ARGs and the GFs, and crucially not by reference to any thematic hierarchy. The arguments are ordered as in Kibort's valency frame, i.e. by their subscript numbers. In other words, the following is true:

(60) ARG_m is higher than ARG_n if and only if $m < n$

We also continue to assume the partial ordering on the GFs given in (15), and repeated here:

(61) $\text{SUBJ} > \text{OBJ}, \text{OBL}_\theta > \text{OBJ}_\theta$

With this in place, the mapping procedure is the same as in Kibort's theory: the highest arguments are linked with the least marked GFs. I leave open the question of how exactly this should be implemented formally: for instance, it could make use of an Optimality-Theoretic framework (in the vein of e.g. Asudeh 2001), or of the similar but distinct approach outlined in Butt *et al.* (1997).

[20] Cognate objects, as in *She yawned a big yawn*, may be thought to pose a problem for this statement. However, unlike the understood arguments of e.g. *eat* and *drink*, they are not core arguments of the predicate, and instead behave semantically like adjuncts, adding additional information about the event which the predicate describes (Asudeh *et al.* 2014, 78–80).

To see the theory in action, let us return to the example of *devour*. The lexical entry for *devoured* will look something like (62):[21]

(62) *devoured* V $(\uparrow \textsc{pred}) = \text{`devour'}$
$@\textsc{Past}$
$@\textsc{Agent-Patient-Verb}$

$\lambda e.\mathit{devour}(e) : (\uparrow_\sigma \textsc{event}) \multimap \uparrow_\sigma$

Unpacking the valency template, we obtain (63):

(63) *devoured* V $(\uparrow \textsc{pred}) = \text{`devour'}$
$@\textsc{Past}$

$\{(\uparrow \textsc{minuso})_\sigma = (\uparrow_\sigma \textsc{arg}_1) | @\textsc{NoMap}(\textsc{arg}_1)\}$
$\{(\uparrow \textsc{minusr})_\sigma = (\uparrow_\sigma \textsc{arg}_2) | @\textsc{NoMap}(\textsc{arg}_2)\}$

$\lambda P \lambda y \lambda x \lambda e.P(e) \wedge \mathit{agent}(e) = x \wedge \mathit{patient}(e) = y :$
$[(\uparrow_\sigma \textsc{event}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \textsc{arg}_2) \multimap (\uparrow_\sigma \textsc{arg}_1) \multimap (\uparrow_\sigma \textsc{event}) \multimap \uparrow_\sigma$

$\lambda e.\mathit{devour}(e) : (\uparrow_\sigma \textsc{event}) \multimap \uparrow_\sigma$

Assuming that these arguments are syntactically realised (which, in the absence of some valency-reducing alternation such as the passive, they will have to be), we can extract the following mapping equations from (63), with the disjunctions spelled out in the (b) examples:

(64) a. $(\uparrow \textsc{minuso})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
b. $(\uparrow \{\textsc{subj}|\textsc{obl}_\theta\})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$

(65) a. $(\uparrow \textsc{minusr})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$
b. $(\uparrow \{\textsc{subj}|\textsc{obj}\})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

This gives four possibilities:

[21] The template PAST is just like the template FUTURE, but with appropriate changes:

(i) $\textsc{Past} :=$
$(\uparrow \textsc{tense}) = \textsc{past}$
$\lambda P.\exists e[P(e) \wedge \mathit{past}(e)] : [(\uparrow_\sigma \textsc{event}) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma$

(66) a. $(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
$(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

b. $(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
$(\uparrow \textsc{obj})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

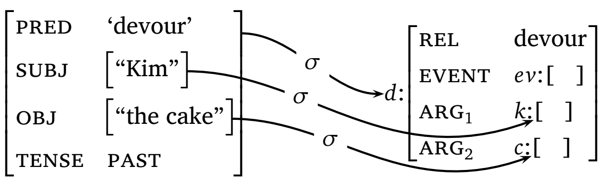
c. $(\uparrow \textsc{obl}_\theta)_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
$(\uparrow \textsc{subj})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

d. $(\uparrow \textsc{obl}_\theta)_\sigma = (\uparrow_\sigma \textsc{arg}_1)$
$(\uparrow \textsc{obj})_\sigma = (\uparrow_\sigma \textsc{arg}_2)$

By appealing to some version of Function-Argument Biuniqueness, we can rule out (66a). If we assume the Subject Condition (that is, that clauses must have subjects), we can also rule out (66d); notice then that the two remaining mappings are the correct ones for the active and passive respectively. However, we do not need to assume the Subject Condition. Following our mapping principles, we simply link the highest argument with the highest GF; however this is achieved formally, (66b) will be the optimal linking, since the highest argument, $\textsc{arg}_1$, is matched with the highest GF, SUBJ. The resulting mapping between f-structure and s-structure is shown in Figure 3.[22]

The meaning constructor for AGENT-PATIENT-VERB in (59) will make $\textsc{arg}_1$ the Agent and $\textsc{arg}_2$ the Patient, as shown in the Glue proof in Figure 4, and so, coupled with the mapping in (66b), we see that the SUBJect is the $\textsc{arg}_1$ which is the devourer, while the OBJect is the $\textsc{arg}_2$ which is the devourum, exactly as desired. [23]

Figure 3: Structures and correspondences for *Kim devoured the cake*



[22] I abbreviate the contents of f-structures for the sake of readability.

[23] In proofs, meaning constructors have been instantiated with respect to the s-structures given in the text.

$$
\dfrac{\begin{array}{l}\textbf{[Past]}\\ \lambda P.\exists e[P(e) \wedge past(e)] :\\ [ev \multimap d] \multimap d\end{array} \qquad \dfrac{\dfrac{\begin{array}{l}\vdots\\ \textbf{[the cake]}\\ \iota z[cake(z)] :\\ c\end{array} \qquad \dfrac{\begin{array}{l}\textbf{[devoured]}\\ \lambda e.devour(e) :\\ ev \multimap d\end{array} \qquad \begin{array}{l}\textbf{[Agent-Patient-Verb]}\\ \lambda P \lambda y \lambda x \lambda e.P(e) \wedge agent(e) = x \wedge patient(e) = y :\\ [ev \multimap d] \multimap c \multimap k \multimap ev \multimap d\end{array}}{\begin{array}{l}\lambda y \lambda x \lambda e.devour(e) \wedge agent(e) = x \wedge patient(e) = y :\\ c \multimap k \multimap ev \multimap d\end{array}}}{\begin{array}{l}\lambda x \lambda e.devour(e) \wedge agent(e) = x \wedge patient(e) = \iota z[cake(z)] :\\ k \multimap ev \multimap d\end{array}} \qquad \begin{array}{l}\textbf{[Kim]}\\ kim :\\ k\end{array}}{\begin{array}{l}\lambda e.devour(e) \wedge agent(e) = kim \wedge patient(e) = \iota z[cake(z)] :\\ ev \multimap d\end{array}}}{\exists e[devour(e) \wedge agent(e) = kim \wedge patient(e) = \iota z[cake(z)] \wedge past(e)] : d}
$$

Figure 4: Proof for *Kim devoured the cake*

6.3 *Argument alternations*

The test of any mapping theory is how well it handles argument alternations. In this section, I demonstrate how the current theory handles two such processes, namely the passive and the benefactive. See Asudeh *et al.* (2014) for an example of how it can handle cognate objects, and Lowe (2015) for a compatible analysis of causatives and other complex predicates.[24]

6.3.1 The passive

As mentioned above, the passive involves restricting the mapping for the highest argument, ARG_1, so that it can appear only as an OBL_θ, if it is realised syntactically at all. We can encode this and the remaining information in a template, after A&G (p. 79):

(67) PASSIVE :=

$$(\uparrow \text{VOICE}) = \text{PASSIVE}$$
$$\{@\text{MAP}(\text{PLUSR}, \text{ARG}_1) | @\text{NOMAP}(\text{ARG}_1)\}$$
$$(\lambda P.\exists x[P(x)] : [(\uparrow_\sigma \text{ARG}_1) \multimap \uparrow_\sigma] \multimap \uparrow_\sigma)$$

The first line supplies the relevant voice information for f-structure. The second line restricts ARG_1 in the appropriate way (as we will see below in more detail). The third line contributes an optional meaning constructor which existentially closes a dependency on the meaning of ARG_1; this will be selected in the short passive but left unused in the *by*-passive (see A&G pp. 75–76 for more detailed discussion).

The lexical entry for passive *devoured* is given in (68):

(68) *devoured* V

$$(\uparrow \text{PRED}) = \text{'devour'}$$
$$@\text{PASSIVE}$$
$$@\text{AGENT-PATIENT-VERB}$$
$$\lambda e.\mathit{devour}(e) : (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$$

Extracting the mapping information from the two templates, we have the following information:

[24] Lowe (2015) adopts the proposals of A&G on which this paper builds, and his analysis is wholly compatible with the elaboration of that framework presented here.

(69) a. {@MAP(MINUSO, ARG_1)|@NOMAP(ARG_1)}

b. {@MAP(PLUSR, ARG_1)|@NOMAP(ARG_1)}

(70) {@MAP(MINUSR, ARG_2)|@NOMAP(ARG_2)}

Assuming both arguments are syntactically realised, we have the following mapping equations:

(71) a. $(\uparrow \{\text{SUBJ}|\text{OBL}_\theta\}) = (\uparrow_\sigma \text{ARG}_1)$

b. $(\uparrow \{\text{OBL}_\theta|\text{OBJ}_\theta\}) = (\uparrow_\sigma \text{ARG}_1)$

(72) $(\uparrow \{\text{SUBJ}|\text{OBJ}\}) = (\uparrow_\sigma \text{ARG}_2)$

The only way to resolve the ARG_1 mapping disjunctions without contradiction is for the argument to be realised as an OBL_θ. This gives us only two options for the mapping:

(73) a. $(\uparrow \text{OBL}_\theta)_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{SUBJ})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

b. $(\uparrow \text{OBL}_\theta)_\sigma = (\uparrow_\sigma \text{ARG}_1)$
$(\uparrow \text{OBJ})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

Since SUBJ $>$ OBJ on our GF hierarchy, the optimal mapping is (73a), as we require. This is shown in Figure 5.

Notice that regardless of whether ARG_1 is syntactically realised or not, the optimal mapping for ARG_2 will always, correctly, be from the SUBJ.

Assuming that passive *by* is semantically vacuous, the proof for *The cake was devoured by Kim* is identical to that given in Figure 4 (except that the **[PAST]** meaning constructor is provided by the auxiliary *was*).

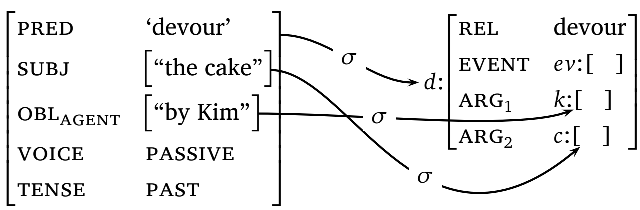


Figure 5: Structures and correspondences for *The cake was devoured by Kim*

6.3.2 The benefactive

Certain verbs in English, like *draw* or *cook*, have lexical alternants which take a core Beneficiary argument:

(74) Alicia drew New York City.

(75) Alicia drew Harry New York City.

We can treat this as zero-marked benefactive morphology, where the morphology introduces the information given in the BENEFACTIVE template below:[25]

(76) BENEFACTIVE :=
@ARG3

$\lambda x \lambda P \lambda y \lambda e.P(y)(e) \wedge \textit{beneficiary}(e) = x :$
$(\uparrow_\sigma \text{ ARG}_2) \multimap$
$[(\uparrow_\sigma \text{ ARG}_2) \multimap (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ ARG}_3) \multimap (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$

As per the discussion of benefactives in Kibort (2007), this adds a new ARG_3 argument to the verb's valency. In addition, the meaning constructor in (76) operationalises Kibort's notion of semantic participant re-alignment (Kibort 2007, 2008), as we will see below.

The lexical entry for regular transitive *drew* is given in (77):

(77) *drew* V $(\uparrow \text{ PRED}) = \text{'draw'}$
@PAST
@AGENT-REPRESENTED-VERB

$\lambda e.\textit{draw}(e) : (\uparrow_\sigma \text{ EVENT}) \multimap \uparrow_\sigma$

[25] Asudeh *et al.* (2014, 81) introduce the benefactive meaning constructor via an annotated c-structure rule, but Müller (2016) has pointed out various shortcomings facing such an account, including problems with coordination. In the text, I treat it as lexically introduced, thus avoiding these issues. Asudeh (2013) also uses the meaning constructor in (76), as well as the one in (83), below, to encode the requirement of animacy on the subject of the main clause; I omit this in order to simplify the analysis, but it could easily be reinstated.

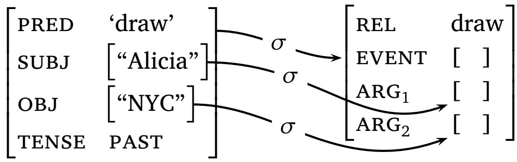


Figure 6:
Structures and correspondences for *Alicia drew New York City*

(78) AGENT-REPRESENTED-VERB :=
@ARG1
@ARG2

$\lambda P \lambda y \lambda x \lambda e.P(e) \wedge \mathit{agent}(e) = x \wedge \mathit{represented}(e) = y$:
$[(\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma] \multimap$
$(\uparrow_\sigma \text{ARG}_2) \multimap (\uparrow_\sigma \text{ARG}_1) \multimap (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

The mapping proceeds exactly as for *devoured*, and indeed as it would for any simple transitive verb which takes an ARG_1 and an ARG_2. We therefore obtain the structures and correspondences in Figure 6 for a sentence like *Alicia drew New York City*.

The lexical entry for benefactive *drew* is just as in (77), but with the addition of the BENEFACTIVE template:

(79) *drew* V $(\uparrow \text{PRED}) =$ 'draw'
@PAST
@BENEFACTIVE
@AGENT-REPRESENTED-VERB

$\lambda e.\mathit{draw}(e) : (\uparrow_\sigma \text{EVENT}) \multimap \uparrow_\sigma$

There are now three arguments to be mapped. Since all of them will have to be realised syntactically, we have the following three mapping equations:

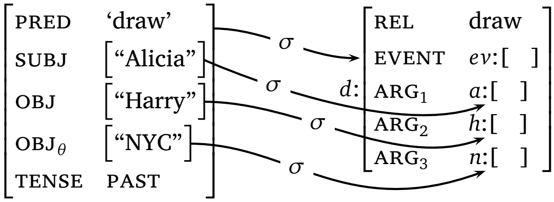
(80) $(\uparrow \text{MINUSO})_\sigma = (\uparrow_\sigma \text{ARG}_1)$

(81) $(\uparrow \text{MINUSR})_\sigma = (\uparrow_\sigma \text{ARG}_2)$

(82) $(\uparrow \text{PLUSO})_\sigma = (\uparrow_\sigma \text{ARG}_3)$

I will not list all the possibilities, but we can describe impressionistically how the mapping is determined. Firstly, the highest argument, ARG_1, is linked with the highest available MINUSO GF, namely the SUBJ. Secondly, the next highest argument, ARG_2, is linked with the highest available MINUSR GF, which is now the OBJ. Thirdly, and finally, ARG_3 is linked with the highest available PLUSO GF; since the

Figure 7:
Structures and correspondences for
Alicia drew Harry New York City



direct OBJ position has been taken, this is OBJ_θ. The mapping is thus as shown in Figure 7.

Notice that the OBJ/ARG_2 no longer corresponds to the drawn entity, but rather to the Beneficiary. This is what Kibort (2007, 2008) refers to as semantic participant re-alignment: in other words, the semantic role of a particular argument position has changed. We achieve this in Glue with the meaning constructor introduced by the BENEFACTIVE template. This specifies that the ARG_2 is the Beneficiary, and then modifies the main verbal meaning so that ARG_3 rather than ARG_2 is passed to it in the position of the Represented argument. This is shown in the Glue proof in Figure 8.

Lexical alternation is not the only way that English can introduce a Beneficiary argument. It can also do so syntactically, using the preposition *for*. In this case, the Beneficiary is a derived argument, and so there is no argument alternation, strictly speaking. This is evidenced in the fact that the basic mapping for the Agent and Represented arguments does not change.

The lexical entry for beneficiary-*for* is given in (83) (after Asudeh 2013):

(83) *for* P $(\uparrow \text{PRED}) = \text{'for'}$
$(\uparrow \text{OBJ})_\sigma = ((\text{OBL} \uparrow)_\sigma \ \text{BENEFICIARY})$

$\lambda x \lambda P \lambda e.P(e) \wedge \textit{beneficiary}(e) = x :$
$(\uparrow \text{OBJ})_\sigma \multimap$
$[((\text{OBL} \uparrow)_\sigma \ \text{EVENT}) \multimap (\text{OBL} \uparrow)_\sigma] \multimap$
$((\text{OBL} \uparrow)_\sigma \ \text{EVENT}) \multimap (\text{OBL} \uparrow)_\sigma$

This does several things. In the second line, using an inside-out expression (Halvorsen and Kaplan 1988; Dalrymple 2001, 143–146), it maps its object to a new, idiosyncratically named argument position, BENEFICIARY, in the main clause's semantic structure. Since derived arguments do not take part in LMT proper, the attribute names ARG_1–ARG_4 are reserved for core arguments, and derived arguments

$$\frac{\dfrac{\textbf{[drew]}\quad \lambda e.draw(e) : ev \multimap d \qquad \textbf{[AGENT-REPRESENTED-VERB]}\quad \lambda P \lambda y \lambda x \lambda e.P(e) \wedge agent(e) = x \wedge represented(e) = y : [ev \multimap d] \multimap h \multimap a \multimap ev \multimap d}{\lambda y \lambda x \lambda e.draw(e) \wedge agent(e) = x \wedge represented(e) = y : h \multimap a \multimap ev \multimap d}\qquad [u : h]^1}{\lambda x \lambda e.draw(e) \wedge agent(e) = x \wedge represented(e) = u : a \multimap ev \multimap d}$$

$$\frac{\lambda x \lambda e.draw(e) \wedge agent(e) = x \wedge represented(e) = u : a \multimap ev \multimap d \qquad \textbf{[Alicia]}\quad alicia : a}{\lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = u : ev \multimap d}$$

$$\frac{\lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = u : ev \multimap d}{\lambda u \lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = u : h \multimap ev \multimap d}\multimap_{\mathscr{I},1}$$

$$\frac{\textbf{[Harry]}\quad harry : h \qquad \textbf{[BENEFACTIVE]}\quad \lambda x \lambda P \lambda y \lambda e.P(y)(e) \wedge beneficiary(e) = x : h \multimap [h \multimap ev \multimap d] \multimap n \multimap ev \multimap d}{\lambda P \lambda y \lambda e.P(y)(e) \wedge beneficiary(e) = harry : [h \multimap ev \multimap d] \multimap n \multimap ev \multimap d}$$

$$\frac{\lambda P \lambda y \lambda e.P(y)(e) \wedge beneficiary(e) = harry : [h \multimap ev \multimap d] \multimap n \multimap ev \multimap d \qquad \lambda u \lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = u : h \multimap ev \multimap d}{\lambda y \lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = y \wedge beneficiary(e) = harry : n \multimap ev \multimap d}$$

$$\frac{\lambda y \lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = y \wedge beneficiary(e) = harry : n \multimap ev \multimap d \qquad \textbf{[NYC]}\quad NYC : n}{\lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC \wedge beneficiary(e) = harry : ev \multimap d}$$

$$\frac{\textbf{[PAST]}\quad \lambda P.\exists e[P(e) \wedge past(e)] : [ev \multimap d] \multimap d \qquad \lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC \wedge beneficiary(e) = harry : ev \multimap d}{\exists e[draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC \wedge beneficiary(e) = harry \wedge past(e)] : d}$$

Figure 8: Proof for *Alicia drew Harry New York City*

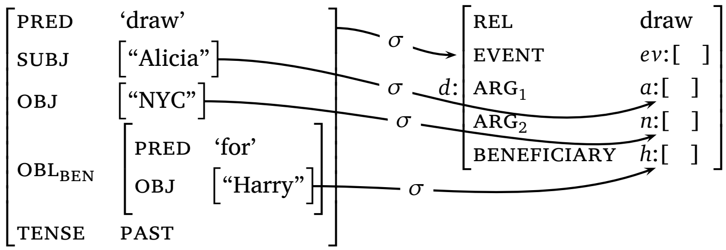


Figure 9: Structures and correspondences for *Alicia drew New York City for Harry*

are instead given mnemonic names at s-structure for (a) distinctiveness and (b) uniqueness, the latter being enforced by the functional nature of s-structure, as discussed above. The mappings for *Alicia drew New York City for Harry* are shown in Figure 9 (the ARG_1 and ARG_2 mappings will proceed as discussed above for regular transitive *drew*).

The third line of *for*'s lexical entry is a meaning constructor which introduces the appropriate Beneficiary meaning. Using the lexical entry for simple transitive *drew* given above, the Glue proof in Figure 10 shows this in action.

7 CONCLUSION

A&G's proposal, to do away with argument structure as a separate level of representation, promises major advances in theoretical parsimony, as well as additional explanatory power. Our grammar is ontologically simpler, and we have a whole new connected structure with internal relations that can be exploited in semantic analyses. However, in the absence of a satisfactory theory of the mapping between arguments and grammatical functions, we lose a great deal of the explanatory power that an a-structure-based mapping theory granted us. In this paper, I hope to have shown that such a theory can be developed, and have chosen to base my approach on recent work in LMT by Kibort. One of the things which sets her proposal apart from earlier versions of LMT is that it argues for a separation of thematic role information and argument structure, which makes it eminently compatible with the A&G proposal, since these authors advocate a very similar position. It is surely encouraging that independent strands of research should have converged in this way. By demonstrating that it is possible to formalise Kibort's theory in terms compatible with the approach of

[drew] $\lambda e.draw(e) : ev \multimap d$

[Agent-Represented-Verb] $\lambda P \lambda y \lambda x \lambda e.P(e) \wedge agent(e) = x \wedge represented(e) = y : [ev \multimap d] \multimap n \multimap a \multimap ev \multimap d$

$$\lambda y \lambda x \lambda e.draw(e) \wedge agent(e) = x \wedge represented(e) = y : n \multimap a \multimap ev \multimap d$$

[NYC] $NYC : n$

$$\lambda x \lambda e.draw(e) \wedge agent(e) = x \wedge represented(e) = NYC : a \multimap ev \multimap d$$

[Alicia] $alicia : a$

$$\lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC : ev \multimap d$$

[Harry] $harry : h$

[for] $\lambda x \lambda P \lambda e.P(e) \wedge beneficiary(e) = x : h \multimap [ev \multimap d] \multimap ev \multimap d$

$$\lambda P \lambda e.P(e) \wedge beneficiary(e) = harry : [ev \multimap d] \multimap ev \multimap d$$

$$\lambda e.draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC \wedge beneficiary(e) = harry : ev \multimap d$$

[Past] $\lambda P.\exists e[P(e) \wedge past(e)] : [ev \multimap d] \multimap d$

$$\exists e[draw(e) \wedge agent(e) = alicia \wedge represented(e) = NYC \wedge beneficiary(e) = harry \wedge past(e)] : d$$

Figure 10: Proof for *Alicia drew New York City for Harry*

A&G, I hope to have lent additional support to both proposals, and laid the foundations for further fruitful work which takes advantage of the strengths of both.

REFERENCES

Farrell ACKERMAN and John MOORE (2001), *Proto-properties and grammatical encoding*, CSLI Publications, Stanford, CA.

Farrell ACKERMAN, Gregory T. STUMP, and Gert WEBELHUTH (2011), Lexicalism, periphrasis, and implicative morphology, in Robert D. BORSLEY and Kersti BÖRJARS, editors, *Non-transformational syntax: Formal and explicit models of grammar*, pp. 325–358, Wiley-Blackwell, Oxford, UK.

Alex ALSINA (1996), *The role of argument structure in grammar: Evidence from Romance*, CSLI Publications, Stanford, CA.

Ash ASUDEH (2001), Linking, optionality, and ambiguity in Marathi, in Peter SELLS, editor, *Formal and empirical issues in Optimality-Theoretic syntax*, pp. 257–312, CSLI Publications, Stanford, CA.

Ash ASUDEH (2012), *The logic of pronominal resumption*, Oxford University Press, Oxford, UK.

Ash ASUDEH (2013), Flexible composition and the argument/adjunct distinction in LFG + Glue, paper presented at the LLI Lab, Institute of Cognitive Science, Carleton University.

Ash ASUDEH, Mary DALRYMPLE, and Ida TOIVONEN (2013), Constructions with Lexical Integrity, *Journal of Language Modelling*, 1(1):1–54, http://jlm.ipipan.waw.pl/index.php/JLM/article/view/56/49.

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, http://www.stanford.edu/group/cslipublications/cslipublications/LFG/17/papers/lfg12asudehgiorgolo.pdf.

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, http://web.stanford.edu/group/cslipublications/cslipublications/LFG/19/papers/lfg14asudehetal.pdf.

Ash ASUDEH and Ida TOIVONEN (2015), Lexical-Functional Grammar, in Bernd HEINE and Heiko NARROG, editors, *The Oxford Handbook of Linguistic Analysis* (2nd edn.), pp. 373–406, Oxford University Press, Oxford, UK, http://www.oxfordhandbooks.com/view/10.1093/oxfordhb/9780199544004.001.0001/oxfordhb-9780199544004-e-017.

Joan BRESNAN (1980), Polyadicity, in Teun HOEKSTRA, Harry VAN DER HULST, and Michael MOORTGAT, editors, *Lexical Grammar*, pp. 97–121, Foris., Dordrecht, NL, in revised form in Joan Bresnan (ed.) 1982. *The mental representation of grammatical relations*, 149–172. Cambridge, MA: MIT Press.

Joan BRESNAN (1982), The passive in lexical theory, in Joan BRESNAN, editor, *The mental representation of grammatical relations*, pp. 3–86, MIT Press, Cambridge, MA.

Joan BRESNAN (1990), Levels of representation in locative inversion: A comparison of English and Chicheŵa, manuscript, Stanford University and Xerox Palo Alto Research Center.

Joan BRESNAN (2001), *Lexical-functional syntax*, Blackwell, Oxford, UK.

Joan BRESNAN, Ash ASUDEH, Ida TOIVONEN, and Stephen WECHSLER (2016), *Lexical-functional syntax* (2nd edn.), Wiley-Blackwell, Oxford, UK.

Joan BRESNAN and Jonni M. KANERVA (1989), Locative inversion in Chichewa: A case study of factorization in grammar, *Linguistic Inquiry*, 20(1):1–50.

Miriam BUTT (1995), *The structure of complex predicates in Urdu*, CSLI Publications, Stanford, CA.

Miriam BUTT, Mary DALRYMPLE, and Anette FRANK (1997), An architecture for linking theory in LFG, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG97 Conference*, CSLI Publications, Stanford, CA, http://web.stanford.edu/group/cslipublications/cslipublications/LFG/LFG2-1997/lfg97butt-dalrymple-frank.pdf.

Dick CROUCH, Mary DALRYMPLE, Ron KAPLAN, Tracy KING, John MAXWELL, and Paula NEWMAN (2012), *XLE documentation*, Palo Alto Research Center (PARC), Palo Alto, CA., http://www2.parc.com/isl/groups/nltt/xle/doc/xle.html.

Mary DALRYMPLE, editor (1999), *Semantics and syntax in Lexical Functional Grammar: The resource logic approach*, MIT Press, Cambridge, MA.

Mary DALRYMPLE (2001), *Lexical Functional Grammar*, number 34 in Syntax and Semantics, Academic Press, Stanford, CA.

Mary DALRYMPLE, Angie HINRICHS, John LAMPING, and Vijay SARASWAT (1993), The resource logic of complex predicate interpretation, in *Proceedings of the 1993 Republic of China Computational Linguistics Conference (ROCLING)*, Hsitou National Park, Taiwan, http://www.aclclp.org.tw/rocling/1993/K01.pdf.

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, http://www.stanford.edu/group/cslipublications/cslipublications/LFG/9/lfg04dkk.pdf.

Anthony R. DAVIS (2011), Thematic roles, in Claudia MAIENBORN, Klaus VON HEUSINGER, and Paul H. PORTNER, editors, *Semantics: an international handbook of natural language meaning*, volume 1, pp. 399–420, Mouton de Gruyter, Berlin, DE.

David DOWTY (1991), Thematic proto-roles and argument selection, *Language*, 67(3):547–619.

Yehuda N. FALK (2001), *Lexical-Functional Grammar: An introduction to parallel constraint-based syntax*, CSLI Publications, Stanford, CA.

Dorothee FEHRMANN, Uwe JUNGHANNS, and Denisa LENERTOVÁ (2010), Two reflexive markers in Slavic, *Russian Linguistics*, 34(3):203–238.

Jean Mark GAWRON (1983), Lexical representations and the semantics of complementation, doctoral dissertation, University of California, Berkeley.

Per-Kristian HALVORSEN and Ronald M. KAPLAN (1988), Projections and semantic descriptions in Lexical-Functional Grammar, in *Proceedings of the International Conference on Fifth Generation Computer Systems*, pp. 1116–1122, Institute for New Generation Systems.

Nikolaus P. HIMMELMANN (2002), Voice in western Austronesian: an update, in Fay WOUK and Malcolm ROSS, editors, *The history and typology of western Austronesian voice systems*, pp. 7–16, Pacific Linguistics, Canberra, AU.

Ray JACKENDOFF (1990), *Semantic structures*, The MIT Press, Cambridge, MA.

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.

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, `http://www.stanford.edu/group/cslipublications/cslipublications/LFG/6/lfg01kibort.pdf`.

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, `http://www.stanford.edu/group/cslipublications/cslipublications/LFG/12/papers/lfg07kibort.pdf`.

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, `http://www.stanford.edu/group/cslipublications/cslipublications/LFG/13/papers/lfg08kibort.pdf`.

Anna KIBORT (2013), Objects and Lexical Mapping Theory [Abstract], in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG13 Conference*, CSLI Publications.

Anna KIBORT (2014), Mapping out a construction invetory with (Lexical) Mapping Theory, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG14 Conference*, pp. 262–282, CSLI Publications, Stanford, CA, `http://web.stanford.edu/group/cslipublications/cslipublications/LFG/19/papers/lfg14kibort.pdf`.

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*, CSLI Publications, Stanford, CA.

Beth LEVIN and Malka RAPPAPORT HOVAV (2005), *Argument realization*, (Research Surveys in Linguistics Series), Cambridge University Press, Cambridge, UK.

John J. LOWE (2015), Complex predicates: an LFG + glue analysis, *Journal of Language Modelling*, 3(2):413–462.

Stefan MÜLLER (2016), Flexible phrasal constructions, constituent structure and (cross-linguistic) generalizations: a discussion of template-based phrasal LFG approaches, in Doug ARNOLD, Miriam BUTT, Berthold CRYSMANN, Tracy Holloway KING, and Stefan MÜLLER, editors, *Proceedings of the Joint 2016 Conference on Head-driven Phrase Structure Grammar and Lexical Functional Grammar*, pp. 457–477, CSLI Publications, Stanford, CA, `http://web.stanford.edu/group/cslipublications/cslipublications/HPSG/2016/headlex2016-mueller.pdf`.

Stephanie NEEDHAM and Ida TOIVONEN (2011), Derived arguments, in Miriam BUTT and Tracy Holloway KING, editors, *Proceedings of the LFG11 Conference*, pp. 401–421, `http://www.stanford.edu/group/cslipublications/cslipublications/LFG/16/papers/lfg11needhamtoivonen.pdf`.

Frederick J. NEWMEYER (2002), Optimality and functionality: A critique of functionally-based Optimality-Theoretic syntax, *Natural Language and Linguistic Theory*, 20(1):43–80.

Terence PARSONS (1990), *Events in the semantics of English*, MIT Press, Cambridge, MA.

Beatrice PRIMUS (1999), *Cases and thematic roles: ergative, accusative and active*, Niemeyer, Tübingen, DE.

Willard V. QUINE (1956), Quantifiers and propositional attitudes, *The Journal of Philosophy*, 53(5):177–187.

Malka RAPPAPORT HOVAV and Beth LEVIN (2007), Deconstructing thematic hierarchies, in Annie ZAENEN, Jane SIMPSON, Tracy Holloway KING, Jane

GRIMSHAW, Joan MALING, and Chris MANNING, editors, *Architecture, rules, and preferences: Variations on themes by Joan W. Bresnan*, pp. 385–402, CSLI Publications, Stanford, CA.

This work is licensed under the Creative Commons Attribution 3.0 Unported License.
http://creativecommons.org/licenses/by/3.0/