Multiword expressions and lexicalism*

Jamie Y. Findlay
jamie.findlay@ling-phil.ox.ac.uk

SE-LFG22
SOAS
4 February 2017

1 Introduction

- Lexicalist theories like LFG tend to adhere to what Ackerman et al. (2011:326) call the *Principle of unary expression* (PUE):

  (1) **Principle of unary expression:** In syntax, a lexeme is uniformly expressed as a single morphophonologically integrated and syntactically atomic word form.

- But this is problematic for e.g. periphrasis, and the authors argue for an abandonment of PUE in order to better keep to other aspects of lexicalism (avoiding the formation of complex predicates in the syntax rather than the lexicon, for example).

- I present further evidence against PUE from idioms, and show that this data is actually more problematic for our theory of the syntax-lexicon interface than periphrasis:
  - Like periphrasis, a single lexical item be realised as multiple, syntactically independent word forms.
  - But unlike periphrasis, they can also appear arbitrarily far apart, and in no easily characterisable syntactic configuration.

- Given these data, I propose a change to the LFG architecture, making use of a Tree Adjoining Grammar (TAG: Joshi et al. 1975; Abeillé & Rambow 2000) as the c-structure component.

*I would like to thank many people for their helpful comments and discussions on this topic, including Ash Asudeh, Mary Dalrymple, Timm Lichte, John Lowe, and Stephen Pulman. I would especially like to thank Alex Biswas for giving me a clearer insight into the shape of the problem.

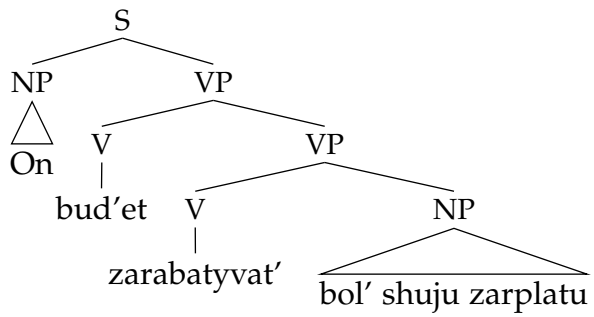
2 Explananda

2.1 Periphrasis

- Ackerman et al. (2011) suggest that there is nothing problematic about the morphology producing multiple word forms as the exponent of some cells in a lexeme's paradigm.
- Question: how does this interact with the syntax?

> The syntactic configuration of morphologically defined periphrases [...] is defined by rules that we here assume to be language-specific; in the case of any periphrase [Y Z] defined by the implicative rule [for the Russian imperfective future], we assume that the verb form Y is the head of [Y Z] and that the verb form Z appears in c-structure as the head of Y's complement, as in [(2)]: (Ackerman et al. 2011:339)

(2)



'He will be earning a big salary.'

- Perhaps OK for locally-defined structures like this (although there is still the question of cashing this out formally), but idioms are more complicated.

2.2 Idioms

- Idioms are non-compositional in the sense that their meanings are not a function of the literal meaning of their parts and the way they are put together.
- Their meanings therefore have to be learned, and oftentimes seem to be just as arbitrary as any given lexical entry (e.g. *kick the bucket*, *look a gift horse in the mouth*, *shoot the breeze*).[1]
- If this is the case, then they are good contenders for being stored in the lexicon.
- But of course, just like periphrases, they are 'spelt out' as multiple word forms, which appear independently in the syntax (and can be separated, modified, and inflect individually).
- Unlike in the case of periphrasis, however, idioms are inherently multiword expressions (MWEs). There are no(?) languages where *all* cells of a paradigm are periphrastic, whereas *every* realisation of an idiom will consist of multiple word forms.

[1] Although all of these may have perfectly logical meanings based on their histories, it must be admitted that for most speakers they are synchronically opaque.

- Even if we are prepared to accept a weakening of PUE in the case of periphrasis, then, we might still resist it in the case of idioms.
- What's more, even if we are quite happy with abandoning PUE entirely, the simple analysis of the interface between c-structure and the lexicon suggested by Ackerman et al. is inadequate for idioms.
- Some idioms share the limited syntactic flexibility of periphrases:

  (3) a. Old Man Mose kicked the bucket.
  b. # The bucket was kicked (by Old Man Mose).
  c. # Which bucket did Old Man Mose kick?
  d. # The bucket that Old Man Mose kicked was {sudden/sad/...}.

- But this is a matter of degree, with other idioms showing a considerable amount of flexibility:

  (4) a. He pulled strings to get me assigned to his command.
  b. Strings were pulled to get me assigned to his command.
  c. Which strings did he pull to get you assigned to his command?
  d. The strings that he pulled got me assigned to his command.

- Section 3: arguments against a PUE-retaining analysis of idioms.
- Section 4: a more powerful c-structure to account for idiom flexibility.

3 The lexical ambiguity approach

- The natural response to idioms in a lexicalist theory is what we might call the *lexical ambiguity approach* (LA).
- In such an approach, idioms are treated as made up of special versions of the words they contain, which combine to give the appropriate meaning for the whole expression.
- Words like *pull* and *strings* become ambiguous, meaning either **pull**′and **strings**′ or **exploit**′and **connections**′.
- This sidesteps the PUE issue, since idioms are not single lexical items, but rather collections of separate lexical items which conspire to create the overall meaning.
- Examples: Sailer (2000) in HPSG, Kay et al. (2015) in SBCG, Lichte & Kallmeyer (2016) in LTAG.
- LA is particularly well suited to explaining so-called decomposable idioms, where the meaning of the whole can be distributed across the parts.

- Since the idiom meaning is assigned to the individual words in LA, this immediately explains the fact that these idioms can be separated by syntactic operations (e.g. *the cat has been let out of the bag*, *the strings that Kim pulled…*), or that they are open to internal modification and/or quantification (e.g. *Delhi's politicians pass the polluted buck*, *Maybe by writing this book I'll offend a few people or touch a few nerves*, *Tom won't pull family strings to get himself out of debt*).

3.1 Problems with LA

Non-decomposable idioms

- It is not so clear how such an approach should handle non-decomposable idioms, like *kick the bucket*, *blow off steam*, *shoot the breeze*, etc., where there is no obvious way of breaking down the meaning of the idiom such that its parts correspond to the words that make up the expression.

- One solution: Lichte & Kallmeyer (2016) argue for what they call 'idiomatic mirroring', whereby each of the parts of the idiom contributes the meaning of the whole expression, so that *kick* means **die**′, *bucket* means **die**′, and, presumably, *the* means **die**′as well.

- This is possible in the unification-based semantics that they assume (Kallmeyer & Romero 2008), but is dependent on this choice, and otherwise will lead to compositional issues.

- Without idiomatic mirroring, we are forced to assume that only one of the words in the expression bears the meaning, and the rest are semantically inert. For example, perhaps there is a $kick_{id}$ which means **die**′, and selects for special semantically inert forms the_{id} and $bucket_{id}$.

- But the choice of where to locate the meaning is ultimately arbitrary. While it might intuitively seem to make sense to assign it to the verb, since it is the head of the VP which makes up the expression, formally it makes no difference: we may as well have $bucket_{id}$ meaning **die**′, or even the_{id}, provided they select for the other inert forms and then pass their meaning up to the whole VP.

- It also leads to another formal issue: we now face an explosive proliferation of semantically inert forms throughout the lexicon.

- What is more, each of these must be restricted so that it does not appear outside of the appropriate expression. But this means that the the_{id} in *kick the bucket* can't be the same the_{id} as in *shoot the breeze*.

- We need as many *the*s as there are expressions which include it. Instead of having to expand the lexicon by as many entries as there are idioms, we have to expand it by as many entries as there are *words in idioms*.

Processing

- Swinney & Cutler (1979): idioms are processed in the same way as regular compositional expressions; i.e. there is no special 'idiom mode' of comprehension.

- At the same time, these authors and others have found that idiomatic meanings are processed faster and in preference to literal ones (Estill & Kemper 1982; Gibbs 1986; Cronk 1992).
- If both these things are true, then LA is in trouble: in this approach, semantic composition of idioms is exactly the same as of literal expressions. There is no reason to think idioms should be processed any faster; if anything, we might expect them to be slower, since they involve ambiguity by definition.
- But is this a concern about performance, rather than competence?

Restricting the distribution of idiom words

- If *pull* can mean **exploit**′ and *strings* can mean **connections**′, we clearly have to prevent them occurring apart from one another:

  (5) # You shouldn't pull his good nature.

  (6) # Peter was impressed by Claudia's many strings.

- We can treat idiom formation as a kind of limit case of selectional restriction, and make those restrictions mutual:

  (7) *pull* V $(\uparrow \textsc{pred}) = \text{`pull}_{id}\text{'}$
  $(\uparrow \textsc{obj pred fn}) =_c \text{strings}_{id}$

  (8) *strings* N $(\uparrow \textsc{pred}) = \text{`strings}_{id}\text{'}$
  $((\textsc{obj} \uparrow) \textsc{pred fn}) =_c \text{pull}_{id}$

- This is too restrictive, of course, since this idiom can passivise.
- We can loosen the restriction by moving the constraint from f-structure to s-structure:

  (9) *pull* V $(\uparrow \textsc{pred}) = \text{`pull}_{id}\text{'}$
  $((\uparrow_\sigma \textsc{arg}_2)_{\sigma^{-1}} \textsc{pred fn}) =_c \text{strings}_{id}$

  (10) *strings* N $(\uparrow \textsc{pred}) = \text{`strings}_{id}\text{'}$
  $((\textsc{arg}_2 \uparrow_\sigma)_{\sigma^{-1}} \textsc{pred fn}) =_c \text{pull}_{id}$

- But this doesn't help with relative clauses:

  (11) *The strings (that) he pulled…*

$$\begin{bmatrix} \textsc{pred} & \text{`STRINGS}_{id}\text{'} \\ \textsc{adj} & \left\{ \begin{bmatrix} \textsc{pred} & \text{`PULL}_{id}\text{'} \\ \textsc{topic} & \begin{bmatrix} \textsc{pred} & \text{`PRO'} \end{bmatrix} \\ \textsc{subj} & \begin{bmatrix} \text{``HE''} \end{bmatrix} \\ \textsc{obj} & \boxed{} \end{bmatrix} \right\} \end{bmatrix}$$

- The word *strings* is never the object, nor the $\textsc{arg}_2$, of *pull*, so it shouldn't be licensed.
- Falk (2010) sees this as evidence for an 'unmediated' analysis of relative clauses. If we stick with the 'mediated' version, however, we cannot explain the distribution of (at least some) idiom chunks.

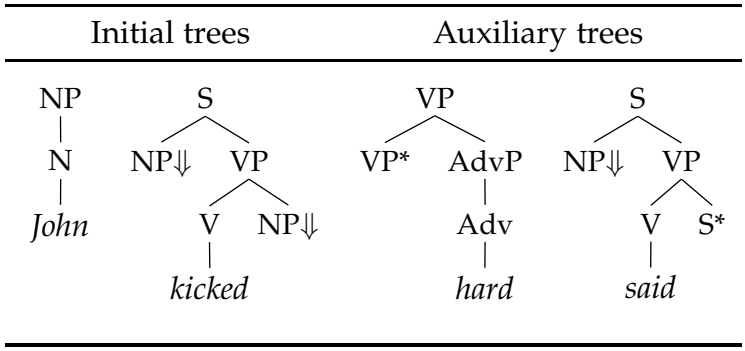


Table 1: Some elementary trees

4 Extending the power of c-structure

- The psycholinguistic findings plead for what seems intuitively appealing anyway: that idioms are inserted *en bloc*, being stored in the lexicon as units.
- This is consonant with Ackerman et al.'s (2011) approach to periphrasis: abandon PUE so that lexemes can be realised as multiple word forms.
- But the idiom data is worse: parts of the idiom can be arbitrarily far apart in c-structure, and do not have to appear in a fixed order.
- And, owing particularly to the 'mediated' analysis of relative clauses, we can't define the relationship at another level of representation either.
- Proposal: add power to the c-structure component so that such relations *are* statable. The 'extended domain of locality' of a TAG does just this.

4.1 LTAG

- In a TAG, trees, not words, are the elementary components of the grammar.
- TAG is a broad term for a mathematical formalism. Lexicalised TAG (LTAG) is the linguistically relevant subtype, where each tree must be 'anchored' by a word.
- A TAG consists of a set of *elementary trees* and the two operations of *substitution* and *adjunction* for combining them.

Elementary trees

- Elementary trees come in two types: *initial* and *auxiliary* (Table 1).
- An initial tree is a tree where all of the frontier nodes are either terminals or else non-terminals marked as *substitution sites* by a down arrow (⇓).[2] Substitution sites correspond to the arguments of a predicate.

[2] I depart from standard TAG practice of using ↓ so as to avoid confusion with the LFG metavariable.

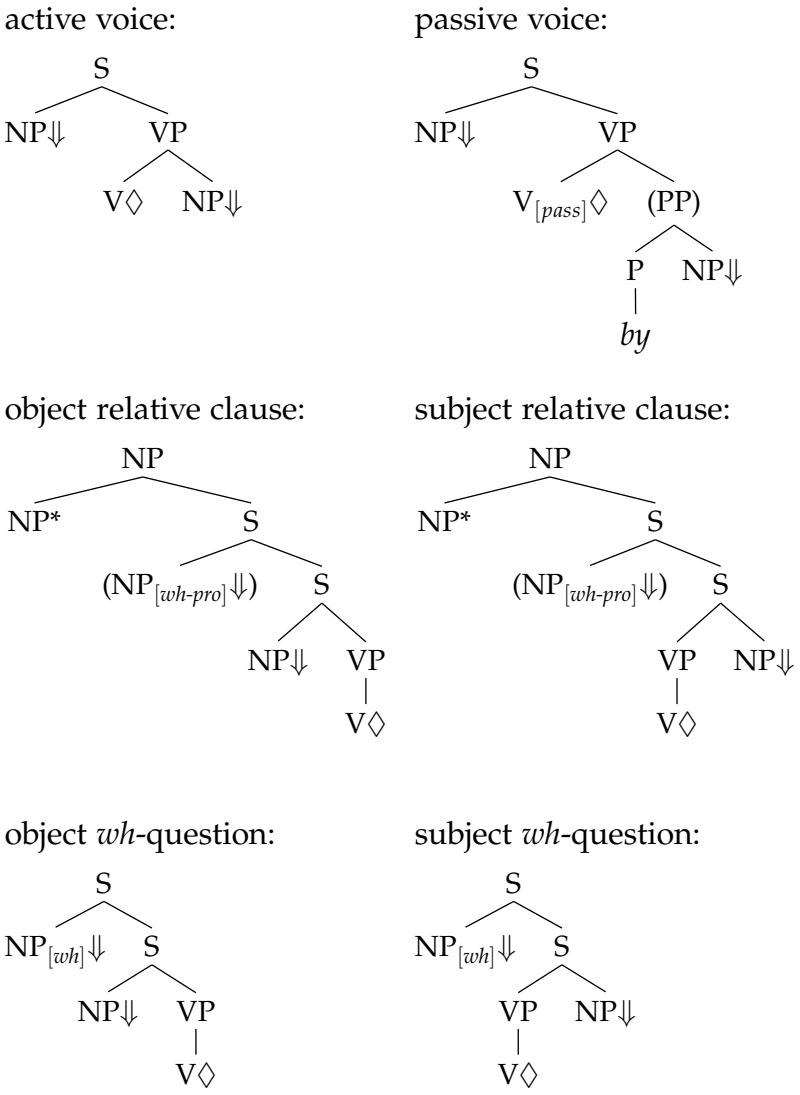
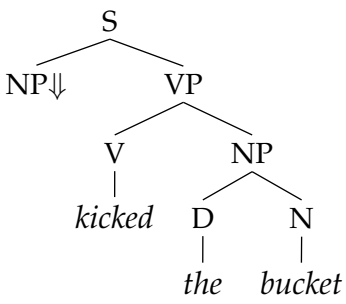


Figure 1: (Partial) tree family for a transitive verb

- An auxiliary tree is an elementary tree in which one of the frontier nodes is specified as the *foot* node, and marked with an asterisk (*). This node must be labelled with the same symbol as the root node of the auxiliary tree.
- Lexemes are associated with *tree families*, sets of trees which represent the potential syntactic realisations of the lexeme (Figure 1).
- Elementary trees can be 'multiply anchored', so that more than one frontier node is filled by a terminal node. This gives a good way of accounting for MWEs such as idioms (Abeillé 1995).

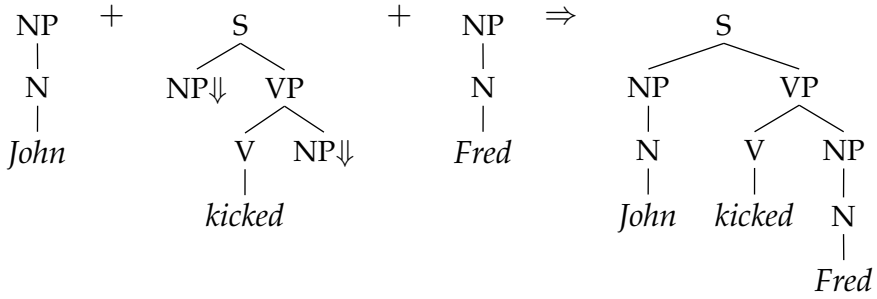
(12)



Substitution and adjunction

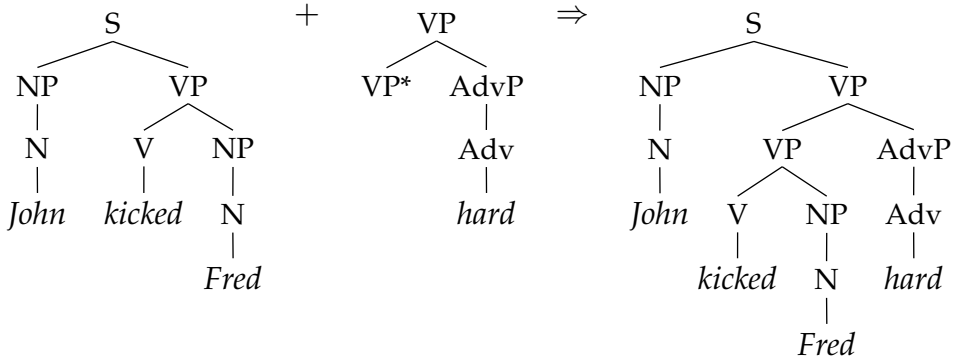
- Substitution is simply the replacement of an appropriate substitution site by an initial or derived tree whose root node matches the symbol at the substitution site.

(13)



- Adjunction, on the other hand, involves the splicing in of an auxiliary tree, from root node to foot node, in the place of any non-terminal node in another tree which matches the foot and root nodes of the auxiliary tree.

(14)



- In effect, the auxiliary tree is inserted at the adjunction site and 'expands' the node around itself.
- In addition to adjuncts, this is also how LTAG accounts for unbounded dependencies.
- Adjunction allows trees to grow 'from the inside out', as it were. This affords us what is often called an *extended domain of locality*: relationships can be encoded locally, even though the elements involved may end up arbitrarily far apart.

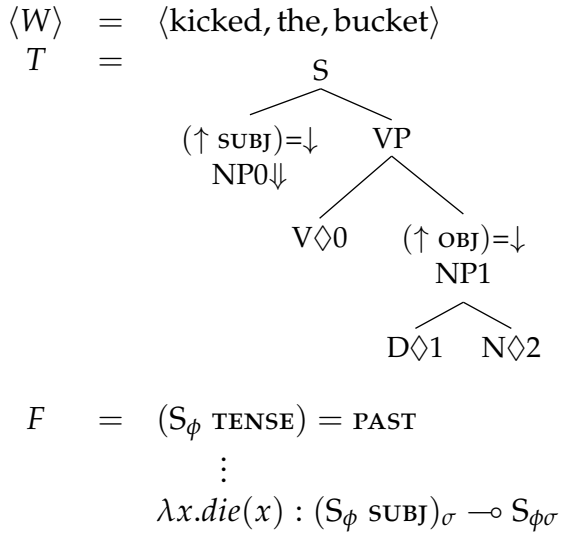


Figure 2: TAG-LFG lexical entry for *kicked the bucket*

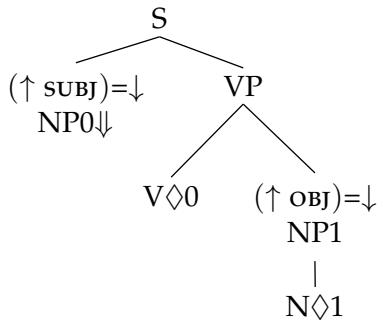
4.2 TAG-LFG

- In standard LFG, a lexical entry is a triple (W, C, F), where W is a word form, i.e. the terminal node in the phrase-structure tree, C is a c-structure category, i.e. the pre-terminal node, and F is a functional description, i.e. a set of expressions spelling out additional linguistic information via the correspondence architecture.
- In TAG-LFG, a lexical entry is instead a triple $(\langle W \rangle, T, F)$, consisting of a list of word forms, a tree, provided by some metagrammar (Crabbé et al. 2013), and a functional description. An example is given in Figure 2.
- The word forms occur as a list because the trees for MWEs will be multiply anchored. For regular lexical entries, this list will be a singleton.
- The lexical anchors, marked with ◊s, are numbered according to the list index of the lexeme that is to be inserted there.
- The functional description remains the same, although it now allows reference to more remote nodes, and so instead of ↑ or ↓ I use node labels as a shorthand for the nodes in question.[3]
- Decomposable idioms can have more than one meaning constructor associated with them, accounting for their internal modifiability (e.g. Figure 3).
- The differences in syntactic flexibility are represented in the different tree families which the idioms are related to.

[3]In reality, the node labels are not the nodes: they are the output of a node labelling function λ applied to each node (Kaplan 1995).

$\langle W \rangle = \langle \text{pulled, strings} \rangle$

$T =$



$F = (S_\phi \text{ TENSE}) = \text{PAST}$

$\vdots$

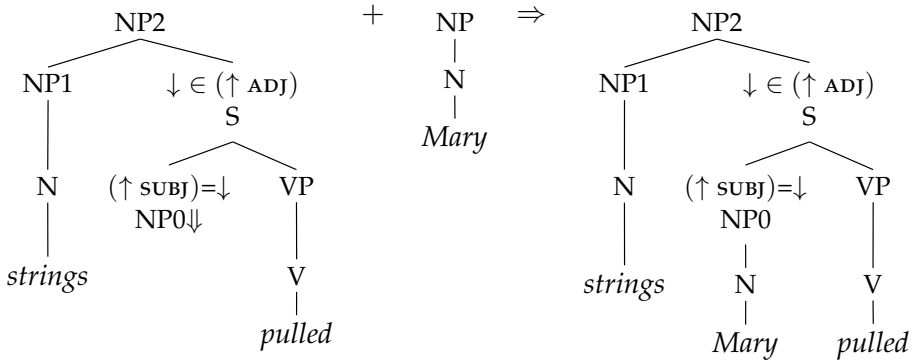
$\lambda x.connections(x) : (NP1_{\phi\sigma} \text{ VAR}) \multimap (NP1_{\phi\sigma} \text{ RESTR})$

$\lambda x \lambda y.exploit(x, y) : (S_\phi \text{ SUBJ})_\sigma \multimap (S_\phi \text{ OBJ})_\sigma \multimap S_{\phi\sigma}$

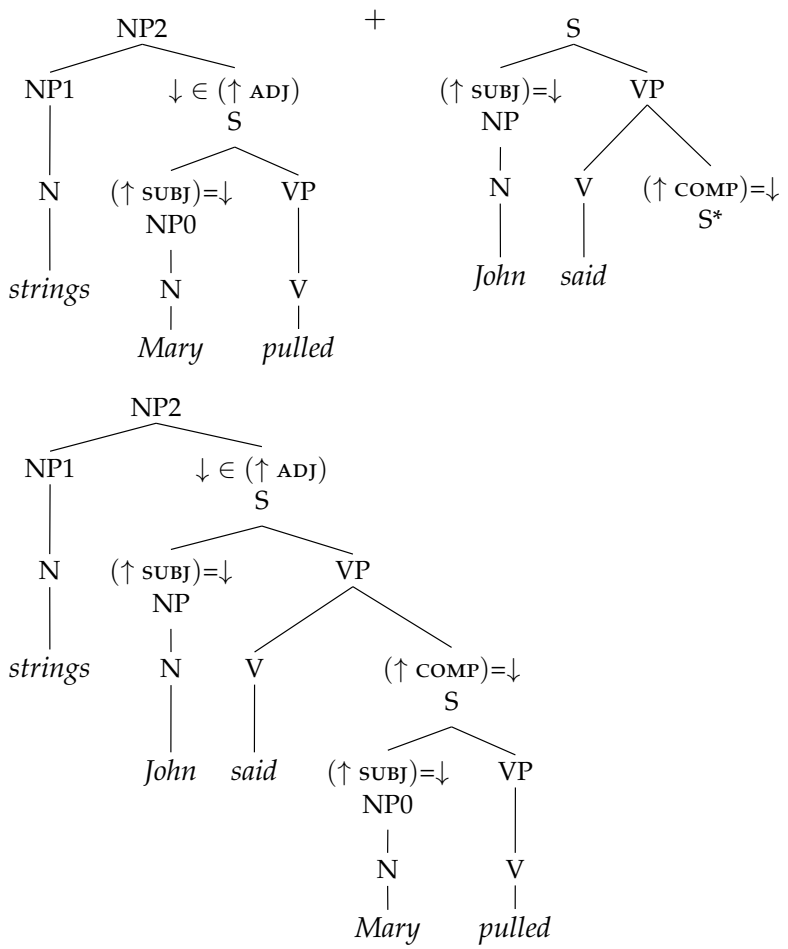
Figure 3: TAG-LFG lexical entry for *pulled strings*

- The unbounded dependency facts are accounted for by the extended domain of locality:

(15) a.



b.



- The TAG-based approach also aligns with the psycholinguistic findings, since a parse involving an idiom will involve fewer elementary trees (instead of the three trees for *kick*, *the*, and *bucket*, it will just involve the one for *kick the bucket*, for example), explaining the increased speed.

5 Conclusion

- Idioms, like periphrases, are cases where we want the lexicon to provide multi-word expressions to the syntax.
- Since some idioms are highly syntactically flexible, the final phrasal configuration of their parts cannot be described in advance, which motivates a move to a more powerful c-structure.
- Combining TAG with LFG allows us to take advantage of the 'extended domain of locality' of the former: this means we can describe the relationships between idiom parts locally, even if they are ultimately realised arbitrarily far apart.
- This allows us to describe each idiom in one place, in the lexicon, where such arbitrariness of form-meaning correspondence sits naturally.

References

Abeillé, Anne. 1995. The flexibility of French idioms: A representation with Lexicalized Tree Adjoining Grammar. In Martin Everaert, Erik-Jan van der Linden, André Schenk & Rob Schreuder (eds.), *Idioms: Structural and psychological perspectives*, Hove, UK: Lawrence Erlbaum.

Abeillé, Anne & Owen Rambow (eds.). 2000. *Tree Adjoining Grammars: Formalisms, linguistic analysis and processing*. Stanford, CA: CSLI Publications.

Ackerman, Farrell, Gregory T. Stump & Gert Webelhuth. 2011. Lexicalism, periphrasis, and implicative morphology. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar*, 325–358. Oxford, UK: Wiley-Blackwell.

Crabbé, Benoît, Denys Duchier, Claire Gardent, Joseph Le Roux & Yannick Parmentier. 2013. XMG: eXtensible MetaGrammar. *Computational Linguistics* 39(3). 591–629.

Cronk, Brian C. 1992. The comprehension of idioms: The effects of familiarity, literalness, and usage. *Applied Psycholinguistics* 13. 131–146.

Estill, Robert B. & Susan Kemper. 1982. Interpreting idioms. *Journal of Psycholinguistic Research* 11(6). 559–568.

Falk, Yehuda N. 2010. An unmediated analysis of relative clauses. In Miriam Butt & Tracy Holloway King (eds.), *Proceedings of the LFG10 Conference*, 207–227. CSLI Publications. `http://web.stanford.edu/group/cslipublications/cslipublications/LFG/15/papers/lfg10falk.pdf`.

Gibbs, Raymond W., Jr. 1986. Skating on thin ice: Literal meaning and understanding idioms in context. *Discourse Processes* 9. 17–30.

Joshi, Aravind K., Leon S. Levy & Masako Takahashi. 1975. Tree adjunct grammars. *Journal of Computer and System Sciences* 10(1). 136–163.

Kallmeyer, Laura & Maribel Romero. 2008. Scope and situation binding in LTAG using semantic unification. *Research on language and computation* 6(1). 3–52.

Kaplan, Ronald M. 1995. The formal architecture of Lexical-Functional Grammar. In Mary Dalrymple, Ronald M. Kaplan, John T. Maxwell, III, & Annie Zaenen (eds.), *Formal issues in Lexical-Functional Grammar*, 7–28. Stanford, CA: CSLI Publications.

Kay, Paul, Ivan A. Sag & Daniel P. Flickinger. 2015. A lexical theory of phrasal idioms. Unpublished ms., CSLI, Stanford. `http://www1.icsi.berkeley.edu/~kay/idiom-pdflatex.11-13-15.pdf`.

Lichte, Timm & Laura Kallmeyer. 2016. Same syntax, different semantics: A compositional approach to idiomaticity in multi-word expressions. In Christopher Piñón (ed.), *Empirical issues in syntax and semantics 11*, Paris: Colloque de Syntaxe et Sémantique à Paris (CSSP).

Sailer, Manfred. 2000. Combinatorial semantics and idiomatic expressions in Head-Driven Phrase Structure Grammar. Doctoral dissertation, Eberhard-Karls-Universität Tübingen.

Swinney, David A. & Anne Cutler. 1979. The access and processing of idiomatic expressions. *Journal of Verbal Learning and Verbal Behavior* 18. 523–534.