An incremental approach to verb clusters in German

Petter Haugereid
Western Norway University of Applied Sciences
petter.haugereid@hvl.no

Abstract

This paper presents an incremental approach to verb clusters in German which radically differs from standard HPSG accounts. While the common assumption is that the verbs in subordinate clauses form clusters and accumulate all their valence requirements on a SUBCAT list, the assumption in this paper is that the arguments in verb final clauses are encapsulated incrementally into syntactic and semantic structures before the verbs are attached. The proposed analysis is in line with psycholinguistic findings. A grammar fragment of German demonstrating an implementation of the analysis is presented.

1 Verb clusters in German HPSG

A widely studied topic in German syntax is that of verbal clusters, as illustrated in (1).

(1) daß ich den Jungen das Buch holen sah
that I the boy the book fetch saw
'that I saw the boy fetch the book'

The clause has an AcI[1] verb *sehen* 'see' which takes an infinitival complement and takes the subject of the infinitival complement as its direct object *den Jungen* 'the boy'. In Müller (2007a) it is given the SUBCAT value shown in Figure 1. The first element on the SUBCAT list is an NP subject (in (1) *ich* 'I'). The last element on the list is an embedded verb (in (1) *holen* 'fetch') which SUBJ and SUBCAT values ($\boxed{1}$ and $\boxed{2}$) also appear on the SUBCAT list of the AcI verb. This ensures that the arguments of the embedded verb (*den Jungen* 'the boy' and *das Buch* 'the book') end up on the subcat frame of the AcI verb.[2]

$$\begin{bmatrix} \textit{AcI-verb} \\ \text{SUBCAT} \left\langle \text{NP} \right\rangle \oplus \boxed{1} \oplus \boxed{2} \oplus \left\langle \text{V} \begin{bmatrix} \text{SUBJ } \boxed{1} \\ \text{SUBCAT } \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 1: AcI verb adapted from Müller (2007a)

The schema in Figure 2 shows how complex predicates are combined (Hinrichs and Nakazawa, 1994). In a clause like (1) the AcI verb and the embedded verb are combined. The AcI verb will then be the head daughter. The last element on its SUBCAT list is unified with the SYNSEM of the embedded verb ($\boxed{2}$). The SUBCAT list of the complex predicate ($\boxed{1}$) is the subcat list of the head daughter, except from the last element.

The arguments are subsequently realized by the Head Argument Schema (Müller, 2007a) shown in Figure 3.

[1] Accusative and Infinitive.

[2] Semantic roles and case are also important parts of the account, but that will not be discussed here.

$$\textit{head-cluster-phrase} \Rightarrow \begin{bmatrix} \text{SUBCAT} & \boxed{1} \\ \text{HEAD-DTR} & \begin{bmatrix} \text{SUBCAT } \boxed{1} \oplus \left\langle \boxed{2} \right\rangle \end{bmatrix} \\ \text{NON-HEAD-DTRS} & \left\langle \boxed{2} \right\rangle \end{bmatrix}$$

Figure 2: Schema for complex predicates (adapted from Müller (2007a, 242))

$$\textit{head-argument-phrase} \Rightarrow \begin{bmatrix} \text{SUBCAT} & \boxed{1} \oplus \boxed{3} \\ \text{HEAD-DTR} & \begin{bmatrix} \text{SUBCAT } \boxed{1} \oplus \left\langle \boxed{2} \right\rangle \oplus \boxed{3} \end{bmatrix} \\ \text{NON-HEAD-DTRS} & \left\langle \boxed{2} \right\rangle \end{bmatrix}$$

Figure 3: Head Argument Schema (adapted from Müller (2007a, 79))

This rule attaches the arguments one by one in a binary fashion. The fact that the rule splits the SUBCAT list of the head daughter in three, realizes the middle element ($\boxed{2}$) as the argument, and then concatenates the initial list ($\boxed{1}$) and the final list ($\boxed{3}$) in the SUBCAT of the mother, accounts for the fact that arguments may be permuted. The middle list may contain any of the arguments, since the lengths of list $\boxed{1}$ and $\boxed{3}$ are underspecified.

A challenge with the Head Argument Schema is that one needs to search through the SUBCAT list of the head daughter in order to find out where to do the split. The German Grammar, which does not allow for this functionality (Crysmann, 2003), resolves the problem by having different Head Cluster Rules, one for each possible permutation of the arguments. In this way, the argument realization rule does not have to split the SUBCAT list, it just needs to realize the first element. This, however, will lead to a large number of combinations of Head Cluster Rules if the number of embedding verbs is larger than one.

Another challenge with the approach is that it presupposes that the verbs are parsed before the arguments are assigned a syntactic or semantic role. The notion of words being incrementally added to an overall syntactic structure one by one (incremental processing) is well established in the psycholinguistic literature, evidenced by studies showing that sentences in head-final languages do not require higher processing than sentences in head-initial languages (Swets *et al.*, 2008). Studies on German show that there is an unmarked order in which arguments are processed (see Kretzschmar *et al.* (2012) and references therein). If an argument is locally ambiguous with regard to nominative or accusative case and it appears first of the arguments, it will typically be interpreted as the subject. If the final verb reveals that it is not the initial argument that is the subject, we get a garden path effect, and the clause will be reanalyzed.

2 An incremental approach

In this section I will show how complex predicates with multiple verb embeddings can be analyzed within the framework of Haugereid (2007, 2009).

2.1 Haugereid (2007)

In Haugereid (2007, 2009), arguments are assumed to be realized by five types of valence rules. There is one type of rules for agent or source arguments (CMP1-rules), one type for patient/theme arguments (CMP2-rules), one type for benefactive or recipient arguments (CMP3-rules), one for resultative or end-of-path arguments (CMP4-rules) and one for antecedents (CMP5-rules). These rules may apply before the verb(s) of the clause are attached. In addition to linking the argument to the predicate of the clause, each valence rule contributes an atomic valence type, and during the parse, the valence types are unified with an argument structure type assigned to the verb. When these types are unified, their greatest lower bound is a construction type. If the types do not have a greatest lower bound, the parse fails. This prevents verbs from being assigned arguments that they are not compatible with. Given the fact that the valence information of a verb is specified by the position of the argument frame type in the type hierarchy of valence types, and not by means of valence lists, the order of the arguments is not fixed in the lexicon. This opens for permutations of arguments in a way that is not possible with a lexicalist approach, as shown in (2). Here, the arguments are realized in a left-branching manner by the valence rules before the verb is attached. The binary design also allows for interspersable adjuncts.

(2) a. [[[[COMPL ARG4] ARG1] ARG2] V]
b. [[[[COMPL ARG4] ARG2] ARG1] V]
c. [[[[COMPL ARG1] ARG4] ARG2] V]
d. [[[[COMPL ARG2] ARG4] ARG1] V]

2.2 Criticism of Haugereid (2007)

Müller (2007b) points out a problem with the approach in Haugereid (2007), namely that there will be a need for a new set of valence rules for each embedding verb (raising verbs and control verbs) in a verbal cluster. The rules assumed in Haugereid (2007) only account for the arguments of the matrix verb. In order to account for the arguments of the embedded verb(s), additional rules would be needed in order to link the arguments of the embedded verbs. The example in (3) has two embedding verbs (*helfen* 'help' and *läßt* 'let'), and an analysis would require three sets of valence rules, linking at different levels of embedding. This number would be multiplied by two since each rule has an extraction variant. Müller (2007b) argues that the number of embeddings in verbal clusters is limited by performance, and that a grammar in principle should allow for an unlimited number of embeddings. This would be unfeasible with the N levels deep linking approach inferred from Haugereid (2007).

(3) weil Hans Cecilia John das Nilpferd
because Hans Cecilia John the hippo
füttern helfen läßt.
feed help let
'because Hans lets Cecilia help John feed the hippo.'

2.3 Analysis of verb clusters in German

The problem with the asserted N levels deep linking approach can be solved by means of three unary embedding rules, one for linking the subject of the embedded clause to the subject of the matrix clause (subject raising/control), one for linking the subject of the embedded clause to the indirect object of the matrix clause (object control), and one for linking the subject of the embedded clause to the direct object of the clause (AcI verbs). Figure 4 shows the rule for object control. This rule takes as input a structure, and outputs a structure embedded in the initial structure. The SYNSEM of the input structure is put on a STACK. The rule constrains the argument frame type of the input structure (the matrix clause) to be of type *arg123*, which means that it should have three arguments (an agent, a patient/theme, and a benefactive). The ARG2 of the input structure is linked to the label of the output (the embedded clause). The ARG3 of the input structure is linked to the subject of the embedded clause.

Once the embedded structure has been entered, the valence rules can be employed in a regular fashion. There is principally no limit to how many times the unary embedding rule can be used, and so the linking of arguments embedded two levels deep is no longer

$$\begin{bmatrix} \textit{unary-obj-control-phrase} \\ \text{SYNSEM } \boxed{1} \begin{bmatrix} \text{SUBJ} & [\text{INDEX } \boxed{2}] \\ \text{LTOP} & \boxed{3} \\ \text{INDEX} & [\text{TENSE } \textit{infin}] \end{bmatrix} \\ \text{STACK } \langle \boxed{4} \rangle \\ \text{ARGS } \left\langle \begin{bmatrix} \text{SYNSEM } \boxed{4} \begin{bmatrix} \text{VAL} \begin{bmatrix} \text{ARGFRAME } \textit{arg123} \\ \text{CMP2 } \boxed{1} \end{bmatrix} \\ \text{KEYREL} \begin{bmatrix} \text{ARG2 } \boxed{3} \\ \text{ARG3 } \boxed{2} \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 4: Rule for entering embedded structures with object control in German

a problem.

In addition to the unary embedding rules, the grammar also has a unary popping rule, which pops out of embedded structures (see Figure 5).

$$\begin{bmatrix} \textit{popping-rule} \\ \text{SYNSEM} & \boxed{1} \\ \text{STACK} & \boxed{2} \\ \text{ARGS} & \left\langle \begin{bmatrix} \text{STACK} & \langle \boxed{1} \rangle \oplus \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

Figure 5: Rule for exiting embedded structures

The embedding and popping rules works in tandem with the valence rules as shown in Figure 6.

The parse starts in the bottom left corner with the complementizer *weil*. First the subject *Hans*, and the indirect object *Cecilia* are attached. Then the AcI rule works. It enters the SYNSEM of the AVM parsed so far, onto a STACK in the mother. Now, the second indirect object *John* is attached. Note that it is the same type of *cmp3-rule* attaching both *Cecilia* and *John*. No extra valence rule is required, even though the two arguments are at different levels of embedding. The next step is to enter another level of embedding (an object control structure, see Figure 4) before the final argument *das Nilpferd* is attached. At this point there are two elements on the STACK list, showing the level of embedding. After the arguments are attached, the verbs are attached at the appropriate levels of embedding.

3 Implementation and discussion

The analysis is implemented with the LKB system (Copestake, 2002) in a German demo grammar (Haugereid, 2009, 308-313) based on the Norwegian HPSG grammar Norsyg (Haugereid, 2009). Apart from the lexicon, only slight alterations are made in order to account for the basic clause structures in German.[3] It successfully analyses the examples in (1) and (3) and produces proper semantic representations. The implementation demonstrates that the analysis works, and the grammar analyzes verb-final clauses with multiple embeddings like example (3).

Currently, the implementation only opens for scrambling locally, that is, at the same level of embedding. In order to allow for scrambling between embeddings, allowing for example *das Nilpferd* in (3) to come before the other arguments, the embedding rules need to be less constrained, that is, they will have to be applicable before all arguments at a level of embedding are realized.[4] This loosening of constraints is not feasible, since the embedding rule then could take itself as input, and the LKB system does not have a way to explore one level of embedding at a time and stop when it arrived (or did not

[3] In addition to the changes described in Haugereid (2009, 308–310), a unary version is made of the object control rule, and both the object control rule and the AcI rule (which already was unary) were allowed to apply before the verb.

[4] Currently, they are constrained to apply after the arguments at the matrix level are realized.

arrive) at an analysis. This would however be an interesting path to pursue, as it would be in line with psycholinguistic findings of garden-path effects, involving backtracking and reanalysis. Whenever the parser has to backtrack from attempting to parse the unmarked order of the arguments of a sentence, the effort on the parser would increase, just like the human processing efforts are increasing when attempting to process a garden path sentence.

References

Copestake, A. (2002). *Implementing Typed Feature Structure Grammars*. CSLI publications.

Crysmann, B. (2003). On the efficient implementation of German verb placement in HPSG. In *Proceedings of RANLP 2003*, pages 112–116, Borovets, Bulgaria.

Haugereid, P. (2007). Decomposed phrasal constructions. In S. Müller, editor, *The Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, pages 120–129, Stanford. CSLI Publications.

Haugereid, P. (2009). *Phrasal subconstructions: A constructionalist grammar design, exemplified with Norwegian and English*. Ph.D. thesis, Norwegian University of Science and Technology.

Hinrichs, E. and Nakazawa, T. (1994). Linearizing AUXs in German verbal complexes. In J. Nerbonne, K. Netter, and C. J. Pollard, editors, *German in Head-Driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes, pages 11–37. CSLI Publications, Stanford University.

Kretzschmar, F., Bornkessel-Schlesewsky, I., Staub, A., Roehm, D., and Schlesewsky, M. (2012). Prominence facilitates ambiguity resolution: On the interaction between referentiality, thematic roles and word order in syntactic reanalysis. In *Case, word order and prominence*, pages 239–271. Springer.

Müller, S. (2007a). *Head-Driven Phrase Structure Grammar: Eine Einführung*. Number 17 in Stauffenburg Einführungen. Stauffenburg Verlag, Tübingen.

Müller, S. (2007b). Phrasal or lexical constructions: Some comments on underspecification of constituent order, compositionality, and control. In S. Müller, editor, *The Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, pages 373–393, Stanford. CSLI Publications.

Swets, B., Desmet, T., Clifton Jr., C., and Ferreira, F. (2008). Underspecification of syntactic ambiguities: Evidence from self-paced reading. *Memory & Cognition*, **36**(1), 201–216.

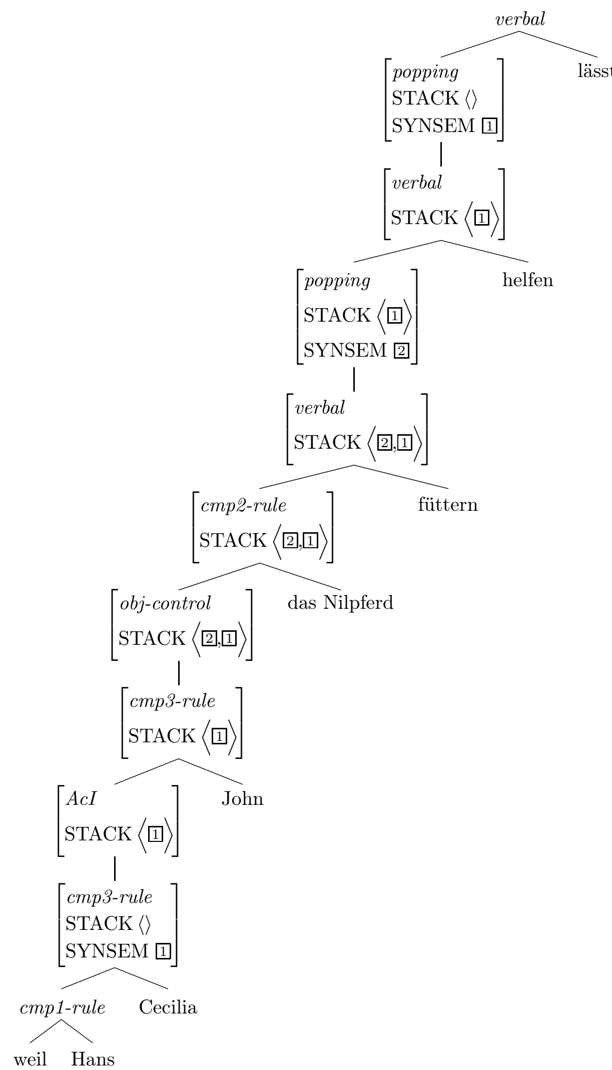

Figure 6: Left-branching analysis of German subordinate clause with two embeddings