Integrating elliptic conjunctions into a CCG based parser

Tilman Vierhuff, Sabine Heidenreich und Bernd Hildebrandt

Abstract

The variant of Steedman's Combinatory Categorial Grammar used in the Collaborative Research Center 360 "Situated Artificial Communicators" allows efficient and incremental processing of German instructions for a cognitive robotics system. This paper introduces the grammar formalism and elaborates on the rules Steedman suggested for inclusion of coordinating constructs. It is shown that several instances of coordination rules are needed in order to capture different phenomena of coordination.

The traditional unification mechanism prevailing in contemporary computational linguistics and also used in the SFB 360 is incapable of coping with elliptic coordination. We will show that the introduction of directional information flow overcomes this problem without sacrificing the descriptive potential of unification. Additionally, a limited short–term memory is integrated into the parser which allows for processing of many different elliptic sentences.

1 Introduction

Categorial Grammars have become more popular since Steedman introduced *Combinatory Catgorial Grammar* by including combinatory rules [Ste87] into traditional Categorial Grammar, thereby raising its weak generative capacity from the context–free to the mildly context–sensitive level [JVW91]. Furthermore, he suggested that the processing of coordinative structures could be made possible by means of two general coordination rules that bind a conjunction to its right and its left conjunct, respectively [Ste93].

In the context of the Collaborative Research Center 360 "Situated Artificial Communicators", Steedman's CCG has been adapted to fit the specific properties of German sentences in a Baufix construction scenario. A two–armed robot may be given instructions in imperative ("take a bolt") or assertive form ("and now you take a bolt"). The flexibility of the grammar allows free German word order as well as the use of modal verbs [HE99].

However, in natural human language, coordinative structures, especially elliptic ones, are quite common though not yet covered in the SFB 360 grammar. This paper will show how Steedman's coordination rules may be adapted and what the implications are that must be attended to as a consequence.

2 Combinatory Categorial Grammar

Whereas phrase structure grammars (PSGs) such as the widely used head–driven phrase structure grammar (HPSG, e.g. [PS94]) define various rules for any meaningful combination of terminal (i.e. words) or nonterminal symbols (i.e. phrases and their projections), Categorial Grammars (CGs) are extensively lexicalised. Lexicon entries contain all information about possible arguments whereas their generic rules do not depend on the symbols they combine.

2.1 Categories

Each usage of a word is assigned a category, which may be atomic or complex. Complex categories can be viewed as functions over categories, which are composed or applied to their arguments by the parser. In contrast to the mathematical notation of a function $f : x \rightarrow y$, a functional category taking an argument of type x and yielding a result of type y is written as y/x or $y\backslash x$, depending on whether the argument x is expected to the right or to the left of the function. The leftmost category is always the result, followed by

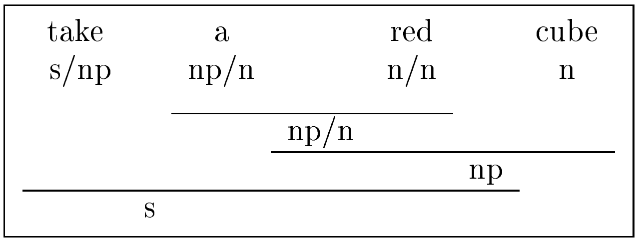

Figure 1: Sample derivation structure using forward composition and application

its arguments[1].

As an example, the verb “take” in the sentence “take a bolt” can be interpreted as a function mapping nominal phrases to sentences (s/np) and the determiner “the” as a function mapping nouns to the nominal phrases (np/n). According to mathematical rules, these two functions can be composed into a function from the nouns into the sentences (s/n) and consequently be applied to the noun “bolt” (n), yielding a sentence as an overall result. Parameters may be satisfied inedependently of each other which allows incremental processing.

Note that there is no parse tree generated as an intermediate data structure. Categorial grammars construct the semantic interpretation of a sentence incrementally as they combine categories. While it is possible to depict a so–called derivation structure, this merely reflects the temporal development. An example is given in figure 1.

2.2 Reduction rules

Since the categories already contain all the information about possible arguments, it is unnecessary to repeat their relations in huge sets of specific rules as phrase structure grammars do. Instead, a small number of generic rules is defined which may be applied to any category with a corresponding functional structure. Two simple rules, namely instances of functional composition and application, have been implicitly used in figure 1. From a mathematical point of view, their actions are quite intuitive, nonetheless in a categorial grammar they are stated explicitly as $X/Y + Y/Z \to X/Z$ (functional composition) and $X/Y + Y \to X$ (functional application).

Originally, functional application was the only way to combine categories

[1] Category structures are not formed consistently across publications. This paper follows Steedmans notation and may not fit the way the reader is accustomed to.

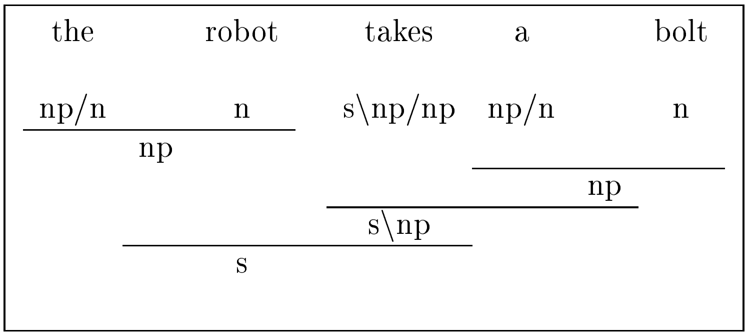

Figure 2: Derivation using only forward application (pure CG)

in pure categorial grammar. Functional composition was introduced by Steedman along with other rules which implement logical combinators as discussed by [CF58].

As categories may be connected by forward or backward connectors, there exist two variations of functional application and four of functional composition. However, only the first of the four variants of functional composition is used in the grammar discussed here.

Functional application, forward	$X/Y + Y \rightarrow X$
Functional application, backward	$Y + X\backslash Y \rightarrow X$
Forward functional composition	$X/Y + Y/Z \rightarrow X/Z$
Forward crossing functional composition	$X/Y + Y\backslash Z \rightarrow X\backslash Z$
Backward functional composition	$Y\backslash Z + X\backslash Z \rightarrow X\backslash Z$
Backward crossing functional composition	$Y/Z + X\backslash Y \rightarrow X/Z$

The sentence “the robot takes a bolt” may be processed using only the forward application rule (figure 2). If incremental parsing is desired, though, the subject should be bound to the verb before the object is considered. This requires another class of rules, which raise the type of a given category to another complex type of higher order, thus called *type–raising rules.*

Forward type–raising	$X \rightarrow T/(T\backslash X)$
Backward type–raising	$X \rightarrow T\backslash(T/X)$
Forward crossed type–raising	$X \rightarrow T/(T/X)$
Backward crossed type–raising	$X \rightarrow T\backslash(T\backslash X)$

Usually, only the first two rules are used, because the latter ones violate the inherent directionality of the connectors. Although the principle of di-

```
the          robot    takes        a       bolt

np/n         np       (s\np)/np    np/n    n
------------------
     np
------------------
  s/(s\np)
----------------------------
             s/np
                                 -----------------
                                        np
             ----------------------------------
                              s
```

Figure 3: Incremental derivation of subject *np* using explicit type–raising

rectionality was proposed by Steedman as an axiom for CCG [Ste93], we will see later that in certain contexts controlled violations are desirable.

The first type–raising rule allows incremental processing of the aforementioned sentence (figure 3).

There are some problems associated with type–raising. If type–raising is not restricted, it can be applied to its own results recursively, thus creating more and more complex categories. Furthermore, it is possible to use type–raising unnecessarily. For example, before the last step in figure 2, forward type–raising could be applied to the subject *np* without any effect to the derivation structure, thus complicating the parsing unnecessarily. Finally, the category for T is not defined by the rule and must be provided somehow.

Several solutions to these problems have been proposed. Michael Niv [Niv94] suggests putting type–raised categories into the lexicon, thereby rendering explicit type–raising rules dispensable. Obviously, this solves the problem with the undefined category T too, as is must be chosen when designing the lexicon. This approach would lead to an even more lexicalised grammar, such as is striven for by categorial grammars. But since it is not always clear which combinations of categories need to be raised, the lexicon gets more complex than necessary or may not even be able to capture the full power proper type–rasing rules would have provided. At any rate, blowing up the lexicon creates ambiguities at word level.

Steedman integrates another adjacent category into the type–raising rule and hence makes sure it can be applied only once, as long as it fits the context:

$$X + (T\backslash X)/Y \rightarrow T/(T\backslash X) + (T\backslash X)/Y$$

Moreover, the context category provides the wanted category for T. But this does not help against fully redundant applications of type–raising rules to categories which can be reduced otherwise, as pointed out by [HE99].

Our approach is based on the fact that sensible type–raising is only needed to enable functional composition of categories. Because it is always followed by a functional composition, we enforce this combination of applications by deriving new rules with implicit type–raising.

Type–Raised Forward Composition	$X/Y/Z + Y \rightarrow X/Z$
Type–Raised Backward Composition	$Y + X\backslash Y/Z \rightarrow X/Z$

These rules do exactly the same as the explicit type–raising in figure 3, albeit only under meaningful circumstances. We therefore refrain from giving another example showing their use.

2.3 Word order and constituent counting

In contrast to English, the first constituent in a German main clause need not be the subject, but may be any complement of the verb (e.g. "den Würfel nimmt der Roboter", "the cube$_{obj}$ takes the robot$_{subj}$"), even if the case is not marked by declination. Therefore, the direction of the slashes may be dropped for verb complements, thus allowing topicalization of any complement.

One problem with this approach is the fact that ignoring slash directions allows arbitrary numbers of constituents before and after the verb. Yet, there are three well–defined classes of sentences in German which must be clearly reflected in the grammar formalism. In main clauses and wh–questions one constituent precedes the conjugated verb while the others follow, in yes–no–questions the conjugated verb stands in the first and in subordinate clauses in the last position. Additionally, parts of the verbal complex may be placed at the end of the sentence, thereby forming the German "verbal paranthese" (*Verbklammer*). As verbs in subordinate clauses, they require all their complements to precede them.

To prevent over–generalization, a counter is introduced which restricts word order to German syntax. Initially, it is set to the complete number of complements. Every time, a constituent is drawn in front of the verb, the counter is decreased. By checking this counter after successful parsing, the sentence type may be deduced: A value of one indicates a main clause whereas two indicates a subordinate clause and yes–no–questions let the counter come down to zero. Other counter values, denoting a certain degree of ungrammaticality, may occur in poetic speech where German sentence structure is somewhat relaxed or in spontaneous speech (e.g. supplementary information after a previously closed relative clause)[2].

Further, in combination with Steedman's "dollar convention" (also called "generalised rules", cf. [Ste90]), these rules can cope with German free word order. Since the dollar convention means that the rightmost complement of a complex category may stand for an arbitrarily long list of complements, any embedded complement is accessible despite of its position. Thus, any topicalised constituent may precede the verb in a main clause.

In the following rules, all slashes are written as '/' because their direction is no longer of importance and may be overridden by the second rule. The index i represents the constituent counter, which also serves as a constraint since these rules may only be applied to categories that contain such a counter, i.e. verbs.

Indexed Type–raised Forward Composition	$Y + X_i/Y/Z \rightarrow X_{i-1}/Z$
Indexed Type–raised Backward Composition	$Y + X_i/Y/Z \rightarrow X_{i-1}/Z$

3 Processing of conjunctions

Steedman proposes integration of coordinational structures into CCG by means of a new type of category and a pair of generic rules which bind two conjoined categories successively together [Ste93]. The semantics of the final result are to contain the duplicated semantics of the original categories.

Coordination rule A	$conj + X \rightarrow [X]_{\&}$
Coordination rule B	$X' + [X]_{\&} \rightarrow X''$

The same result can be achieved with a single rule binding two categories together in one reduction. Steedman justifies the splitting of the rules by arguing that English conjunction is right–associative. For speech processing issues, this is of minor importance since the results do not differ, hence we prefer a simple rule that combines both conjuncts in a single step.

Combined Coordination rule $\quad X + conj + X' \rightarrow X''$

[2]In verbal complexes consisting of several auxiliary, modal or main verbs, possibly as participles, infinitives or other forms, each verb carries its own counter. For example, auxiliary and modal verbs are modelled as functions over participles with a maximal counter value. The discussion of this topic goes beyond the scope of this paper, for a more detailed discussion see e.g. [HE99]

Superficially, this rule seems flexible enough to combine any two categories of the same type. But for different categories conjoining them means different things [Vie00]:

- Conjoining nouns results in another noun whose number is plural, independent of the number of the original categories
- Conjoining of adjectives (of category n/n) obviously yields another adjective. Because the properties of adjectives and nouns are kept in lists, which may cumulate unconfined entries, the semantics part of a rule combining adjectives must be tailored to manage merging these property lists
- Conjoining of complex categories such as verbs lacking several complements is special, because not only the complements must match, but also the semantics needed for the result must be pulled out of a value–complement structure of unknown complexity
- Conjoining of different categories as in "I don't know much about history nor what a slide rule is for" and of more complicated elliptic constructions (see section 5) is not addressed by these rules.

Consequently, Steedman's coordination rules stand for a group of specialised rules with the necessary abilities. The speech processing system of the SFB 360 therefore defines several different coordination rules, each serving a different purpose.

First, the case of nouns becoming plural can be covered by the following rule.

Noun coordination rule $\quad np_{num:?} + conj + np_{num:?} \rightarrow np_{num:plural}$

While other agreement information, such as case, is included in the result category, genus may be ignored completely in German because there is only one plural form. In romanic languages like French of Spanish where genus–marked plural forms affect other words, genus would have to be accounted for.

Second, adjectives are conjoined by the following rule. Boxed numbers stand for structure sharing as used in unification grammars. Necessary adaptions of the unification formalism will be discussed in the next section. For the present, asssume equal numbers mean identical substructures.

Adjective coordination rule $\quad n_{\boxed{1}S_1}/n_{\boxed{1}} + conj + n_{\boxed{2}S_2}/n_{\boxed{2}} \rightarrow n_{\boxed{3}SEM:S_1 \sqcup S_2}/n_{\boxed{3}}$

It would be possible to capture the subtle difference between “a red and italian wine” and “a red italian wine” by marking the adjective semantics accordingly on the right hand side. In the context of the SFB 360, this difference is irrelevant and therefore ignored.

Third, categories of any complexity without unusual changes in their syntactic or semantic representaion are processed using the subsequent rule

$$\text{Complex coordination rule} \quad X/\$Y + conj + X'/\$Y \rightarrow X''/\$Y.$$

As usual with Steedman's dollar convention, $\$Y$ stands for an arbitrarily long sequence of arguments [Ste90], including the emtpy sequence. A new concept is the triple appearance of the argument variable $\$Y$. Arguments found in one of the input categories must be checked against the other and in case of success transferred to the result. To prevent conflicts with the previously mentioned rules, a constraint forbids its use when X is of type n.

Elliptic coordinations in general are far more versatile and thus are discussed in more detail within section 5.

4 Elliptic constructs and unification

Although the whole domain of ellipsis is not covered by Steedman's rules, some elliptic constructs are processed syntactically in a seemingly easy manner by the rules defined up to now, as shown in figure 4. Yet, there is a subtle problem in semantics.

Internally, categories are represented as attribute–value–matrices (AVMs) and processed via unification, as defined, for example, in [Shi86]. This mechanism is widely used under various grammar formalisms in contemporary computational linguistics.

A core feature of unification is the notion of *structure sharing* which means that different AVMs may contain references to identical substructures. If such a substructure is altered in any way, it changes its contents for all AVMs that contain a reference to it. This feature leads to unwanted mixing of feature structures in German sentences as in example 1, which is an order to take two cubes, of which one is red and one is small.

Example 1 *Nimm einen roten $\emptyset$ und einen kleinen Würfel.*
(take one red and one small cube)

In order to construct the semantic structures for the two mentioned cubes, it is necessary to unify the semantic structure of “cube” with the structure of “red” on the one hand and with the structure of “small” on the other. Doing

einen	roten	und	einen	kleinen	Würfel
np/n	n/n	$conj$	np/n	n/n	n
$np_{(rot)}/n$					
			$np_{(lang)}/n$		
		$np_{(rot,lang)}/n$			
			$np_{(rot,lang)}$		

Figure 4: Problematic derivation

so by means of structure sharing leads to the three structures contained in the sentence becoming identical because the sharing relation is transitive. The intermediate representation of the partial phrase "den roten und den kleinen", which only lacks a noun in order to become a nominal phrase, is shown in figure 5.

We will not dive deeply into unification issues here, assuming that the process is well–known. Nonetheless, we remind the reader that equally numbered tags stand for identical substructures. Therefore, not only does the information of the expected noun flow into the meaning of the two adjectival structures, but also these fuse with each other. The order gets misunderstood as referring to two cubes that are both red and small.

Directed structure flow

The problem is that traditional unification is too general because it assumes identity between structures. Actually, only one way of information flow is needed, namely a flow from simpler to more complex constituents. Since there is no need to pass any information from higher order constituents downwards, we replace structure sharing with directed structure flow, written as $\boxed{i} \leftarrow \boxed{j}$.

This means, that all of the information in structure $\boxed{j}$ is taken over into structure $\boxed{i}$ without regarding both as being identical. More precisely, structure $\boxed{j}$ stays exactly as before, only structure $\boxed{i}$ is enriched with all of $\boxed{j}$'s information. We call $\boxed{j}$ the source and $\boxed{i}$ the destination structure [Vie2001].

Using this approach, the structure in question becomes as shown in figure 6. When this structure is unified with the missing noun "cube", the meaning of "cube" is safely transferred into the result semantics in the noun phrase part without the two becoming identical.

As traditional structure sharing tags may accumulate during parsing, directed flow tags may form chains. For instance, if $\boxed{j}$ in $\boxed{i} \leftarrow \boxed{j}$ is unified with

$$\left[\begin{array}{ll} \text{VAL}: & \left[\begin{array}{ll} \text{SYN}: & \left[\begin{array}{ll} \text{CAT}: & \text{np} \\ \text{AGREE}: & \left[\begin{array}{ll} \text{CAS}: & \text{ACC} \\ \text{GEN}: & \text{MASC} \\ \text{NUM}: & \text{SING} \end{array}\right]\boxed{1} \end{array}\right] \\ \text{SEM}: & \left[\begin{array}{ll} \text{TYPE}: & \text{CONJ} \\ \text{ARG1}: & \left[\text{SIZE}: \ \text{long}\right]\boxed{2} \\ \text{ARG2}: & \left[\text{COLOR}: \ \text{red}\right]\boxed{3} \end{array}\right] \end{array}\right] \\ \text{OP}: & / \\ \text{ARG}: & \left[\begin{array}{ll} \text{SYN}: & \left[\begin{array}{ll} \text{CAT}: & \text{n} \\ \text{AGREE}: & \boxed{1} \end{array}\right] \\ \text{SEM}: & \boxed{2}\ \boxed{3} \end{array}\right] \end{array}\right]$$

Figure 5: Problematic feature structure: Substructures 2 and 3 are identical, thus the properties *red* and *long* are mixed up

$$\left[\begin{array}{ll} \text{VAL}: & \left[\begin{array}{ll} \text{SYN}: & \left[\begin{array}{ll} \text{CAT}: & \text{np} \\ \text{AGREE}: & \left[\begin{array}{ll} \text{CAS}: & \text{acc} \\ \text{GEN}: & \text{masc} \\ \text{NUM}: & \text{sing} \end{array}\right]\boxed{1} \end{array}\right] \\ \text{SEM}: & \left[\begin{array}{lll} \text{TYPE}: & \text{CONJ} & \\ \text{ARG1}: & \left[\text{SIZE}: \ \text{long}\right] & \boxed{2} \leftarrow \boxed{4} \\ \text{ARG2}: & \left[\text{COLOR}: \ \text{red}\right] & \boxed{3} \leftarrow \boxed{4} \end{array}\right] \\ \text{OP}: & / \\ \text{ARG}: & \left[\begin{array}{ll} \text{SYN}: & \left[\begin{array}{ll} \text{CAT}: & \text{n} \\ \text{AGREE}: & \boxed{1} \end{array}\right] \\ \text{SEM}: & \boxed{4} \end{array}\right] \end{array}\right] \end{array}\right]$$

Figure 6: Feature structure with directed information flow avoids the problem of mixing unrelated conjuncts

$\boxed{j'}$ in $\boxed{j'} \leftarrow \boxed{k}$, the result is of the form $\boxed{i} \leftarrow \boxed{j''} \leftarrow \boxed{k}$, where $j'' = j \sqcup j'$. Just like superflous traditional tags may be removed when occurring in sequences (which is a common phenomenon after unification), the intermediates of directed flow chains may be omitted as soon as their information has been integrated into the destination structure, thus yielding $\boxed{i \sqcup j''} \leftarrow \boxed{k}$ as a result.

This notion is only more specific than Shieber unification without sacrificing any functionality. We merely removed undesirable generality, which was never needed and which in elliptic sentences corrupts the proper interpretation.

Note that the problem discussed and solved here is no artefact of categorial grammar. Any grammar which resolves elliptic references by means of unification must face the problem that two incomplete constituents are completed by the same complement. The use of structure sharing will always lead to common substructures, unless unification becomes restricted as demonstrated here.

5 General elliptic constructs

There are far more than the aforementioned elliptic constructs with adjoining categories of equal syntactic type. [GKS+93] discriminates four classes of ellipses:

- **cataphoric coordinated ellipses** omit a constituent which will be mentioned in a later position. In German they occur only factored out to the right in phrase sequences of the form (a *conj* b)c as in example 2.

  Example 2 *"Der Kanzler begrüsst $\emptyset$ und der Reporter befragt den Staatschef"*
  (the chancellor welcomes and the reporter interviews the head of state).

- **anaphoric coordinated ellipses** omit a previously introduced constituent. They exist in three variations.

  - **anaphoric coordinated ellipses factored out to the left** mirror the behaviour of the first class as they appear in the form a(b *conj* c) as example 3.

    Example 3 *"Den Staatschef begrüsst der Kanzler und $\emptyset$ befragt der Reporter"* *([the head of state]$_{obj}$ welcomes [the chancellor]$_{subj}$ and interviews [the reporter]$_{subj}$)*

The factored out constituent must be functionally equivalent in both contexts, mere syntactic equality is not enough. Thus, example 4 is inacceptable. The same applies to ellipses factored out to the right.

Example 4 **Dem Lehrer gefällt das Mädchen und* $\emptyset$ *misstraut der Junge*[3]
([The teacher]$_{obj}$ appeals to [the girl]$_{subj}$ and distrusts [the boy]$_{subj}$)

– **Gapping** is a form of inner ellipsis where parts of the verbal phrase including the finite verb may be omitted, partly in combination with the subject. An instance for the latter case is example 5.

  Example 5 *"Heute begrüsst der Kanzler den Staatschef und morgen den Minister"*
  (today the chancellor welcomes the head of state and tomorrow $\emptyset$ *the minister)*

  Agreement between constituents and a gapped verb is not required, as example 6 proves—the correct verb form for the elliptic part of the sentence would be *"isst"* (eats), not *"esse"* (eat). On th other hand, example 7 is strange at best because of the different syntactic type of otherwise identical verb complements.

  Example 6 *Ich esse Reis und sie Kartoffeln*
  (I eat rice and she, potatoes)

  Example 7 **Gert fährt seine Kinder zur Schule und Michael* $\emptyset$ *seinem Kollegen zu schnell*
  (Gert drives his children to school and Michael $\emptyset$ *too fast for his colleagues)*

– Finally, **interior subject ellipses** resemble gapping with the exception that their omitted constituents need not share the same syntactic structure (example 8). On the other hand, there are structural constraints, as the incorrect example 9 demonstrates.

[3]Grammatically incorrect sentences are marked with '*' as usual in linguistics

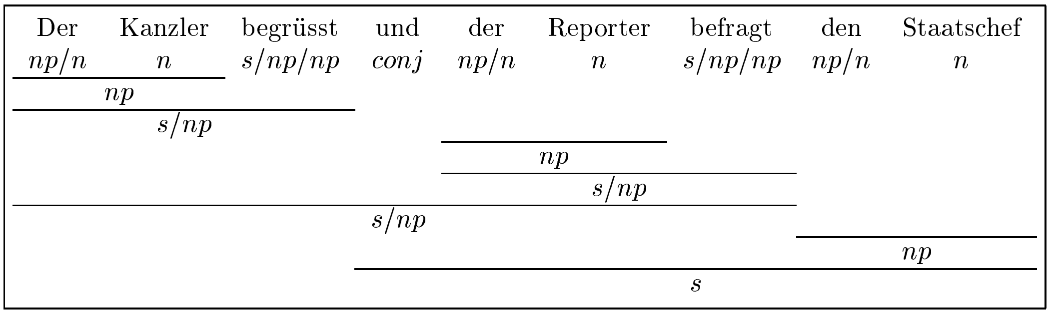

Figure 7: Processing of a left factored out ellipse using the general coordination rule

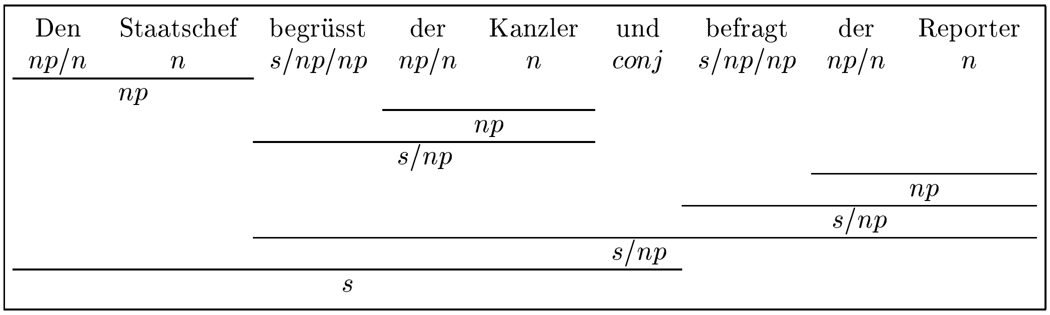

Figure 8: Processing of a right factored out ellipse using the general coordination rule

> **Example 8** *In den Wald ging der Jäger und fing einen Hasen ([Into the forest] the hunter went and ∅ caught a hare).*

> **Example 9** **In den Wald ging der Jäger und einen Hasen fing ([Into the forest] the hunter went and ∅ a hare caught).*

As figure 7 and 8 show, ellipsis factored out to the left and right are covered by the general coordination rule.

For gapping, Steedman suggests the "left conjunct revealing rule" which decomposes formerly reduced categories into several parts. Figure 9 shows one of Steedman's examples, where the previously parsed category s is decomposed into $(s\backslash np)/np$ and $s\backslash((s\backslash np)/np)$. Although the results of decomposition are considerably more complex than their input, Steedman leaves open where the necessary information for decomposition comes from ([Ste90]). Note that for processing the noun phrases to the right of the conjunction, type–raising must be used extensively with essentially the same problem. Information for both conjuncts must be deduced somehow from the context,

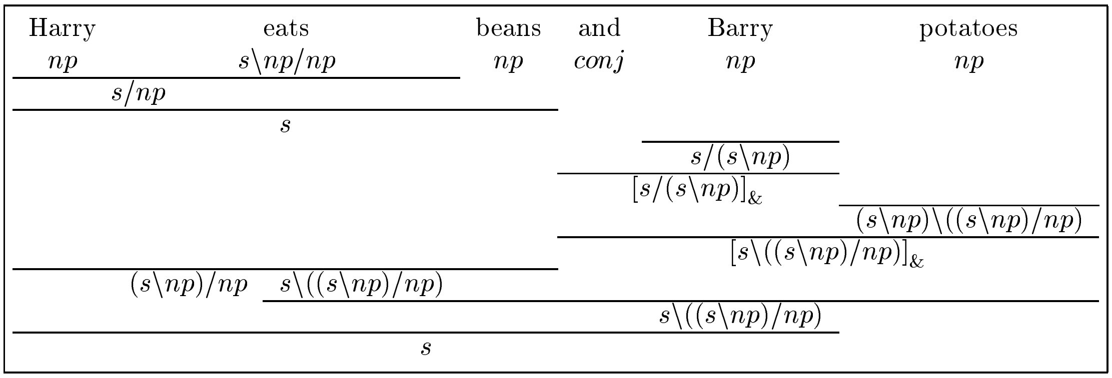

Figure 9: Application of Steedman's decomposition rule. Consistent with [Ste90], Steedman's split coordination rules and explicit type–raising are used.

without anything being said about its sources.

Another unsatisfactory property of decomposition is that by creating new categories it works contrary to the usual reduction rules. Finally, the category to be decomposed is the leftmost category in the sequence to be processed. In the common computational model of using a stack as a data structure for the parser, this element would not be accessible.

Steedman gives no evidence that human sentence processing connotes arbitrary reanalysis of information already understood. Apparently, humans don't need any backtracking when hearing a sentence like "Tom ate rice and Harry, fish". In our opinion, it triggers a symmetrical analysis in which the gap between "Harry" and "rice" is automatically filled with the previously memorised verb during sentence understanding.

To implement this mechanism, a short–term memory was introduced into an experimental CCG parser reflecting the most important yet not all features of the SFB grammar. In this memory, every category that seems useful for filling possibly forthcoming elliptic gaps is stored. If the parser cannot reduce the sentence successfully, the memory provides memorized categories as required by unsatisfied functions. Meaningless and unnecessary categories recalled by the memory are subsequently eliminated by failing application of reduction rules.

When using the memory, parsing of all above sentences in principle yields correct semantic interpretations. Unfortunately, the memory aggravates the problem of spurious ambiguities as it allows additional derivations. As an example, Figure 10 shows an alternative syntactic structure for the sentence of figure 8 in which the memory provides a missing category instead of using the

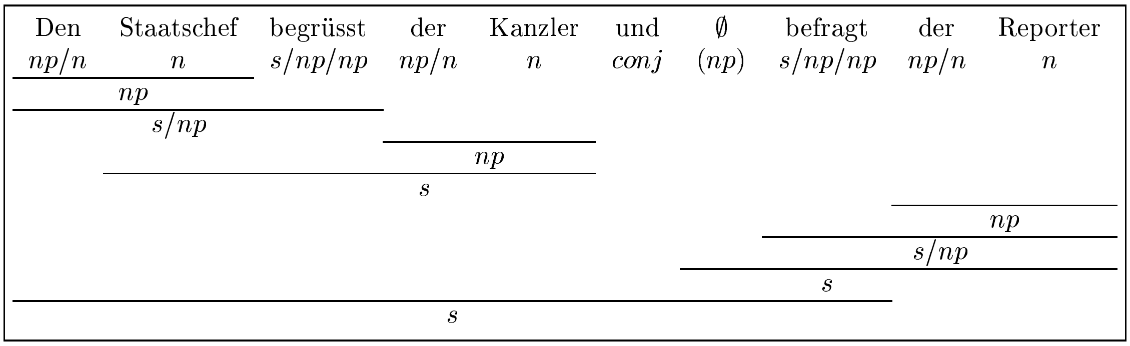

Figure 10: Derivation using the category memory

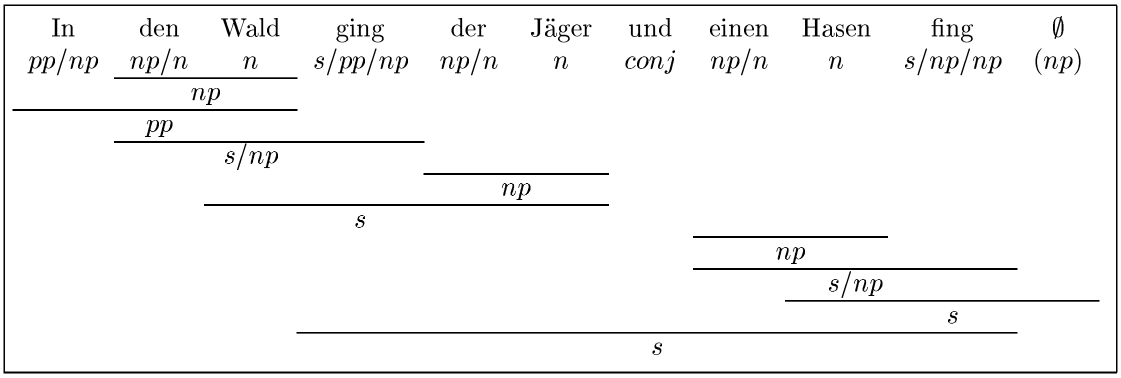

Figure 11: Undiserable yet possible derivation of an incorrect sentence

standard coordination rule. Both result in the same semantic interpretation.

Worse, the memory mechanism does not only generate more spurious ambiguities than before, but it also is too general to model German ellipses, as it incorrectly allows the derivation shown in figure 11. To reject this sentence as ungrammatical, further constraints are needed which restrict recalled categories to syntactically sensible ones. Chosen adequately, these contraints could also eliminate the problem of further spurious ambiguities.

Concerning some other seemingly incorrect sentences, one could argue that accepting them would mean a justifiable level of robustness, as oral language does not always comply with grammar. However, although spontaneous speech may contain sentences with unusual word order, no German native speaker would utter the one depicted in figure 11. Experiments must show how wide the scope of robustness should be before suitable constraints may be defined.

6 Efficiency issues

Conjunctions in general multipy the number of potential ambuigities. When processing simple sentences, shift–reduce conflicts may be resolved in favor of reduction, which leads to a higher degree of incrementality. In contrast, coordination may range over categories of different complexity, thus reduction often must be postponed until the constituent to be reduced has succesfully been conjoined. Figure 12 depicts conjunctions of varying complexity. When processed from left to right, there is no cue if or when a coordination starts until the conjunctions has been encountered.

Obviously, it is possible to proceed the other way round and to combine categories as soon as it is clear which conjunct will be needed. However, if you strive for incremental processing as a model of human understanding or to provide intermediate partial interpretations as soon as possible, this approach is hardly satisfying.

Our current solution is to track possibly coordinated constituents as soon as they emerge. The memory introduced for recalling gapped categories may be used for this purpose as well. When a conjunction is encountered, all alternatives are considered. The first few categories behind the conjunction determine the scope of coordination. Though this approach is not quite satisfactory, it has proven acceptable in practice.

So far, we only present data from parsing using the coordination rule instances. The memory structure, though powerful, is not yet sophisticated enough to be evaluated. With the discussed rules implemented, our parser yields an interpretation even of complicated sentences within some seconds. Simpler sentences, such as short imperatives, are much faster and can be processed in less than a second. These response times allow the system to be used in a language–controlled robotics environment such as the Baufix construction scenario of the SFB360.

7 Conclusion

In this paper, we first have shown how coordinations and some elliptic phenomena can be integrated into an existing German speech processing system using a unification–based variant of Combinatory Categorial Grammar. Apart from introducing a family of reduction rules, changes have to be made to the unification mechanism which otherwise would scramble semantics. Further, Steedman's dollar convention is generalised to allow triple occurences of the same arbitrarily long category.

In order to capture the whole range of German ellipses, a simple short–

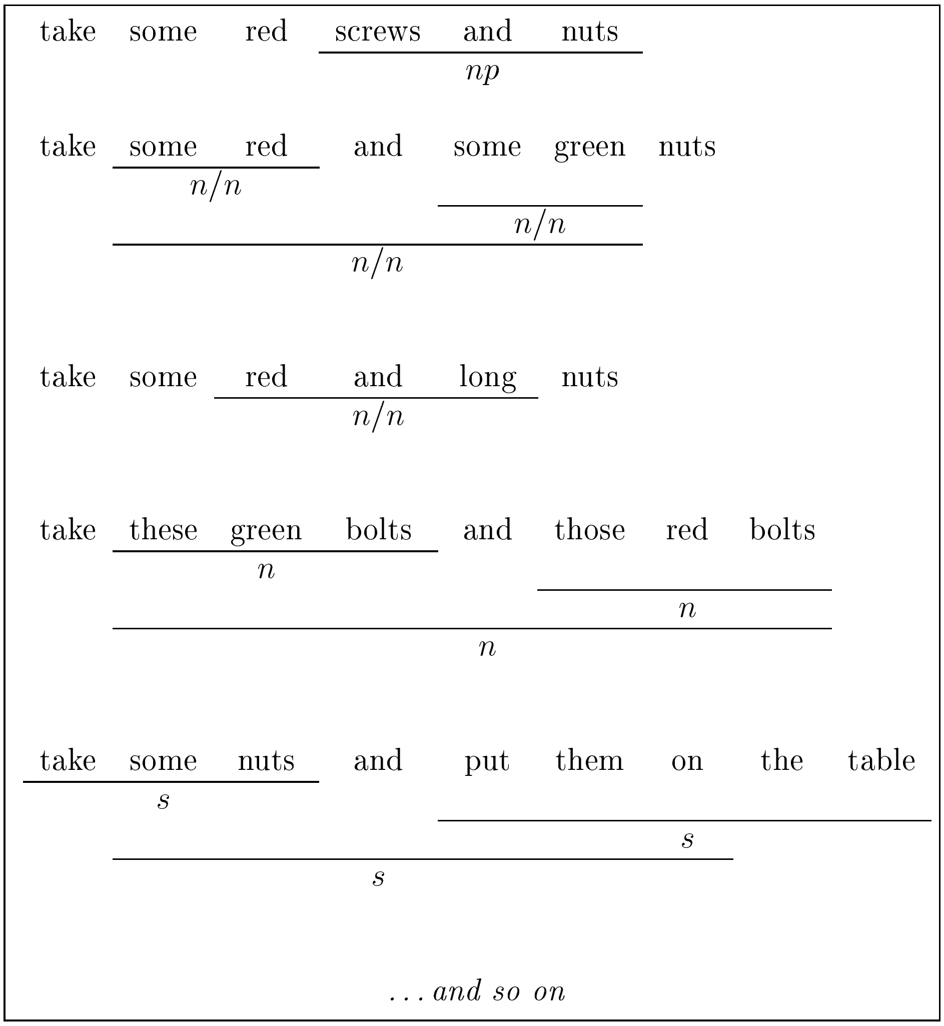

Figure 12: Varying complexity of conjuncts

time memory is added to the system. Although it has proven an acceptable performance even with complicated sentences, there are still some drawbacks to be addressed. First, the generative power must be constrained because it accepts elliptic sentences which are inacceptable in German grammar and allows unwanted alternative derivations. Second, a satisfactory mechanism has to be found which allows incremental processing of constituents without making their components inaccessible for coordination.

The memory model could be enhanced in several ways. So far, no pronouns are accepted, though the objects they could possibly denominate are already kept in mind. Pronouns do not only appear frequently in coordinated sentences but are also used as references across sentence boundaries. Thus, they would allow the step from single orders to longer sequences of instructions.

When the memory becomes extended to retain categories between sentences, it could handle adjacency ellipses such as minimal answers to Wh-questions which consist of single or few categories. This would enable dialogues between humans and the robot.

Finally, there are several classes of memory known in cognitive psychology. An inclusion of long–term, episodic and other kinds of memory would enhance the cognitive relevance of the SFB system considerably.

References

[CF58] Curry, H. & Feys, R. *Combinatory Logic, Vol. 1.* Amsterdam: North-Holland, 1953.

[GKS+93] Günther, U., Kindt, W., Schade, U., Sichelschmidt, M. & Strohner, H. *Elliptische Koordination. Strukturen und Prozesse lokaler Textkohärenz.* Linguistische Berichte 146: 311–328, Westdeutscher Verlag, 1993.

[HE99] Hildebrandt, B. & Eikmeyer, H.-J. *Sprachverarbeitung mit Combinatory Categorial Grammar: Inkrementalität und Effizienz.* Report 1999/05. Universität Bielefeld, 1999.

[JVW91] Joshi, A., Vijay-Shanker, K. & Weir, D. *The Convergence of Mildly Context-Sensitive Formalisms.* In P. Sells, S. Shieber & T. Wasow (Herausgeber): *Processing of Linguistic Structure*, 31–81, MIT Press, Cambridge MA, 1991.

[Kle93] Klein, W. *Ellipse.* In J.Jakobs, A. von Steckow: *Syntax. Ein internationales Handbuch zeitgenössischer Forschung.* 1. Halbband, Vol.1. 763–799, 1993.

[Niv94] Niv, M. *A Psycholinguistically Motivated Parser for CCG.* 32nd Annual Meeting of the Association for Computational Linguistics. Proceedings of the Conference, 27–30 June 1994, Las Cruces, New Mexico, USA. Morristown, NJ: ACL, 125–132, 1994.

[PS94] Pollard, C. & Sag, I. A. *Head–Driven Phrase Structure Grammar.* Chicago: University of Chicago Press, 1994.

[Shi86] Shieber, S. M. *An Introduction to Unification Based Grammar.* Center for the Study of Linguistics and Information, Stanford, 1986.

[Ste87] Steedman, M. *Combinatory Grammars And Human Language Processing.* In J. L. Garfield (Herausgeber): *Modularity In Knowledge Representation And Natural–Language Understanding*, 197–205, Cambridge, MA: MIT Press, 1987.

[Ste90] Steedman, M. *Gapping As Constituent Coordination.* Linguistics and Philosophy 13: 207–263, Kluwer Academic Publishers, 1990.

[Ste91] Steedman, M. *29th Annual Meeting of the Association for Computational Linguistics, Proceeding of the Conference, 18-21 June 1991, Berkley, California, USA*. Type-Raising and Directionality in Combinatory Grammar. Morristown, NJ: ACL, 1991

[Ste93] Steedman, M. *Categorial Grammar*. Lingua, 90, 221-258, 1993.

[Ste00] Steedman, M. *The Syntactic Process*. The MIT Press, Cambridge Mass, 2000.

[Vie00] Vierhuff, T. *Ein sprachverarbeitendes System zur Verarbeitung koordinativer Strukturen in der Combinatory Categorial Grammar*. Unveröffentlichte Diplomarbeit. Universität Bielefeld. 2000.

[Vie2001] Vierhuff, T. *Directional Unification for Processing Elliptic Coordinations in CCG*. 5. Fachtagung der Gesellschaft für Kognitionswissenschaft, 25.9.-28.9.2001, Max-Planck-Institut für Neuropsychologie, Leipzig