Head-Driven Phrase Structure Grammar, Sign-Based Construction Grammar, and Fluid Construction Grammar: Commonalities and differences

Stefan Müller
Arbeitsgruppe Deutsche Syntax
Philosophische Fakultät II
Institut für deutsche Sprache und Linguistik
Unter den Linden 6
10099 Berlin

St.Mueller@hu-berlin.de

November 2, 2017

Running title: HPSG, SBCG, and FCG: Commonalities and differences

Keywords: Construction Grammar, Fluid Construction Grammar, Sign-Based Construction Grammar, Nonlocal Dependencies, Competence/Performance distinction, discontinuous constituents, scope, linearization, argument structure

Abstract

Van Trijp (2013, 2014) claims that Sign-Based Construction Grammar (SBCG) and Head-Driven Phrase Structure Grammar (HPSG) are fundamentally different from Fluid Construction Grammar (FCG). He claims that the former approaches are generative ones while the latter is a cognitive-functional one. I argue that it is not legitimate to draw these distinctions on the basis of what is done in FCG. Van Trijp claims that there are differences in the scientific model, the linguistic approach, formalization, the way constructions are seen and in terms of processing. This paper discusses all these alleged differences. Van Trijp also claims that his cognitive-functional approach is superior in terms of completeness, explanatory adequacy, and theoretical parsimony. In order to facilitate a discussion and comparison, I introduce the reader to basic assumptions made in FCG and the analyses suggested by Van Trijp: I first deal with the representations that are used in FCG, talk about argument structure constructions, the combination operations fusion and merging that are used in FCG, I than discuss the analysis of nonlocal dependencies and show that the suggested FCG analysis is not explanatory adequate since it is not descriptively adequate and that a full formalization of approaches with discontinuous constituents is not more parsimonious than existing HPSG analyses either. After the discussion of specific analyses, I then provide a detailed comparison of FCG and SBCG/HPSG and discuss questions like the competence/performance distinction, mathematical formalization vs. computer implementation, fuzziness and fluidity in grammars, and permissiveness of theories. I conclude that HPSG, SBCG and FCG belong to the same family of theories and that all claims to the contrary are unjustified.

1 Introduction

This paper compares Head-Driven Phrase Structure Grammar (HPSG, Pollard & Sag 1987, 1994; Sag 1997) and Sign-based Construction Grammar (SBCG, Sag 2012) with Fluid Construction Grammar (FCG, Steels & De Beule 2006; Steels 2011). The comparison makes reference to papers by van Trijp (2013, 2014) in which he claims that HPSG and SBCG are fundamentally different from FCG. He claims that the former approaches are generative ones while the latter is a cognitive-functional one. I think that it is not legitimate to draw these distinctions on the basis of what is done in FCG.[1] Van Trijp claims that there are differences in the scientific model, the linguistic approach, formalization, the way constructions are seen and

[*] I thank Stephan Oepen for discussion of topics related to this paper. Over the years I had many intense discussions with Remi van Trijp, Luc Steels and other members of the FCG group. Thanks for all these discussions.

[1] Steels (2013, 153) emphasizes the point that FCG is a technical tool for implementing constructionist ideas rather than a theoretical framework of its own. However, authors working with the FCG system publish linguistic papers that share a certain formal background and certain linguistic assumptions. This paper addresses some of the key assumptions made and some of the mechanisms used in FCG as a framework in this sense.

in terms of theories of human performance. In order to show that it is not legitimate to draw a wedge between these rather similar constraint-based theories I compare some of the key concepts and assumptions in this paper. In Section 2, I compare the formal tools of FCG and HPSG. Section 3 deals with the approaches to argument structure constructions, Section 4 discusses the treatment of non-local dependencies, Section 5 discusses the competence/performance distinction and Section 6 discusses issues related to formalization and implementation of the theories under consideration. Section 7 discusses the overall scientific approach, which – in my opinion – HPSG/SBCG and FCG share. Section 8 discusses the permissiveness of theories and Section 9 concludes the paper.

2 General remarks on the representational format

Fluid Construction Grammar is similar to HPSG in that it uses attribute value matrices (AVMs) to represent linguistic objects. However, these AVMs are untyped. Since there are no types, there are no inheritance hierarchies that can be used to capture generalizations, but one can use macros to reach similar effects. Constructions can refer to more general constructions (van Trijp, 2013, 105). Every AVM comes with a name and can be depicted as follows:

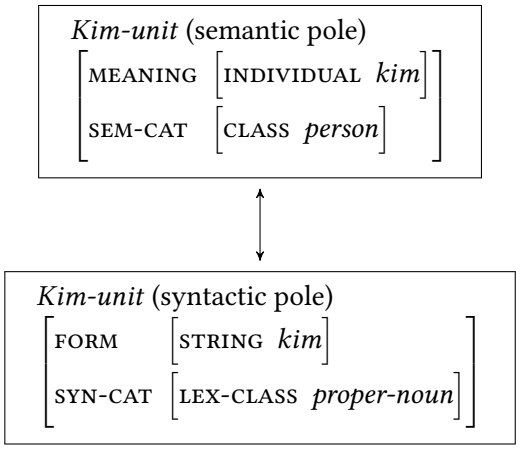
(1)
$$\boxed{\begin{array}{l}\textit{unit-name}\\ \begin{bmatrix}\textsc{feature}_1 & \textit{value}_1\\ \ldots & \\ \textsc{feature}_n & \textit{value}_n\end{bmatrix}\end{array}}$$

The HPSG equivalent would be the AVM in (2), where *type* is a type that is associated with the AVM:

(2)
$$\begin{bmatrix}\textit{type} & \\ \textsc{feature}_1 & \textit{value}_1\\ \ldots & \\ \textsc{feature}_n & \textit{value}_n\end{bmatrix}$$

Linguistic objects have a form and a meaning pole. The two poles could be organized into a single feature description by using a SYN and a SEM feature (see Pollard & Sag 1987; Sag 2012), but in FCG papers the two poles are presented separately and connected via a double arrow. (3) is an example:

(3) *Kim* according to van Trijp (2013, 99):



(4) shows how this can be recast with a SYN and a SEM feature:

(4) Using SYN and SEM to represent FCG's syntactic and semantic pole:

$$\left[\begin{array}{ll} \textsc{syn} & \left[\begin{array}{ll} \textsc{form} & \left[\textsc{string}\ \textit{kim}\right] \\ \textsc{syn-cat} & \left[\textsc{lex-class}\ \textit{proper-noun}\right] \end{array}\right] \\ \textsc{sem} & \left[\begin{array}{ll} \textsc{meaning} & \left[\textsc{individual}\ \textit{kim}\right] \\ \textsc{sem-cat} & \left[\textsc{class}\ \textit{person}\right] \end{array}\right] \end{array}\right]$$

In HPSG, information about the form is usually represented outside of the syntactic information under a feature named PHON, which is an abbreviation for PHONOLOGY. See Bird & Klein (1994), Orgun (1996), Höhle (1999), Walther (1999), Crysmann (2002, Chapter 6), and Bildhauer (2008) for phonology in the framework of HPSG.

Depending on the mode in which the lexical items are used, the syntactic pole or the semantic pole is used first in the computational FCG system. The first processing step is a matching phase in which it is checked whether the semantic pole (for production) or the syntactic pole (for parsing) matches the structure that was built so far.[2] After this test for unification, the actual unification, which is called merging, is carried out (see Section 3.1 on fusion, matching, and merging). After this step, the respective other pole (syntax for generation and semantics for parsing) is merged. This is illustrated in Figure 1 on the facing page.

[2]Some notes on terminology are in order here. According to van Trijp FCG does not make the performance/competence distinction. The computational implementation is supposed to reflect human behavior. So processing/parsing/production in the context of FCG refers to processing/parsing/production by both humans and computers. HPSG and SBCG make the competence/performance distinction and hence processing in computer implementations does not necessarily mirror processing by humans. When I talk about performance in the context of HPSG/SBCG I refer to processing by humans. When computer implementations are at issue I use the term *implementation*. See Sections 4.7 and 5 for discussion of the competence/performance distinction.

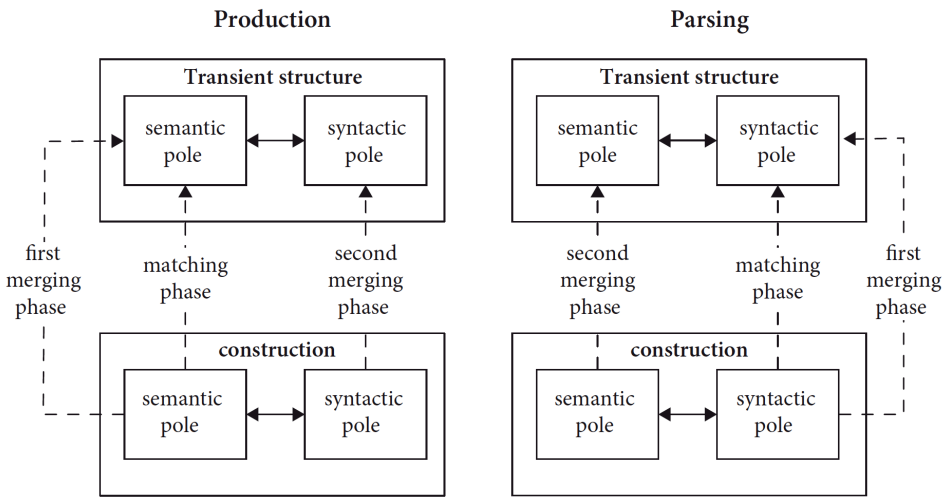


Figure 1: Production and parsing in FCG (van Trijp, 2013, 99)

3 Argument Structure Constructions: phrasal vs. lexical

Fluid Construction Grammar assumes a phrasal approach to argument structure, that is, it is assumed that lexical items enter into phrasal configurations that contribute independent meaning (van Trijp, 2011). The FCG approach is one version of implementing Goldberg's plugging approach to argument structure constructions (Goldberg, 1995). Van Trijp suggests that every lexical item comes with a representation of potential argument roles like Agent, Patient, Recipient, and Goal. Phrasal argument structure constructions are combined with the respective lexical items and realize a subset of the argument roles, that is they assign them to grammatical functions. Figure 2 on the next page shows an example: the verb *sent* has the semantic roles Agent, Patient, Recipient, and Goal (upper left of the figure). Depending on the argument structure construction that is chosen, a subset of these roles is selected for realization.[3] The figures show the relation between sender, sent, and sendee and the more abstract semantic roles and the relation between these roles and grammatical functions for the sentences in (5):

(5) a. He sent her the letter.

[3]It is interesting to note here that van Trijp (2011, 141) actually suggests a lexical account since every lexical item is connected to various phrasal constructions via coapplication links. So every such pair of a lexical item and a phrasal construction corresponds to a lexical item in Lexicalized Tree Adjoining Grammar (LTAG, Schabes et al. 1988). See also Müller & Wechsler (2014a, 25) on Goldberg's assumption that every lexical item is associated with phrasal constructions.

Note that such coapplication links are needed since without them the approach cannot account for cases in which two or more argument roles can only be realized together but not in isolation or in any other combination with other listed roles.

b. He sent the letter.

c. The letter was sent to her.

While in (5a) the agent, the patient and the recipient are mapped to grammatical functions, only the agent and the patient are mapped to grammatical functions in (5b). The recipient is left out. (5c) shows an argument realization in which the sendee is realized as a *to* phrase. According to van Trijp this semantic role is not a recipient but a goal.

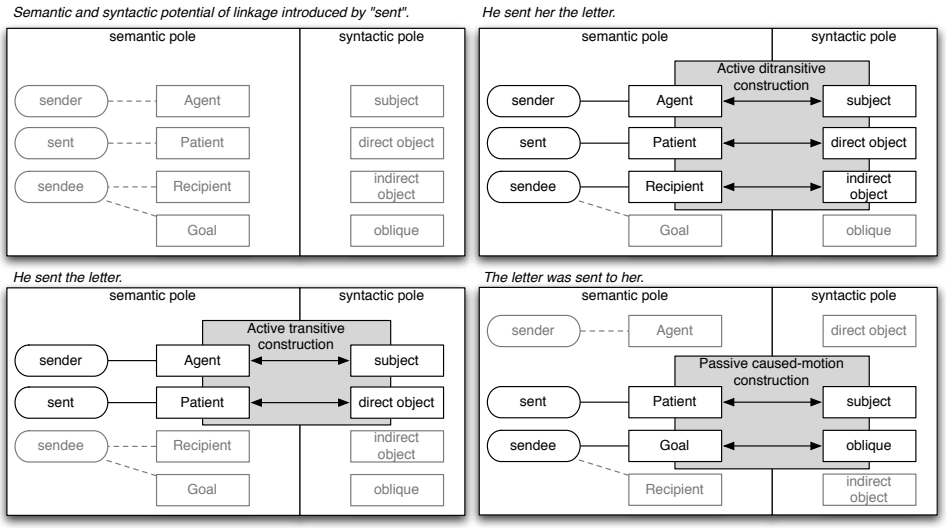


Figure 2: Lexical items and phrasal constructions. Figure taken from van Trijp (2011, 122)

Note that under such an approach, it is necessary to have a passive variant of every active construction. For languages that allow for the combination of passive and impersonal constructions, one would be forced to assume a transitive-passive-impersonal construction. As was argued in Müller (2006, Section 2.6) free datives (commodi/incommodi) in German can be added to almost any construction. They interact with the dative passive and hence should be treated as arguments. So, for the resultative construction one would need an active variant, a passive variant, a variant with dative argument, a variant with dative argument and dative passive, and a middle variant. While it is technically possible to list all these patterns and it is imaginable that we store all this information in our brains, the question is whether such listings really reflect our linguistic knowledge. If a new construction comes into existence, lets say an active sentence pattern with a nominative and two datives in German, wouldn't we expect that this pattern can be used in the passive? While proposals that establish relations between active and passive constructions would predict this, alternative proposals that just list the attested possibilities do not.

The issue of how such generalizations should be captured was discussed in

connection with the organization of the lexicon in HPSG (Flickinger, 1987; Meurers, 2001). In the lexical world, one could simply categorize all verbs according to their valence and say that *loves* is a bi-valent verb and the passive variant *loved* is an mono-valent verb. Similarly *gives* would be categorized as a ditransitive verb and *given* as a two-place verb. Obviously this misses the point that *loved* and *given* share something: they both are related to their active form in a systematic way. This kind of generalization is captured by lexical rules that relate two lexical items. The respective generalizations that are captured by lexical rules are called horizontal generalizations as compared to vertical generalizations, which describe relations between subtypes and supertypes in an inheritance hierarchy (Meurers, 2001, 161).

The issue is independent of the lexical organization of knowledge, it can be applied to phrasal representations as well. Phrasal constructions can be organized in hierarchies (vertical), but the relation between certain variants is not covered by this. The analog to the lexical rules in a lexical approach are GPSG-like metarules in a phrasal approach. So what seems to be missing in FCG is something that relates phrasal patterns, e.g., allostructions (Cappelle, 2006; Goldberg, 2014, 116).

3.1 Fusion, matching and merging

As was pointed out by Dowty (1989, 89–90), checking for semantic compatibility is not sufficient when deciding whether a verb may enter (or be fused with) a certain construction. The example is the contrast between *dine*, *eat*, and *devour*. While the thing that is eaten may not be realized with *dine*, its realization is optional with *eat* and obligatory with *devour*. So the lexical items have to come with some information about this.

Van Trijp (2011) and Steels & van Trijp (2011) make an interesting suggestion that could help here: every verb comes with a list of potential roles and argument structure constructions can pick subsets of these roles (see Figure 2). This is called *matching*: introducing new argument roles is not allowed. This would make it possible to account for *dine*: one could say that there is something that is eaten, but that no Theme role is made available for linking to the grammatical functions. This would be a misuse of thematic roles for syntactic purposes though since *dine* is semantically a two-place predicate. To account for the extension of argument roles as it is observed in the caused motion construction (Goldberg, 1995, Chapter 7), Steels & van Trijp (2011) suggest a process called *merging*. Merging is seen as a repair strategy: if an utterance involves a strictly intransitive verb and some other material, the utterance cannot be processed with matching alone. For example, when processing Goldberg's example in (6), *he sneezed* could be parsed, but *the foam* and *off the cappuccino* would be unintegrated.

(6) He sneezed the foam off the cappuccino.[4]

[4] Goldberg (2006, 42).

So, Steels & van Trijp (2011, 319–320) suggest that only if regular constructions cannot apply, merging is allowed. The problem with this is that human language is highly ambiguous and in the case at hand this could result in situations in which there is a reading for an utterance, so that the repair strategy would never kick in. Consider (7):[5]

(7) Schlag den Mann tot!
beat the man dead
'Beat the man to death!' or 'Beat the dead man!'

(7) has two readings: the resultative reading in which *tot* 'dead' expresses the result of the beating and another reading in which *tot* is a depictive predicate. The second reading is dispreferred, since the activity of beating dead people is uncommon, but the structure is parallel to other sentences with depictive predicates:

(8) Iß den Fisch roh!
eat the fish raw

The depictive reading can be forced by coordinating *tot* with a predicate that is not a plausible result predicate:

(9) Schlag ihn tot oder lebendig!
beat him dead or alive
'Beat him when he is dead or while he is alive!'

So, the problem is that (7) has a reading which does not require the invocation of the repair mechanism: *schlug* 'beat' is used with the transitive construction and *tot* is an adjunct (see Winkler 1997, 310). However, the more likely analysis of (7) is the one with the resultative analysis, in which the valence frame is extended by an oblique element. So this means that one has to allow the application of merging independent of other analyses that might be possible. As Steels & van Trijp (2011, 320) note, if merging is allowed to apply freely, utterances like (10a) will be allowed and of course (10b) as well.

(10) a. * She sneezed her boyfriend.
b. * She dined a steak.

In (10) *sneeze* and *dined* are used in the transitive construction.

The way out of this dilemma is to establish information in lexical items that specifies in which syntactic environments a verb can be used. This information can be weighted and for instance the probability of *dine* to be used transitively

[5] I apologize for these examples .... An English example that shows that there may be ambiguity between the depictive and the resultative construction is the following one that is due to Haider (2016):

(i) They cooked the chicken dry.

I use the German example below since the resultative reading is strongly preferred over the depictive one.

would be extremely low. Steels and van Trijp would connect their lexical items to phrasal constructions via so-called coapplication links and the strength of the respective link would be very low for *dine* and the transitive construction and reasonably high for *sneeze* and the caused-motion construction. This would explain the phenomena (and in a usage-based way), but it would be a lexical approach, as it is common in CG, HPSG, SBCG, and DG.

The alternative view is that lexical items are related via lexical rules as in HPSG/SBCG (Flickinger, 1987; Pollard & Sag, 1987). As is explained in Briscoe & Copestake (1999) lexical rules can be associated with probabilities. Briscoe & Copestake show how low ranked lexical rules can be used to license marked ditranistivizations as in (11):

(11) She smiled herself an upgrade.

So rather than having a clear-cut failure and repair, low ranked lexical rules take the function of repair rules. Provided the weights are set appropriately this nicely explains why the resultative reading is the preferred one in (7): the depictive reading is dispreferred for semantic reasons and hence the analysis involving the resultative lexical rule is ranked higher.

4 Long-distance dependencies

This section compares the analysis of nonlocal dependencies in HPSG and SBCG with the FCG analysis suggested by van Trijp (2014). Van Trijp (2014) claims that there are fundamental differences between SBCG and FCG and assigns SBCG to the class of generative grammars, while placing FCG in the class of cognitive-functional approaches. He claims that his cognitive-functional approach is superior in terms of completeness, explanatory adequacy, and theoretical parsimony (p. 2). I take up these three points in what follows. I first provide a sketch of the analyses of nonlocal dependencies in HPSG/SBCG and FCG in Section 4.1. I then discusses information structure in Section 4.2. Section 4.4 points out problems that the linearization-based FCG approach has with scope phenomena. Extraction path marking languages are discussed in Section 4.5, and Section 4.6 is devoted to Across the Board extraction in coordination, another area of grammar that is problematic for the FCG analysis. I show that covering all the phenomena that find a natural explanation in the SLASH-based accounts of GPSG and HPSG/SBCG will involve a large number of stipulations and a lot of additional machinery. Section 4.7 shows that the computational parsing procedure that van Trijp describes is not compatible with psycholinguistic results and hence his FCG account is not empirically adequate. Section 4.8 shows that adjacency constraints are needed to rule out certain strings. Without these constraints the FCG analysis would not be empirically adequate. With these constraints it is probably not simpler than the HPSG/SBCG analyses. Section 4.9 discusses further problems that arise if one allows discontinuous constituents in grammars. Section 4.10 concludes the part about nonlocal dependencies.

4.1 Sketch of the analyses

HPSG and SBCG use the SLASH mechanism that was developed in GPSG (Gazdar, 1981) for the analysis of nonlocal dependencies. In one variant of the analysis of (12) a trace is assumed at the position of the object in sentences without extraction and this trace is related to a filler at the beginning of the sentence.

(12) People like him$_i$, everybody knows I dislike $\__i$.

The information about the local properties of the object (NP with accusative case) are percolated up in the structure and finally bound off in a head-filler configuration that licenses the combination of a sentence with an extracted element (something in SLASH) and the fronted element that has to be compatible with what is missing from the sentence, that is, it has to be compatible with the information in SLASH. The analysis is sketched in Figure 3.[6]

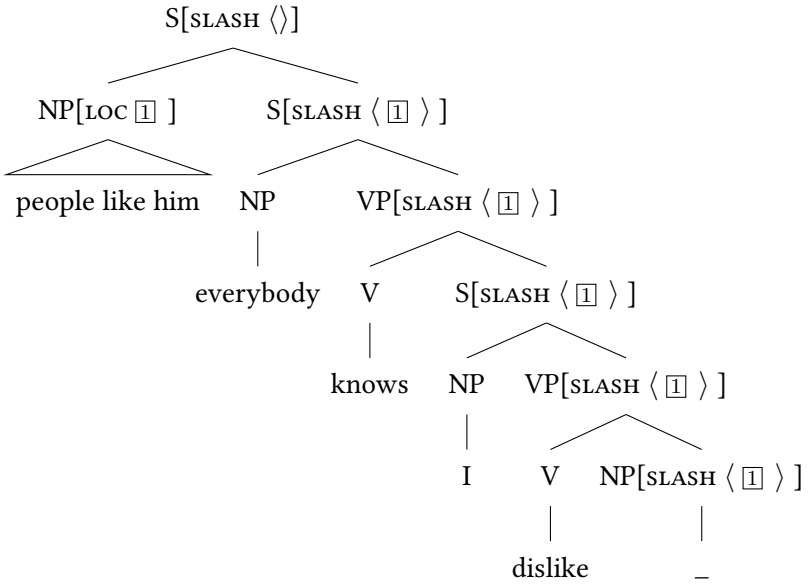


Figure 3: GPSG/HPSG analysis of nonlocal dependencies

What van Trijp (2014) suggests is basically an analysis that was suggested by Reape (2000) in unpublished work (see Reape (1994) for a published version of an linearization-based approach and Kathol (2000); Müller (1996, 1999a, 2002) for linearization-based approaches that despite of being linearization-based assume the SLASH approach for nonlocal dependencies). Van Trijp develops a model of grammar that allows for discontinuous constituents and just treats the serialization of the object in sentences like (13) as an alternative linearization option.

[6]Some versions of HPSG do not assume traces. They assume that the nonlocal dependency is introduced lexically by the lexical item for *dislike* (Bouma, Malouf & Sag, 2001). The figure for this analysis would be similar: the right-most NP would be missing and the respective SLASH information would be present at the V node dominating *dislike*.

(13) a. This book, I read.
b. What did the boy hit?

Van Trijp's analysis involves several units that do not normally exist in phrase structure grammars, but can be modeled via adjacency constraints or which represent relations between items which are part of lexical representations in HPSG/SBCG anyway. An example is the subject-verb anchor that connects the subject and the verb to represent the fact that these two items play an important functional role. Figure 4 shows the analysis of (14).

(14) What did the boy hit?

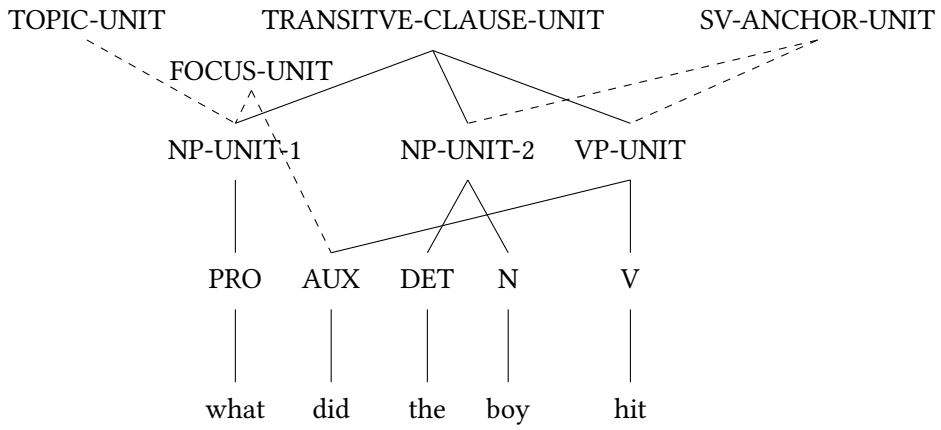


Figure 4: The analysis of *What did the boy hit?* according to van Trijp (2014, 265)

In the following subsections I discuss several aspects of Figure 4 in more detail. Subsection 4.2 deals with information structure and subsection 4.3 discusses the analysis of *do* support. Subsection 4.4 points out problems in accounting for scope and subsection 4.5 discusses challenges that extraction path marking languages would pose for analyses of the type suggested by van Trijp. As mentioned in the introduction of this section, the discussion will address van Trijp's claim that his cognitive-functional approach is superior in terms of completeness, explanatory adequacy, and theoretical parsimony (van Trijp, 2014, 2).

4.2 Information structure in FCG and HPSG/SBCG

As can be seen in the Figure 4, van Trijp refers to information structural terms like topic and focus while nothing is said about topic and focus in the HPSG analysis in Figure 3. It should be noted here that the analysis of information structure has quite some history in the framework of HPSG (Engdahl & Vallduví, 1996; Kuhn, 1995, 1996; Günther et al., 1999; Wilcock, 2005; De Kuthy, 2002; Paggio, 2005; Bildhauer, 2008; Bildhauer & Cook, 2010). The fact that information structure is not talked about in syntax papers like Sag (2012) does not entail that information

structure is ignored or should be ignored in theories like HPSG and SBCG. The authors mentioned above combine the analysis of nonlocal dependencies with information structure constraints. Since the difference between earlier versions of HPSG and SBCG are minor as far as nonlocal dependencies are concerned, all analyses of information structure carry over to SBCG. This shows that van Trijp's claims on lacking completeness are unwarranted. The same holds of course for explanatory adequacy: information structural constraints are not part of Sag's work but they have been formulated in various HPSG publications. This leaves us with theoretical parsimony, but before I comment on this, I want to discuss van Trijp's analysis in a little bit more detail in order to show that many of his claims are empirically problematic and that his theory therefore cannot be explanatory since empirical correctness is a precondition for explanatory adequacy.

Van Trijp claims that sentences with nonlocal dependency constructions in English start with a topic.[7] Bresnan (2001, 97) provides the following examples that show that the fronted elements are not necessarily topics:

(15) Q: What did you name your cat?
A: Rosie I named her. (*Rosie* = FOCUS)

(16) Q: What did you name your pets?
A: My dog, I named Harold. My cat, I named Rosie. (*my dog*, *my cat* = TOPIC)

So, a statement saying that the fronted element is a topic is empirically not correct. If the pre-subject position is to be associated with an information structural function, this association has to be a disjunction admitting both topics and focused constituents.

4.3 *do* support

A further problematic aspect of van Trijp's analysis is that he assumes that the auxiliary *do* is an object marker (p. 10, 22) or a non-subject marker (p. 23). It is true that *do* support is not necessary in subject questions like (17a), but only in (17b), but this does not imply that all items that are followed by *do* are objects.

(17) a. Who saw the man?
b. Who did John see?

First, *do* can be used to emphasize the verb:

(18) Who *did* see the man?

Second all types of other grammatical functions can precede the verb:

[7] Van Trijp (2014, 256) uses the following definitions for topic and focus: "Topicality is defined in terms of aboutness: the topic of an utterance is what the utterance is 'about'. Focality is defined in terms of salience: focus is used for highlighting the most important information given the current communicative setting."

(19) a. Where did you see the man? (adverbial)
b. How tall is the man? (predicative)
c. What did John consider Peter? (predicative)
d. What does this book cost? (adverbial)
e. About what did you talk? (prepositional object)

And finally, even a subject can appear in front of *do* if it is extracted from another clause:

(20) Who does he think saw this man? (subject)

4.4 Scope

There is a further empirical problem: approaches that assume that a filler is related to its origin can explain scope ambiguities that only arise when an element is extracted. Compare for instance the sentence in (21a) with the sentences in (21b, c): although the order of *oft* and *nicht* in (21a) and (21c) is the same, (21a) is ambiguous but (21c) is not.

(21) a. Oft liest er das Buch nicht.
often reads he the book not
'It is often that he does not read the book.' or 'It is not the case that he reads the book often.'
b. dass er das Buch nicht oft liest
that he the book not often reads
'that it is not the case that he reads the book often'
c. dass er das Buch oft nicht liest
that he the book often not reads
'that it is often that he does not read the book'

(21a) has the two readings that correspond to (21b) and (21c). A purely linearization-based approach probably has difficulties to explain this.[8] A SLASH-based approach can assume that (21a) has a gap (or some similar means for the introduction of nonlocal dependencies) at the position of *oft* in (21b) or (21c). The gap information is taken into account in the semantic composition at the site of the gap. This automatically accounts for the observed readings.

4.5 Extraction path marking

Another empirical problem that has to be solved is the existence of extraction path marking languages. Bouma, Malouf & Sag (2001) list a number of languages in which elements vary depending on the existence or absence of a gap in a constituent they attach to. For instance, Irish has complementizers that have one

[8] See also Müller (2016, Section 11.7.1) for a discussion of scope assignment in those versions of Dependency Grammar assuming that fronted elements are treated like normal dependents of a head.

form if the clause they attach to has an element extracted and another form if it does not. SLASH-based proposals can account for this in a straight-forward way: the fact that a constituent is missing in a phrase is represented in the SLASH value of the trace and this information is percolated up the tree. So even complex structures contain the information that there is a constituent missing inside them. Complementizers that are combined with sentences therefore can select sentences with SLASH values that correspond to the form of the complementizer. Van Trijp's answer to this challenge is that all languages are different (van Trijp, 2014, 263) and that the evidence from one language does not necessarily mean that the analysis for that language is also appropriate for another language. While I agree with this view in principle (see Müller 2015), I do think that extraction is a rather fundamental property of languages and that nonlocal dependencies should be analyzed in parallel for those languages that have it.

4.6 Coordination

One of the success stories of non-transformational grammar is the SLASH-based analysis of nonlocal dependencies by Gazdar (1981). This analysis made it possible for the first time to explain Ross's Across the Board Extraction (Ross, 1967). The following examples illustrate:

(22) a. The kennel which Mary made and Fido sleeps in has been stolen.
(= S/NP & S/NP)

b. The kennel in which Mary keeps drugs and Fido sleeps has been stolen.
(= S/PP & S/PP)

c. * The kennel (in) which Mary made and Fido sleeps has been stolen.
(= S/NP & S/PP)

The generalization is that two (or more) constituents can be coordinated if they have identical syntactic categories and identical SLASH values. This explains why *which* and *in which* in (22a,b) can fill two positions in the respective clauses. Now, theories that do not use a SLASH feature for the percolation of information about missing elements have to find different ways to make sure that all argument slots are filled and that the correct correspondence between extracted elements and the respective argument role is established. Note that this is not straightforward in models like the one suggested by van Trijp, since he has to allow the preposition *in* to be combined with some material to the left of it that is simultaneously also the object of *made*. Usually an NP cannot simply be used by two different heads as their argument. As an example consider (23a):

(23) a. * John said about the cheese that I like.

b. John said about the cheese that I like it.

If it would be possible to use material several times, a structure for (23a) would be possible in which *the cheese* is the object of the preposition *about* and of the

verb *like*. This sentence, however, is totally out: the pronoun *it* has to be used to fill the object slot.

4.7 Empirical adequacy: Discontinuous constituents and performance models

Van Trijp points out that SBCG does not have a performance model and contrasts this with FCG. On page 252 he states:

> So parsing starts by segmenting the utterance into discrete forms, which are then categorized into words by morphological and lexical constructions, and which can then be grouped together as phrases (see Steels, 2011b, for a detailed account of lexico-phrasal processing in FCG). So the parser will find similar constituents for all four utterances, as shown in examples (21–24). Since auxiliary-*do* in example (24) falls outside the immediate domain of the VP, it is not yet recognized as a member of the VP.
>
> All of these phrases are disconnected, which means that the grammar still has to identify the relations between the phrases. (van Trijp, 2014, 252)

In his (21)–(24), van Trijp provides several tree fragments that contain NPs for subject and object, the main verb and the auxiliary. The trees for his (24) are shown in Figure 5. He states that these tree fragments have to be combined in or-

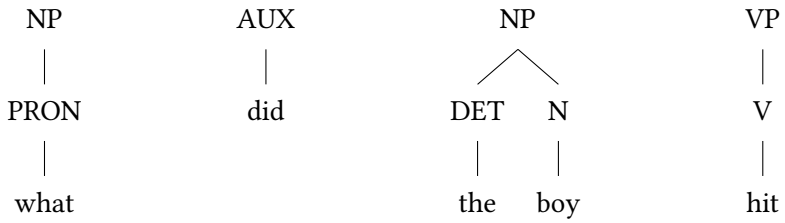


Figure 5: Tree fragments of *What did the boy hit?* according to van Trijp

der to analyze the sentences he discusses. This is empirically inadequate: if FCG does not make the competence/performance distinction, then the way utterances are analyzed should reflect the way humans process language (and this is what is usually claimed about FCG). However, all we know about human language processing points towards an incremental processing, that is, we process information from all linguistic levels as soon as it is available (Marslen-Wilson, 1975; Tanenhaus et al., 1995, 1996). We start to process the first word taking into account all of the relevant aspects (phonology, stress, part of speech, semantics, information structure) and come up with an hypothesis about how the utterance could proceed. As soon as we have two words processed (in fact even earlier: integration already happens during the processing of words) we integrate the second word

into what we know already and continue to follow our hypothesis, or revise it, or simply fail. So, we have to say that van Trijp's analysis fails on empirical grounds: his modeling of performance aspects is not adequate.

The parsing scheme that van Trijp describes is pretty much similar to those of computational HPSG parsers, but these usually come without any claims about human performance. Modeling human performance is rather complex since a lot of factors play a role. It is therefore reasonable to separate competence and performance and continue to work the way it is done in HPSG and FCG. This does not mean that performance aspects should not be modeled, in fact psycholinguistic models using HPSG have been developed in the past (Konieczny, 1996), but developing both a grammar with large coverage and the performance model that combines with it demands a lot of resources.

4.8 Parsimony: Discontinuity vs. Subject-Head and Head-Filler Schema

I now turn to parsimony: van Trijp uses a subject-verb anchor construction that combines the subject and the main verb. Because of examples like (24) it must be possible to have discontinuous subject-verb constructions:[9]

(24) Peter often reads books.

But if such constructions can be discontinuous one has to make sure that (25b) cannot be an instantiation of the subject-verb construction:

(25) a. The boy I think left.
b. * I the boy think left.

Here it is required to have some adjacency between the subject and the verb it belongs to, modulo some intervening adverbials. This is modelled quite nicely in phrase structure grammars that have a VP node. Whatever the internal structure of such a VP node may be, it has to be adjacent to the subject in sentences like (24) and (25a) above. This adjacency is enforced by the Subject-Head Schema, which combines a (projection of a) head with its subject. The dislocated element has to be adjacent to the complex consisting of subject and VP. This is what the Filler-Head Schema does in HPSG and SBCG. Van Trijp criticizes SBCG for having to stipulate such a schema (van Trijp, 2014, Section 4.1), but I cannot see how his grammar can be complete without a statement that ensures the right order of elements in sentences with fronted elements.

[9] Unless modals and tense auxiliaries are treated as main verbs (which they should not in English), constructions with modals seem to be another case where the subject and the main verb are not adjacent:

(i) a. Peter will read the book.
b. Peter has read the book.

Van Trijp stated that FCG differs from what he calls generative approaches in that it does not want to characterize only the well-formed utterances of a language. According to him, the parsing direction is much more liberal in accepting input than other theories. So it could well be that he is happy to find a structure for (25b). Note though that this is incompatible with other claims made by van Trijp: he argued that FCG is superior to other theories in that it comes with a performance model (or rather in not separating competence from performance at all). But then (25b) should be rejected both on competence and performance grounds. It is just unacceptable and speakers reject it for whatever reasons. Any sufficiently worked out theory of language has to account for this.

4.9 Empirical adequacy and parsimony: Restricting discontinuity

There is a further problem related to discontinuity. If one does not restrict continuity, then constituent orders like (26b) are admitted by the grammar:[10]

(26) a. Deshalb klärt, dass Peter kommt, ob Klaus spielt.
therefore resolves that Peter comes whether Klaus plays
'Therefore that Peter comes resolves whether Klaus will play.'

b. * Deshalb klärt dass ob Peter Klaus kommt spielt.
therefore resolves that whether Peter Klaus comes plays

The interesting thing about the word salad in (26b) is that the constituent order with in the *dass* clause and within the *ob* clause is correct. That is, the complementizer precedes the subject, which in turn precedes the verb. The problem is that the constituents of the two clauses are mixed.

In a model that permits discontinuous constituents, one cannot require that all parts of an argument have to be arranged after all parts that belong to another argument since discontinuity is used to account for nonlocal dependencies. So, it must be possible to have *Klaus* before other arguments (or parts of other arguments) since *Klaus* can be extracted. An example of mixing parts of phrases is given in (27):

(27) Dieses Buch hat der Mann mir versprochen, seiner Frau zu geben, der
this book has the man me promised his wife to give who
gestern hier aufgetreten ist.
yesterday here performed is
'The man who performed here yesterday promised me to give this book to his wife.'

We see that material that refers to *der Mann* 'the man', namely the relative clause *der gestern hier aufgetreten ist* 'who performed here yesterday', appears to the right. And the object of *geben* 'to give', which would normally be part of the phrase *dieses Buch seiner Frau zu geben* 'this book his wife to give' appears to the

[10] Again, see Müller (2016, Section 11.7.2.2) for a discussion of related problems for certain variants of Dependency Grammar.

left. So, in general it is possible to mix parts of phrases, but this is possible in a very restricted way only. Some dependencies extend all the way to the left of certain units (fronting) and others all the way to the right (extraposition). Extraposition is clause-bound, while extraction is not. In approaches like GPSG, HPSG and SBCG, the facts are covered by assuming that constituents for a complete clause are continuous apart from constituents that are fronted or extraposed. The fronted and extraposed constituents are represented in SLASH and EXTRA (Keller, 1995; Müller, 1999a, Section 13.2; Crysmann, 2013), respectively, rather than in valence features, so that it is possible to require of constituents that have all their valents saturated to be continuous (Müller, 1999b, 294).

Summing up the discussion of parsimony, it has to be said that van Trijp has to provide the details on how continuity is ensured. The formalization of this is not trivial and only after this is done can FCG be compared with the SLASH-based approach.

In addition to all the points discussed so far, there is a logical flaw in van Trijp's argumentation: he states that:

> whereas the filler-gap analysis cannot explain WHY *do*-support does not occur in *wh*-questions where the subject is assigned questioning focus, this follows naturally from the interaction of different linguistic perspectives in this paper's approach. (van Trijp, 2014, 263)

The issue here is whether a filler-gap analysis or an analysis with discontinuous constituents is suited better for explaining the data. A correct argumentation against the filler-gap analysis would require a proof that information structural or other functional constraints cannot be combined with this analysis. This proof was not provided and in fact I think it cannot be provided since there are approaches that integrate information structure. Simply pointing out that a theory is incomplete does not falsify a theory. This point was already made in my review of Boas (2003) and in a reply to Boas (2014). See Müller (2005a, 655–656), Müller (2007, Chapter 20), and Müller & Wechsler (2014b, Footnote 15).

4.10 Summary

The conclusion about the FCG analysis of nonlocal dependencies is that there are some empirical flaws that can be easily fixed or assumptions that can simply be dropped (role of *do* as object marker, claim that the initial position in English fronting construction is the topic), some empirical shortcomings (coordination, admittance of illformed utterances with discontinuous constituents), some empirical problems when the analysis is extended to other languages (scope of adjuncts in German), and the parsimony of the analyses is not really comparable since the restrictions on continuity are not really worked out (or at least not published). If the formalization of restrictions on continuity in FCG turns out to be even half as complex as the formalization that is necessary for accounts of nonlocal dependen-

cies (extraction and extraposition) in linearization-based HPSG (Kathol & Pollard, 1995; Reape, 2000; Wetta, 2011),[11] the SLASH-based analysis would be favorable.

In any case, I do not see how nonlocal dependencies could be used to drive a wedge between SBCG and FCG. If there are functional considerations that have to be taken into account, they should be modeled in both frameworks. In general, FCG should be more restrictive than SBCG since FCG claims to integrate a performance model, so both competence and performance constraints should be operative.

After concluding this section on nonlocal dependencies, I turn now to some more general reflections on the competence/performance distinction.

5 Competence/performance distinction

Van Trijp (2013, 112) claims that HPSG and SBCG are generative theories since they make the competence/performance distinction (Chomsky, 1965, Section 1.1). I think this use of the term *generative* is confusing: the term is usually used to describe approaches that assume that languages are sets of strings that are generated by some grammar. The alternative to such generative-enumerative approaches are constraint-based approaches and HPSG/SBCG belong to the latter class. The differences are explained in detail by Pullum & Scholz (2001).

Van Trijp contrasts approaches that separate competence and performance with cognitive-functional approaches. Concerning the latter van Trijp writes:

> The goal of a cognitive-functional grammar, on the other hand, is to explain how speakers express their conceptualizations of the world through language (= *production*) and how listeners analyze utterances into meanings (= *parsing*). Cognitive-functional grammars therefore implement both a competence and a processing model. (van Trijp, 2013, 90)

It is true that HPSG and SBCG make a competence/performance distinction (Sag & Wasow, 2011). HPSG theories are theories about the structure of utterances that are motivated by distributional evidence. These theories do not contain any hypothesis regarding brain activation, planning of utterances, processing of utterances (garden path effects) and similar things. In fact, most of the current publications on theories like Categorial Grammar, Dependency Grammar, Tree Adjoining Grammar, Lexical Functional Grammar, HPSG and Construction Grammar do not contain an explicit theory that explains all these things. I think that it is perfectly legitimate to work in this way: it is legitimate to study the structure of

[11] See Kathol & Pollard (1995) for a linearization-based account of extraposition. This account is implemented in the Babel System (Müller, 1996). See (Müller, 1999b) on restricting discontinuity. Linearization-based approaches were argued to not be able to account for apparent multiple frontings in German (Müller, 2005b, 2017) and hence linearizeation-based approaches were replaced by more traditional variants that allow for continuous constituents only.

words without studying their semantics and pragmatics, it is legitimate to study phonology without caring about syntax, it is legitimate to deal with specific semantic problems without caring about phonology and so on, provided there are ways to integrate the results of such research into a bigger picture. In comparison, it is wrong to develop models like those developed in current versions of Minimalism (called Biolinguistics), where it is assumed that utterances are derived in phases (NPs, CPs, depending on the variant of the theory) and then shipped to the interfaces (spell out and semantic interpretation, Chomsky 2008). As was discussed above, this is not what humans do.[12] But if we are neutral with respect towards such issues, we are fine. In fact, there is psycholinguistic work that couples HPSG grammars to performance models (Konieczny, 1996) and similar work exists for TAG (Shieber & Johnson, 1993; Demberg & Keller, 2008) and other frameworks.

Finally, there is also work in Construction Grammar that abstracts away from performance considerations. For instance, Adele Goldberg's book from 1995 does not contain a worked out theory of performance facts. It contains boxes in which grammatical functions are related to semantic roles. So this basically is a competence theory as well. Of course there are statements about how this is connected to psycholinguistic findings, but this is also true for theories like HPSG, SBCG and Simpler Syntax (Jackendoff, 2011, 600) that explicitly make the competence/performance distinction.

6 Theoretical framework vs. implementation platform

This section deals with implementation issues; that is, I discuss the question of how computers can be used to verify linguistic theories. I first discuss formalization vs. implementation and show how SBCG can be implemented in one of the systems that is used for processing HPSG grammars. Section 6.2 discusses the static constraint system of HPSG and compares it with the fluid system of FCG. Section 6.3 argues for a clean separation of linguistic theory and processing system.

6.1 Mathematical formalization vs. implementation

Van Trijp argues that SBCG has a mathematical formalization, while the formalization of FCG is computational. The difference between mathematical and computational formalization is a rather strange distinction to make. I think that a formal and precise description is a prerequisite for implementation (see Müller 2015, Section 1.2). Apart from this, a computer implementation of SBCG is trivial, given the systems that we have for processing HPSG grammars. In order to show this, I want to address one issue that van Trijp discusses. He claims that SBCG

[12] See also Labelle (2007) on the implausibility of such models. Attempts to integrate current Minimalist theories with psycholinguistic findings (Phillips, 2003) are incompatible with core principles of Minimalism like the *No Tampering Condition* of Chomsky (2008).

cannot be directly implemented. On issues of complexity of constraint solving systems he quotes (Levine & Meurers, 2006, Section 4.2.2):

> Actual implementations of HPSG typically handle the problem by guiding the linguistic processor using a (rule-based) phrase structure backbone, but the disadvantage of this approach is that the "organization and formulation of the grammar is different from that of the linguistic theory" (Levine & Meurers, 2006, Section 4.2.2). (van Trijp, 2013, 108)

He concludes:

> Applying all these observations to the operationalization of SBCG, we can conclude that an SBCG grammar is certainly amenable for computational implementation because of its formal explicitness. There are at least two computational platforms available, mostly used for implementing HPSG-based grammars, whose basic tenets are compatible with the foundations of SBCG: LKB (Copestake, 2002) and TRALE (Richter 2006). However, none of these platforms supports a 'direct' implementation of an SBCG grammar as a general constraint system, so SBCG's performance-independence hypothesis remains conjecture until proven otherwise.

There are two issues that should be kept apart here: efficiency and faithfulness to the theory. First, as Levine and Meurers point out, there were many constraint solving systems at the beginning of the 90's. So there are computer systems that can and have been used to implement and process HPSG grammars. This is very valuable since they can be used for direct verification of specific theoretical proposals. As was discussed by Levine and Meurers, trying to solve constraints without any guidance is not the most efficient way to deal with the parsing/generation problem. Therefore, additional control-structure was added. This control structure is used for instance in a parser to determine the syntactic structure of a phrase and other constraints will apply as soon as there is sufficient information available for them to apply. For instance, the assignment of structural case happens once the arguments of a head are realized. Now, is it bad to have a phrase structure backbone? One can write down phrase structure grammars that use phrase structure rules that have nothing to do with what HPSG grammars usually do. The systems TRALE (Meurers, Penn & Richter, 2002; Penn, 2004) and LKB will process them. But one is not forced to do this. For instance, the grammars that I developed for the CoreGram project (Müller, 2013a, 2015) are very close to the linguistic theory. To see that this is really the case, let us look at the Head-Argument Schema. The Head-Argument Schema is basically the type *head-argument-phrase* with certain type constraints that are partly inherited from its supertypes. The type with all the constraints is given here as (28):

(28) (syntactic) constraints on *head-argument-phrase*:

$$\begin{bmatrix} \textit{head-argument-phrase} \\ \textsc{synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \end{bmatrix} \\ \textsc{head-dtr|synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \oplus \langle\, \boxed{3} \,\rangle \end{bmatrix} \\ \textsc{non-head-dtrs} \langle\, [\ \textsc{synsem}\ \boxed{3}\ ] \,\rangle \end{bmatrix}$$

This can be translated into phrase structure grammar rules with complex symbols in a straight-forward way:

(29) a.
$$\begin{bmatrix} \textit{head-argument-phrase} \\ \textsc{synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \end{bmatrix} \\ \textsc{head-dtr}\ \boxed{4}\ \textsc{|synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \oplus \langle\, \boxed{3} \,\rangle \end{bmatrix} \\ \textsc{non-head-dtrs} \langle\, \boxed{5}\ [\ \textsc{synsem}\ \boxed{3}\ ] \,\rangle \end{bmatrix} \rightarrow \boxed{4}, \boxed{5}$$

b.
$$\begin{bmatrix} \textit{head-argument-phrase} \\ \textsc{synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \end{bmatrix} \\ \textsc{head-dtr}\ \boxed{4}\ \textsc{|synsem|loc|cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \oplus \langle\, \boxed{3} \,\rangle \end{bmatrix} \\ \textsc{non-head-dtrs} \langle\, \boxed{5}\ [\ \textsc{synsem}\ \boxed{3}\ ] \,\rangle \end{bmatrix} \rightarrow \boxed{5}, \boxed{4}$$

The left hand side of the rule is the mother node of the tree, that is, the sign that is licensed by the schema provided that the daughters are present. The right hand side in (29a) consists of the head daughter $\boxed{4}$ followed by the non-head daughter $\boxed{5}$. We have the opposite order in (29b), that is, the head daughter follows the non-head daughter. The two orders correspond to the two orders that are permitted by LP-rules: the head precedes its argument if it is marked INITIAL+ and it follows it if it is marked INITIAL−.

The following code shows how (29b) is implemented in TRALE:

```
arg_h rule (head_argument_phrase,
            synsem:loc:cat:head:initial:minus,
            head_dtr:HeadDtr,
            non_head_dtrs:[NonHeadDtr]
           )
  ===>
cat> NonHeadDtr,
cat> HeadDtr.
```

A rule starts with an identifier that is needed for technical reasons like displaying intermediate structures in the parsing process in debugging tools. A description of the mother node follows and after the arrow we find a list of daughters, each introduced by the operator `cat>`.[13] Structure sharing is indicated by values with capital letters. The above TRALE rule is a computer-readable variant of (29b) additionally including the explicit specification of the value of INITIAL.

Now, the translation of a parallel schema using a MOTHER feature like (30a) into a phrase structure rule is almost as simple:

(30) a.
$$\begin{bmatrix} \textit{head-argument-cx} \\ \textsc{mother}|\textsc{synsem}|\textsc{loc}|\textsc{cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \end{bmatrix} \\ \textsc{head-dtr}|\textsc{synsem}|\textsc{loc}|\textsc{cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \oplus \langle\, \boxed{3} \,\rangle \end{bmatrix} \\ \textsc{non-head-dtrs} \langle\, [\ \textsc{synsem}\ \boxed{3}\ ] \,\rangle \end{bmatrix}$$

b. $\boxed{6} \rightarrow \boxed{4}, \boxed{5}$ where
$$\begin{bmatrix} \textit{head-argument-cx} \\ \textsc{mother}\ \boxed{6}\ |\textsc{synsem}|\textsc{loc}|\textsc{cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \end{bmatrix} \\ \textsc{head-dtr}\ \boxed{4}\ |\textsc{synsem}|\textsc{loc}|\textsc{cat} \begin{bmatrix} \textsc{head} & \boxed{1} \\ \textsc{subcat} & \boxed{2} \oplus \langle\, \boxed{3} \,\rangle \end{bmatrix} \\ \textsc{non-head-dtrs} \langle\, \boxed{5}\ [\ \textsc{synsem}\ \boxed{3}\ ] \,\rangle \end{bmatrix}$$

(30b) is only one of the two phrase structure rules that correspond to (30a), but since the other one only differs from (30b) in the ordering of $\boxed{4}$ and $\boxed{5}$, it is not given here.

For grammars in which the order of the elements corresponds to the observable order of the daughters in a DTRS list, the connection to phrase structure rules is even simpler:

(31) $\boxed{1} \rightarrow \boxed{2}$ where
$$\begin{bmatrix} \textit{construction} \\ \textsc{mother} & \boxed{1} \\ \textsc{dtrs} & \boxed{2} \end{bmatrix}$$

The value of DTRS is a list and hence $\boxed{2}$ stands for the list of daughters on the right hand side of the phrase structure rule as well. The type *construction* is a supertype of all constructions and hence (31) can be used to analyze all phrases that are licensed by the grammar. In fact, (31) is one way to put the meta constraint in (32) that is assumed by Sag, Wasow & Bender (2003, 478):

[13] Other operators are possible in TRALE. For instance, `sem_head` can be used to guide the generator. This is control information that has nothing to do with linguistic theory and not necessarily with the way humans process natural language. There is also a `cats` operator, which precedes lists of daughters. This can be used to implement phrase structures with more than one non-head daughter.

(32) Φ is a Well-Formed Structure according to a grammar G if and only if:

1. there is a construction C in G, and
2. there is a feature structure I that is an instantiation of C, such that Φ is the value of the MOTHER-feature of I.

Φ corresponds to $\boxed{1}$ in (31).

So, this shows that the version of SBCG that has been developed by Sag (2012) has a straightforward implementation in TRALE.[14] The question remains whether "SBCG's performance-independence hypothesis remains conjecture until proven otherwise" as van Trijp sees it. The answer is: it is not a conjecture since any of the old constraint-solving systems of the nineties could be used to process SBCG grammars. The question of whether this is efficient is an engineering problem that is entirely irrelevant for theoretical linguistics. Theoretical linguistics is concerned with human languages and how they are processed by humans. So whether some processing system that does not make any claims about human language processing is efficient or not is absolutely irrelevant. Phrase structure-based backbones are therefore irrelevant as well, provided they refer to the grammar as described in theoretical work.

Now, this begs the question of whether there is a contradiction in my claims. In Müller (2013b, 252), I pointed out that SBCG is lacking a formalization in Richter's framework (Richter, 2004). Richter and also Levine & Meurers (2006) pointed out that there are problems with certain theoretically possible expressions and it is these expressions that mathematical linguists care about. So the goal is to be sure that any HPSG grammar has a meaning and that it is clear what it is. Therefore, this goal is much more foundational than writing a single grammar for a particular fragment of a language. There is no such foundational work for FCG since FCG is a specific toolkit that has been used to implement a set of grammars.

6.2 Static constraints vs. dynamic mappings and signature + grammar vs. open-endedness

On very interesting feature of Fluid Construction Grammar is its fluidity, that is there are certain constraints that can be adapted if there is pressure, the inventory of the theory is open-ended, so categories and features can be added if need be.

Again, this is not a fundamental difference between HPSG/SBCG and FCG. An HPSG grammar fragment of a specific language is a declarative representation of linguistic knowledge and as such it of course just represents a certain fragment and does not contain any information how this set of constraints evolved or how it is acquired by speakers. For this we need specific theories about language evolution/language change/language acquisition. This is parallel to what was said about the competence/performance distinction, in order to account for language

[14] A toy fragment of English using a MOTHER feature and phrase structure rules with specifications of the kind given above can be downloaded at `https://hpsg.hu-berlin.de/Fragments/SBCG-TRALE/`.

evolution we would have to have several HPSG grammars and say something about how one developed from the other. This will involve weighted constraints, it will involve recategorization of linguistic items and lots more.[15] So basically HPSG has to be extended, has to be paired with a model about language evolution in the very same way as FCG is.

6.3 A note on engineering

A problematic property of work done in FCG is that linguistic and engineering aspects are mixed.[16] Certain bookkeeping features that are needed only for technical reasons appear in linguistic papers, technical assumptions that are made to get a parser running are mixed with linguistic constraints. Bit vector encodings that are used to represent case information are part of papers about interesting case systems. There is certainly nothing wrong with bit vector encodings. They are used in HPSG implementations as well (Reape, 1991, 55; Müller, 1996, 269), but this is not mixed into the theoretical papers.

It was a big breakthrough in the 80's when theoretical linguists and computational linguists started working together and developed declarative formalisms that were independent of specific parsers and processing systems. This made it possible to take over insights from a lot of linguists who were not concerned with the actual implementation but took care of finding linguistic generalizations and specifying constraints. Since this separation is given up in FCG work, it will remain an engineering project without much appeal to the general linguist.

7 Overall approach: Theoretical physics vs. Darwinian evolutionary theory

Van Trijp compares SBCG and FCG and claims that SBCG follows the model of theoretical physics – like Chomsky does, while FCG adopts a Darwinian model of science – like Croft does, the difference being that SBCG makes certain assumptions that are true of all languages, while FCG does not make any a priori assumptions. The fundamental assumptions made in both theories are that the objects that we model are best described by feature value pairs (a triviality). FCG assumes that there is always a syntactic and a semantic pole (fundamental assumption in the system) and researchers working in HPSG/SBCG assume that if

[15] We had a simple version of weighted constraints in the German HPSG grammar that was developed in the Verb*mobil* project (Müller & Kasper, 2000) already. Further theoretical approaches to integrate weighted constraints are Brew (1995) and more recently Guzmán Naranjo (2015). Usually such weighted constraints are not part of theoretical papers, but there are exceptions as for instance the paper by Briscoe and Copestake about lexical rules (Briscoe & Copestake, 1999).

[16] This is not a problem if all FCG papers are read as papers documenting the FCG-system (see Footnote 1 on page 2) since then it would be necessary to include these technical details. If the FCG papers are to be read as theoretical linguistics papers that document a certain Construction Grammar analysis, the Lisp statements and the implementational details are simply an obstacle.

languages have certain phenomena, they will be analyzed in similar ways. For instance, if a language has nonlocal dependencies, these will be analyzed via the SLASH mechanism. However, this does not entail that one believes that grammars of all languages have a SLASH feature. And in fact, there may even be languages that do not have valence features (Koenig & Michelson, 2010), which may be a problem for FCG since it relies on the SYN-pole for the matching phase. So as far as SBCG is concerned, there is considerable freedom to choose features that are relevant in an analysis, and of course additional features and types can be assumed in case a language is found that provides evidence for this. The only example of a constraint provided by van Trijp that is possibly too strong is the locality constraint imposed by the MOTHER feature: the feature geometry of HPSG was revised in a way that makes it impossible to refer to daughters of daughters since daughters are parts of constructions but not of signs. (33a) shows how schemata look like in constructional HPSG (Sag, 1997) and (33b) shows the respective representation in SBCG:

(33) a.
$$\begin{bmatrix} \textit{sign} & \\ \textsc{phon} & \textit{list of phonemes} \\ \textsc{synsem} & \textit{synsem} \\ \textsc{dtrs} & \textit{list of signs} \end{bmatrix}$$

b.
$$\begin{bmatrix} \textit{construction} & \\ \textsc{mother} & \textit{sign} \\ \textsc{daughters} & \textit{list of signs} \end{bmatrix}$$

The idea about locality is that everything that is of relevance in a more nonlocal context has to be passed up explicitly rather than being selected as a property of a daughter's daughter or a daughter's daughter's daughter. Such passing up is done for nonlocal dependencies (via SLASH) and for instance also for information concerning the form of a preposition in a PP (via PFORM (Pollard & Sag, 1994, 23) or more recently via FORM (Sag, 2012, Section 3.1)). Certain verbs require prepositional objects and restrict the form of the preposition. For instance, *wait* has to make sure that its prepositional object has the preposition *for* in it. Since this information is usually available only at the preposition, it has to be passed up to the PP level in order to be directly selectable by the governing verb.

(34) I am waiting for my man.

So, assuming strict locality of selection requires that all phenomena that cannot be treated locally have to be analyzed by passing information up. Assuming strict locality is a design decision that does not have any empirical consequences, as far as it does not rule out any language or construction in principle. It just requires that information has to be passed up that needs to be accessed at higher nodes. As I showed in Müller & Wechsler (2014b), the locality constraint is easily circumvented even within SBCG and it makes the analysis of idioms unnecessar-

ily complicated and unintuitive, so I suggest dropping the MOTHER feature.[17] But even if MOTHER is kept, it is not justified to draw a distinction between SBCG and FCG along the lines suggested by van Trijp.

Independent of the MOTHER issue, the work done in the CoreGram project (Müller, 2013a, 2015) shows that one can derive generalizations in a bottom-up fashion rather than imposing constraints on grammars in a top-down way. The latter paper discusses Croft's methodological considerations and shows how methodological pitfalls are circumvented in the project. HPSG/SBCG research differs from work in Chomskyan frameworks in not making strong assumptions regarding underlying orders from which other orders are derived. An example of a theory that makes such strong assumptions is the work by Kayne (1994), where it is assumed that all languages have an underlying Specifier-Head-Complement order. While such approaches usually work well for SVO languages like English and Romance languages, they are problematic for SOV languages like German (Haider, 2000). Instead of introducing an anglocentric bias into the theory development, languages are treated on their own as it is common in the Construction Grammar community. This also has the advantage of being compatible with theories of language acquisition that do not assume innate linguistic knowledge. Such a view on language does not imply that there is no interest in generalizations and universals or near universals or tendencies, but again the style of working and the rhetoric in HPSG/SBCG is usually different from the ones in Mainstream Generative Grammar. Therefore, I think that the purported difference between SBCG and FCG does not exist.

8 Permissiveness of the theories

Van Trijp claims that HPSG/SBCG is a "generative grammar" since its aim is to account for and admit only grammatical sentences. FCG on the other hand is more permissive and tries to get the most out of the input even if it is fragmentary or ungrammatical (see also Steels, 2013, 166). While it is an engineering decision to be able to parse ungrammatical input – and there are most certainly systems for the robust processing of HPSG grammars (Kiefer, Krieger & Nederhof, 2000; Copestake, 2007), it is also clear that humans cannot parse everything. There are strong constraints whose violations cause measurable effects in the brain. This is something that a model of language (that includes competence and performance factors or does not make the difference at all) has to explain. The question is what the cause of deviance is: is it processing complexity? Is it a category mismatch? A clash in information structure? So, if FCG permits structures that are not accepted by human native speakers and that do not make any sense whatsoever, additional constraints have to be added. If they are not added, the respective FCG theory is not an adequate theory of the language under consideration. Again, there is no

[17] For a more general comparison of the feature geometries of constructional HPSG (Sag, 1997) and SBCG see Müller (2016, Section 10.6.2).

difference between HPSG/SBCG and FCG.

9 Conclusion

Van Trijp discusses the alleged differences between the HPSG variant Sign-Based Construction Grammar and Fluid Construction Grammar in several papers. He argues that HPSG/SBCG is fundamentally different from FCG in various ways and claims that FCG is superior in terms of completeness, explanatory adequacy, and theoretical parsimony. Van Trijp criticizes HPSG/SBCG for making the competence/performance distinction and compares concrete proposals regarding the analysis of nonlocal dependencies. I showed in this paper that his analysis of nonlocal dependencies is lacking a lot of constraints and that it is not trivial to fill in the missing constraints. Hence the claim of theoretical parsimony is not supported by his papers. On the contrary, analyses of the type he suggests are known in the HPSG world for more than 20 years now. They are well understood, have been well-formalized and implemented and criticized for their empirical shortcomings.

Furthermore I pointed out some of the shortcomings of his analyses showing that these analyses cannot be descriptively adequate from a competence point of view (the only thing that we can compare here if we ignore performance models of HPSG).

The properties of human language that are usually covered by performance theories and that should be covered by theories that do not make the competence/performance distinction are neither covered in van Trijp's analyses nor are they covered in the computer implementations. Hence, the FCG grammars are not observationally adequate, let alone descriptively or explanatorily adequate. If the claim is dropped that the computer implementation is a model of human performance, FCG theories would be competence theories like HPSG/SBCG theories.

I have also pointed out that one very nice property of the FCG system is that it uses weighted constraints and has certain aspects of fluidity. Similar tools are used in computational HPSG systems and I think that the combination of statistical information and linguistic information is something that is really needed if we want to arrive at a broad picture of human language. FCG has the required machinery in place already and interesting work can be done with the system in the areas of language evolution (Wellens, van Trijp, Beuls & Steels, 2013). I think that HPSG/SBCG and FCG are part of the same enterprise and those who worked in these frameworks until now focused on slightly different perspectives. I do not understand why one would wish to show that these two theories are radically different and drive a wedge between the respective communities.

References

Abney, Steven P. & Erhard W. Hinrichs (eds.). 1995. *Proceedings of the Seventh Conference of the European Chapter of the Association for Computational Linguistics*. Dublin: Association for Computational Linguistics.

Bildhauer, Felix. 2008. *Representing information structure in an HPSG grammar of Spanish*: Universität Bremen Dissertation.

Bildhauer, Felix & Philippa Helen Cook. 2010. German multiple fronting and expected topic-hood. In Stefan Müller (ed.), *Proceedings of the 17th International Conference on Head-Driven Phrase Structure Grammar, Université Paris Diderot*, 68–79. Stanford, CA: CSLI Publications.

Bird, Steven & Ewan Klein. 1994. Phonological analysis in typed feature systems. *Computational Linguistics* 20(3). 455–491.

Boas, Hans C. 2003. *A Constructional approach to resultatives* Stanford Monographs in Linguistics. Stanford, CA: CSLI Publications.

Boas, Hans C. 2014. Lexical approaches to argument structure: Two sides of the same coin. *Theoretical Linguistics* 40(1–2). 89–112.

Bouma, Gosse, Robert Malouf & Ivan A. Sag. 2001. Satisfying constraints on extraction and adjunction. *Natural Language and Linguistic Theory* 19(1). 1–65.

Bresnan, Joan. 2001. *Lexical-Functional Syntax*. Oxford, UK/Cambridge, USA: Blackwell.

Brew, Chris. 1995. Stochastic HPSG. In Abney & Hinrichs (1995) 83–89.

Briscoe, Ted J. & Ann Copestake. 1999. Lexical rules in constraint-based grammar. *Computational Linguistics* 25(4). 487–526.

Cappelle, Bert. 2006. Particle placement and the case for "allostructions". *Constructions online* 1(7). 1–28.

Chomsky, Noam. 1965. *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Chomsky, Noam. 2008. On phases. In Robert Freidin, Carlos P. Otero & Maria Luisa Zubizarreta (eds.), *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*, 133–166. Cambridge, MA: MIT Press.

Copestake, Ann. 2002. *Implementing typed feature structure grammars* (CSLI Lecture Notes 110). Stanford, CA: CSLI Publications.

Copestake, Ann. 2007. Applying robust semantics. In *Proceedings of the 10th Conference of the Pacific Assocation for Computational Linguistics (PACLING)*, 1–12.

Crysmann, Berthold. 2002. *Constraint-based co-analysis: Portuguese cliticisation and morphology-syntax interaction in HPSG* (Saarbrücken Dissertations in Computational Linguistics and Language Technology 15). Saarbrücken: Deutsches Forschungszentrum für Künstliche Intelligenz und Universität des Saarlandes.

Crysmann, Berthold. 2013. On the locality of complement clause and relative clause extraposition. In Gert Webelhuth, Manfred Sailer & Heike Walker (eds.), *Rightward movement in a comparative perspective* (Linguistik Aktuell/Linguistics Today 200), 369–396. Amsterdam: John Benjamins Publishing Co.

De Kuthy, Kordula. 2002. *Discontinuous NPs in German* (Studies in Constraint-

Based Lexicalism 14). Stanford, CA: CSLI Publications.

Demberg, Vera & Frank Keller. 2008. A psycholinguistically motivated version of TAG. In *Proceedings of the 9th International Workshop on Tree Adjoining Grammars and Related Formalisms TAG+9*, 25–32. Tübingen.

Dowty, David R. 1989. On the semantic content of the notion 'thematic role'. In Gennaro Chierchia, Barbara H. Partee & Raymond Turner (eds.), *Properties, types and meaning*, vol. 2 Studies in Linguistics and Philosophy, 69–130. Dordrecht: Kluwer Academic Publishers.

Engdahl, Elisabet & Enric Vallduví. 1996. Information packaging in HPSG. In Claire Grover & Enric Vallduví (eds.), *Edinburgh Working Papers in Cognitive Science, vol. 12: Studies in HPSG*, chap. 1, 1–32. Edinburgh: Centre for Cognitive Science, University of Edinburgh. `ftp://ftp.cogsci.ed.ac.uk/pub/CCS-WPs/wp-12.ps.gz`.

Flickinger, Daniel P. 1987. *Lexical rules in the hierarchical lexicon*: Stanford University dissertation.

Gazdar, Gerald. 1981. Unbounded dependencies and coordinate structure. *Linguistic Inquiry* 12. 155–184.

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar approach to argument structure* Cognitive Theory of Language and Culture. Chicago/London: The University of Chicago Press.

Goldberg, Adele E. 2006. *Constructions at work: The nature of generalization in language* Oxford Linguistics. Oxford, New York: Oxford University Press.

Goldberg, Adele E. 2014. Fitting a slim dime between the verb template and argument structure construction approaches. *Theoretical Linguistics* 40(1–2). 113–135.

Günther, Carsten, Claudia Maienborn & Andrea Schopp. 1999. The processing of information structure. In Peter Bosch & Rob van der Sandt (eds.), *Focus: Linguistic, cognitive, and computational perspectives* Studies in Natural Language Processing, 18–42. Cambridge, UK: Cambridge University Press. Rev. papers orig. presented at a conference held 1994, Schloss Wolfsbrunnen, Germany.

Guzmán Naranjo, Matías. 2015. Unifying everything: Integrating quantitative effects into formal models of grammar. In *Proceedings of the 6th Conference on Quantitative Investigations in Theoretical Linguistics*, Tübingen. doi:10.15496/publikation-8636.

Haider, Hubert. 2000. OV is more basic than VO. In Peter Svenonius (ed.), *The derivation of VO and OV*, 45–67. Amsterdam: John Benjamins Publishing Co.

Haider, Hubert. 2016. On predicting resultative adjective constructions. Ms. Universität Salzburg.

Höhle, Tilman N. 1999. An architecture for phonology. In Robert D. Borsley & Adam Przepiórkowski (eds.), *Slavic in Head-Driven Phrase Structure Grammar*, 61–90. Stanford, CA: CSLI Publications. Republished as Höhle (2017).

Höhle, Tilman N. 2017. An architecture for phonology. In Stefan Müller, Marga Reis & Frank Richter (eds.), *Beiträge zur Grammatik des Deutschen* (Classics in Linguistics 5), Berlin: Language Science Press. Originally published as Höhle

(1999).

Jackendoff, Ray S. 2011. What is the human language faculty? Two views. *Language* 87(3). 586–624.

Kathol, Andreas. 2000. *Linear syntax*. New York, Oxford: Oxford University Press.

Kathol, Andreas & Carl J. Pollard. 1995. Extraposition via complex domain formation. In Hans Uszkoreit (ed.), *33rd Annual Meeting of the Association for Computational Linguistics. Proceedings of the conference*, 174–180. Cambridge, MA: Association for Computational Linguistics.

Kayne, Richard S. 1994. *The antisymmetry of syntax* (Linguistic Inquiry Monographs 25). Cambridge, MA: MIT Press.

Keller, Frank. 1995. Towards an account of extraposition in HPSG. In Abney & Hinrichs (1995) 301–306.

Kiefer, Bernd, Hans-Ulrich Krieger & Mark-Jan Nederhof. 2000. Efficient and robust parsing of word hypotheses graphs. In Wolfgang Wahlster (ed.), *Verbmobil: Foundations of speech-to-speech translation* Artificial Intelligence, 280–295. Berlin: Springer Verlag.

Koenig, Jean-Pierre & Karin Michelson. 2010. Argument structure of Oneida kinship terms. *International Journal of American Linguistics* 76(2). 169–205.

Konieczny, Lars. 1996. *Human sentence processing: A semantics-oriented parsing approach*: Universität Freiburg Dissertation. IIG-Berichte 3/96.

Kuhn, Jonas. 1995. Information packaging in German: Some motivation from HPSG-based translation. Universität Stuttgert, ms. `ftp://ftp.ims.uni-stuttgart.de/pub/papers/kuhn/Info-Pack.pdf`.

Kuhn, Jonas. 1996. An underspecified HPSG representation for information structure. In Jun-ichi Tsuji (ed.), *Proceedings of COLING-96. 16th International Conference on Computational Linguistics (COLING96). Copenhagen, Denmark, August 5–9, 1996*, 670–675. Copenhagen, Denmark: Association for Computational Linguistics.

Labelle, Marie. 2007. Biolinguistics, the Minimalist Program, and psycholinguistic reality. *Snippets* 14. 6–7. `http://www.ledonline.it/snippets/`.

Levine, Robert D. & Walt Detmar Meurers. 2006. Head-Driven Phrase Structure Grammar: Linguistic approach, formal foundations, and computational realization. In Keith Brown (ed.), *The encyclopedia of language and linguistics*, 237–252. Oxford: Elsevier Science Publisher B.V. (North-Holland) 2nd edn.

Marslen-Wilson, William. 1975. Sentence perception as an interactive parallel process. *Science* 189(4198). 226–228.

Meurers, Walt Detmar. 2001. On expressing lexical generalizations in HPSG. *Nordic Journal of Linguistics* 24(2). 161–217.

Meurers, Walt Detmar, Gerald Penn & Frank Richter. 2002. A web-based instructional platform for constraint-based grammar formalisms and parsing. In Dragomir Radev & Chris Brew (eds.), *Effective tools and methodologies for teaching NLP and CL*, 18–25. Association for Computational Linguistics. Proceedings of the Workshop held at 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia, PA.

Müller, Stefan. 1996. The Babel-System: An HPSG fragment for German, a parser, and a dialogue component. In *Proceedings of the Fourth International Conference on the Practical Application of Prolog*, 263–277. London.

Müller, Stefan. 1999a. *Deutsche Syntax deklarativ: Head-Driven Phrase Structure Grammar für das Deutsche* (Linguistische Arbeiten 394). Tübingen: Max Niemeyer Verlag.

Müller, Stefan. 1999b. Parsing of an HPSG grammar for German: Word order domains and discontinuousconstituents. In Jost Gippert & Peter Olivier (eds.), *Multilinguale Corpora: Codierung, Strukturierung, Analyse. 11. Jahrestagung der Gesellschaft für Linguistische Datenverarbeitung*, 292–303. Prag: enigma corporation.

Müller, Stefan. 2002. *Complex predicates: Verbal complexes, resultative constructions, and particle verbs in German* (Studies in Constraint-Based Lexicalism 13). Stanford, CA: CSLI Publications.

Müller, Stefan. 2005a. Resultative Constructions: Syntax, world knowledge, and collocational restrictions: Review of Hans C. Boas: A Constructional approach to resultatives. *Studies in Language* 29(3). 651–681.

Müller, Stefan. 2005b. Zur Analyse der scheinbar mehrfachen Vorfeldbesetzung. *Linguistische Berichte* 203. 297–330.

Müller, Stefan. 2006. Phrasal or lexical Constructions? *Language* 82(4). 850–883. doi:10.1353/lan.2006.0213.

Müller, Stefan. 2007. *Head-Driven Phrase Structure Grammar: Eine Einführung* (Stauffenburg Einführungen 17). Tübingen: Stauffenburg Verlag 1st edn.

Müller, Stefan. 2013a. The CoreGram project: A brief overview and motivation. In Denys Duchier & Yannick Parmentier (eds.), *Proceedings of the workshop on high-level methodologies for grammar engineering (HMGE 2013), Düsseldorf*, 93–104.

Müller, Stefan. 2013b. *Grammatiktheorie* (Stauffenburg Einführungen 20). Tübingen: Stauffenburg Verlag 2nd edn.

Müller, Stefan. 2015. The CoreGram project: Theoretical linguistics, theory development and verification. *Journal of Language Modelling* 3(1). 21–86. doi:10.15398/jlm.v3i1.91.

Müller, Stefan. 2016. *Grammatical theory: From Transformational Grammar to constraint-based approaches* (Textbooks in Language Sciences 1). Berlin: Language Science Press. doi:10.17169/langsci.b25.167.

Müller, Stefan. 2017. *German sentence structure: An analysis with special consideration of so-called multiple fronting* Empirically Oriented Theoretical Morphology and Syntax. Berlin: Language Science Press. Revise and resubmit.

Müller, Stefan & Walter Kasper. 2000. HPSG analysis of German. In Wolfgang Wahlster (ed.), *Verbmobil: Foundations of speech-to-speech translation* Artificial Intelligence, 238–253. Berlin: Springer Verlag.

Müller, Stefan & Stephen Mark Wechsler. 2014a. Lexical approaches to argument structure. *Theoretical Linguistics* 40(1–2). 1–76. doi:10.1515/tl-2014-0001.

Müller, Stefan & Stephen Mark Wechsler. 2014b. Two sides of the same slim Boo-

jum: Further arguments for a lexical approach to argument structure. *Theoretical Linguistics* 40(1–2). 187–224.

Orgun, Cemil Orhan. 1996. *Sign-based morphology and phonology*: University of California, Berkeley dissertation.

Paggio, Patrizia. 2005. Representing information structure in a formal grammar of Danish. In Takashi Washio, Akito Sakurai, Katsuto Nakajima, Hideaki Takeda, Satoshi Tojo & Makoto Yokoo (eds.), *New frontiers in artificial intelligence: Joint JSAI 2005 Workshop post-proceedings* (Lecture Notes in Computer Science 4012), 93–102. Berlin: Springer Verlag. doi:10.1007/11780496.

Penn, Gerald. 2004. Balancing clarity and efficiency in typed feature logic through delaying. In Donia Scott (ed.), *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), main volume*, 239–246. Barcelona, Spain.

Phillips, Colin. 2003. Linear order and constituency. *Linguistic Inquiry* 34(1). 37–90.

Pollard, Carl J. & Ivan A. Sag. 1987. *Information-based syntax and semantics* (CSLI Lecture Notes 13). Stanford, CA: CSLI Publications.

Pollard, Carl J. & Ivan A. Sag. 1994. *Head-Driven Phrase Structure Grammar* Studies in Contemporary Linguistics. Chicago: The University of Chicago Press.

Pullum, Geoffrey K. & Barbara C. Scholz. 2001. On the distinction between generative-enumerative and model-theoretic syntactic frameworks. In Philippe de Groote, Glyn Morrill & Christian Retor (eds.), *Logical Aspects of Computational Linguistics: 4th International Conference* (Lecture Notes in Computer Science 2099), 17–43. Berlin: Springer Verlag.

Reape, Mike. 1991. Word order variation in Germanic and parsing. DYANA Report Deliverable R1.1.C University of Edinburgh.

Reape, Mike. 1994. Domain union and word order variation in German. In John Nerbonne, Klaus Netter & Carl J. Pollard (eds.), *German in Head-Driven Phrase Structure Grammar* (CSLI Lecture Notes 46), 151–198. Stanford, CA: CSLI Publications.

Reape, Mike. 2000. Formalisation and abstraction in linguistic theory II: Toward a radical Linearisation Theory of German. unpublished paper.

Richter, Frank. 2004. *A mathematical formalism for linguistic theories with an application in Head-Driven Phrase Structure Grammar*: Eberhard-Karls-Universität Tübingen Phil. Dissertation (2000). https://publikationen.uni-tuebingen.de/xmlui/handle/10900/46230.

Ross, John Robert. 1967. *Constraints on variables in syntax*: MIT dissertation. http://files.eric.ed.gov/fulltext/ED016965.pdf. Reproduced by the Indiana University Linguistics Club and later published as Ross (1986).

Ross, John Robert. 1986. *Infinite syntax!* Norwood, New Jersey: Ablex Publishing Corporation.

Sag, Ivan A. 1997. English relative clause constructions. *Journal of Linguistics* 33(2). 431–484.

Sag, Ivan A. 2012. Sign-Based Construction Grammar: An informal synopsis. In Hans C. Boas & Ivan A. Sag (eds.), *Sign-based Construction Grammar* (CSLI Lecture Notes 193), 69–202. Stanford, CA: CSLI Publications.

Sag, Ivan A. & Thomas Wasow. 2011. Performance-compatible competence grammar. In Robert D. Borsley & Kersti Börjars (eds.), *Non-transformational syntax: Formal and explicit models of grammar: A guide to current models*, 359–377. Oxford, UK/Cambridge, MA: Blackwell Publishing Ltd.

Sag, Ivan A., Thomas Wasow & Emily M. Bender. 2003. *Syntactic theory: A formal introduction* (CSLI Lecture Notes 152). Stanford, CA: CSLI Publications 2nd edn.

Schabes, Yves, Anne Abeillé & Aravind K. Joshi. 1988. Parsing strategies with 'lexicalized' grammars: Application to Tree Adjoining Grammars. Technical Report MS-CIS-88-65. University of Pennsylvania Department of Computer and Information Science.

Shieber, Stuart M. & Mark Johnson. 1993. Variations on incremental interpretation. *Journal of Psycholinguistic Research* 22(2). 287–318.

Steels, Luc (ed.). 2011. *Design patterns in Fluid Construction Grammar* (Constructional Approaches to Language 11). Amsterdam: John Benjamins Publishing Co.

Steels, Luc. 2013. Fluid Construction Grammar. In Thomas Hoffmann & Graeme Trousdale (eds.), *The Oxford handbook of Construction Grammar* Oxford Handbooks, 153–167. Oxford: Oxford University Press.

Steels, Luc & Joachim De Beule. 2006. A (very) brief introduction to Fluid Construction Grammar. Paper presented at the Third International Workshop on Scalable Natural Language Understanding (ScaNaLU 2006) June 8, 2006, following HLT/NAACL, New York City.

Steels, Luc & Remi van Trijp. 2011. How to make Construction Grammars fluid and robust. In Steels (2011) 301–330.

Tanenhaus, Michael K., Michael J. Spivey-Knowlton, Kathleen M. Eberhard & Julie C. Sedivy. 1995. Integration of visual and linguistic information in spoken language comprehension. *Science* 268(5217). 1632–1634.

Tanenhaus, Michael K., Michael J. Spivey-Knowlton, Kathleen M. Eberhard & Julie C. Sedivy. 1996. Using eye movements to study spoken language comprehension: Evidence for visually mediated incremental interpretation. In Toshio Inui & James L. McClelland (eds.), *Information integration in perception and communication* (Attention and Performance XVI), 457–478. Cambridge, MA: MIT Press.

van Trijp, Remi. 2011. A design pattern for argument structure constructions. In Steels (2011) 115–145.

van Trijp, Remi. 2013. A comparison between Fluid Construction Grammar and Sign-Based Construction Grammar. *Constructions and Frames* 5(1). 88–116.

van Trijp, Remi. 2014. Long-distance dependencies without filler–gaps: A cognitive-functional alternative in Fluid Construction Grammar. *Language and Cognition* 6(2). 242–270.

Walther, Markus. 1999. *Deklarative prosodische Morphologie: Constraint-basierte*

Analysen und Computermodelle zum Finnischen und Tigrinya (Linguistische Arbeiten 399). Tübingen: Max Niemeyer Verlag.

Wellens, Pieter, Remi van Trijp, Katrien Beuls & Luc Steels. 2013. Fluid Construction Grammar for historical and evolutionary linguistics. In Miriam Butt & Sarmad Hussain (eds.), *51st Annual Meeting of the Association for Computational Linguistics proceedings of the conference (system demonstrations)*, 127–132. Association for Computational Linguistics.

Wetta, Andrew C. 2011. A Construction-based cross-linguistic analysis of V2 word order. In Stefan Müller (ed.), *Proceedings of the 18th International Conference on Head-Driven Phrase Structure Grammar, University of Washington*, 248–268. Stanford, CA: CSLI Publications. http://csli-publications.stanford.edu/HPSG/2011/.

Wilcock, Graham. 2005. Information structure and Minimal Recursion Semantics. In Antti Arppe, Lauri Carlson, Krister Lindén, Jussi Piitulainen, Mickael Suominen, Martti Vainio, Hanna Westerlund & Anssi Yli-Jyrä (eds.), *Inquiries into words, constraints and contexts: Festschrift for Kimmo Koskenniemi on his 60th birthday* CSLI Studies in Computational Linguistics ONLINE, 268–277. Stanford, CA: CSLI Publications.

Winkler, Susanne. 1997. *Focus and secondary predication* (Studies in Generative Grammar 43). Berlin, New York: Mouton de Gruyter.