Basic Dependency-Based Logical Grammar

Technical Report ÚFAL-TR-5 (June 1998)

Geert-Jan M. Kruijff
⟨*gj@ufal.mff.cuni.cz*⟩

Copies of ÚFAL Technical Reports can be ordered from:

Institute for Formal and Applied Linguistics (ÚFAL)
Faculty of Mathematics and Physics, Charles University
Malostranské náměstí 25, CZ-11800 Prague 1
Czech Republic

or can be obtained via the Web: `http://ufal.mff.cuni.cz`.

Basic Dependency-Based Logical Grammar
(UFAL-TR-5, June 1998)

Document typeset in LaTeX 2ε using the AMS-LaTeX 2ε distribution, and the proof, theorem, covington and epic macro packages.

The research reported in this manuscript was supported by the Grant Agency of the Czech Republic (GAČR) 405/96/K214, and the Ministery of Education Project No. VS96151.

Abstract

A logical grammar is presented that employs (kinds of) *dependency relations* as its basic categories, rather than constituents. The aim with this dependency-based logical grammar is to provide a calculus for doing analysis based on the description of natural language as provided by Sgall et al ([46, 45]) and Petkevič ([42]).

Table of Contents

Motivation **4**
1 Introduction 4
2 Overview 5

1 Functional Generative Description **7**
1 Introduction 7
2 Tectogrammatical Representations 9
2.1 Dependency Relations 10
2.2 Lexical Information 10
2.3 Coordination and Apposition 12
2.4 Contextual Boundness/Nonboundness, and Deep Word Order 12
2.5 Grammatical Coreference 14

2 Multi-Modal Logical Grammar **15**
1 Introduction 15
1.1 General Architecture 16
2 Moortgat & Oehrle's Approach 18
3 Hepple's Approach 19

3 Basic Dependency-Based Logical Grammar **21**
1 Introduction 21
2 Example 23
3 The Surface Dimension 27
3.1 Surface Modes 27
3.2 Features 30
3.3 Agreement 31
3.4 Lexical Underspecification 32
4 The Deep Dimension 33
4.1 Valency Frames as θ-frames 33
4.2 Deep Modes 34
4.3 Extending the Notion of Valency Frame 35
4.4 Agreement Between Heads and Dependents 38
4.5 Features and Agreement 38
4.6 Agreement and n-ary Composition 40

4.7 Proposed Solution 41
5 Form & Function, and Grammaticality 43

4 Term-Assignment in DBLG 45
1 The Idea Of Term-Assignment 45
2 Types in DBLG 46
3 Formalization 47
4 Terms As Dependency Structures 50

5 Extending DBLG 53
1 Introduction 53
2 Categories Combining Surface and Deep Modes 53
3 Extending the Behavior of Deep Modes 57
4 Commutativity and Informativity 57
4.1 Linguistic Intuitions 58
4.2 Structure and Informativity 58
4.3 Basic Hypotheses 59
4.4 Formalization 61
4.5 Topic/Focus-Articulation 64

6 A Few Arguments for DBLG 68
1 Remarks on the Dependency/Constituency Debate 68
2 Some Arguments for DBLG 70
3 Final Remarks 70

A Dependencies and Features 72
1 Dependencies 72
2 Features 72

References 74

Motivation

... ὁδὸς ἄνω κάτω μία καὶ ὡυτή.
... [the] way up [and] down [is] one and [the] same.
Heraclitus, *Diels-Kranz 22 B 60*

1 Introduction

The Prague School of Linguistics has a long standing as a tradition in the description of natural language. With the inception of the Prague Linguistic Circle in the 1920s started an approach to perceiving language in a structural/functional way - a perspective which found a mathematical formulation in Sgall et al's Functional Generative Description (FGD; [46, 45]), recently partly reworked by Petkevič where it concerned the relation between deep structure and surface form ([42]).

FGD is a generative approach in the sense that it shows how different layers of language (for example deep structure, morphology, phonetics) interact with another *for a speaker to formulate an utterance.* The notion of 'generative approach' can be understood in the Chomskian sense of encompassing a generative base, on top of which transformations (or 'transducers') are defined.

A consequence of employing transducers is that it is less straightforward to describe an analytic perspective employing FGD. Although higher level descriptions of the interpretation of utterances have been provided in terms of topic/focus-articulation (cf. Hajičová's [8]), there is less formal explication available for the analysis of utterances in terms of constructing their deep structures. Kosík and Sgall developed in [24] an intensional interpretation of the deep structures, and the TIBAQ system [9] embodies a parser based on FGD, outputting deep structures. Yet, neither of them presents a formal calculus for analysing surface forms in terms of deep structures.

Therefore, what we would like to pursue is the development of a logical grammar which is based on FGD not only in the sense that it outputs the kind of structures in terms of which FGD describes natural language, but which also attempts to mirror FGD in the *way* a sentence is analysed. In this manuscript we present the basics for such a *dependency-based logical grammar.*

2 Overview

We commence the manuscript with describing (our understanding of) FGD. Of particular interest is the deep structure, or 'tectogrammatical representation', which elucidates how the strings of a sentence can be interpreted as playing specific roles or 'functions'. Possible 'functions' are that of head, or dependent modifying a head by a particular dependency relation - whereby 'functioning' as such is being conditioned by, and influences, the actual structure.

It is due to this conception of structural sensitivity to the actual construction (process) that I opt for developing a logical grammar along the lines of *multimodal logical grammar*. In multimodal logical grammar, we conceive of a sentence as being built up by words that are combined by -possibly- different modes of composition. That makes multimodal logical grammar different from traditional type-logical (categorial) grammars, where there is only one mode of composition (namely, $\circ$ with its left- and right-residuals $\backslash, /$). The advantageous aspect of having multiple modes of composition is that we can take each mode to stand for modeling a specific, structural phenomenon (like movement, binding, extraction, etcetera). A grammatical sentence is therefore a sentence in which such structural phenomena are combined in a proper way. To ensure such proper-ness, our analysis is made *resource-sensitive*, that is, sensitive to the structure already formed.

Various approaches to multimodal logical grammar are around, each coming with their own philosophy and the unavoidable advantages and disadvantages. Chapter 2 gives a general introduction to multimodal logical grammar, discussing the ideas behind multimodal logical grammar and reviewing how these ideas are realized in the approaches open to us.

From then on, I will become more concerned with developing *dependency-based logical grammar*, or DBLG for short. Chapter 3 discusses the basics of DBLG. These basics include the logic frameworks for modeling composition at the surface and in the deep dimension, and a formalization of the linguistic form/function relation. Subsequently, in chapter 4, I discuss the idea of term-assignment in proof-theoretical approaches to syntax, and argue how terms (of a λ-calculus) can be perceived of as dependency structures which are built up during analysis.

Chapter 5 discusses two extensions to the basics of DBLG. The first extension concerns so-called "mixed categories" which allow for surface categories to include constraints on their functional interpretation, and for deep categories to include constraints on functional realization. The second extension concerns structural indications of informativity - the syntactic preliminaries of a sentence's topic and focus articulation.

The report finishes with a brief discussion of DBLG as a *dependency-based* framework for natural language syntax.

The contribution I hope to make with the theory developed in this report is as follows. To start with, I attempt to provide a logical grammar which, in its rudimentary form as presented here, shows how analysis can be done covering a small part of FGD. Relative to the more general setting of type-logical grammars, DBLG exemplifies that not all type-logical grammars need to be in the spirit of Categorial Grammar[1].

Acknowledgements

Various people have, at one point or another, been substantial in the development of this manuscript's ideas. In particular I would like to thank (in alphabetical order) Tom Cornell, Eva Hajičová, Mark Hepple, Dirk Heylen, Uwe Mönnich, Michael Moortgat, Karel Oliva, Valeria de Paiva, Jarmila Panevová, Niki Petkevič, Saša Rosen, Petr Sgall and Mark Steedman for there time and efforts. Of course, they should be excused for any remaining errors, since those are my own. Furthermore, I would like to thank the International Centre of the University of Tübingen, and the Department of Computer Science of the University of Birmingham, for financially supporting short stays at the respective universities during which earlier versions of this manuscript were presented, and the University of the Saarland for financially supporting a stay in Saarbrücken, during which various improvements to the theory where made thanks to discussions and joint research with colleagues there. Finally, thanks are due to Herman Hendriks for the quotation gracing the beginning of this chapter (and, more importantly, the accompanying LaTeX code).

This manuscript was typeset using LaTeX 2_ε, the AMS-LaTeX distribution, and the theorem, proof, covington and epic macro packages.

The research reported in this manuscript was supported by the Grant Agency of the Czech Republic (GAČR) 405/96/K214, and the Ministery of Education Project No. VS96151.

[1] Other examples being Cornell's characterization of Minimalism in terms of type-logical grammar, [3], and Johnson's resource sensitive version of Lexical-Functional Grammar, R-LFG [21, 22]

Chapter 1

Functional Generative Description

1 Introduction

The framework of Functional Generative Description (*FGD*) has been developed within the Prague School of Linguistics as a functional approach to the description of language. It describes, in a systematic way, the main properties and principles of language by showing how different layers of language interact with one another in order for a speaker to formulate an utterance.

FGD is thus a *stratificational approach*, which is functional in the sense that a linguistic function (e.g. dependency) at one level is realized by a form in the next lower level[1]. The stratification employed is usually referred to as the *system*, or *strata*, *of language*, and encompasses the following levels:

1. Deep structure, or *tectogrammatical representation.*
2. Morphonemics.
3. Phonemics.
4. Phonetics.

A speaker's intention to convey particular information (or, in other words, to let the utterance have a particular content) is consequently perceived of in the following way. Given (extra-lingual) content, it is postulated that the speaker has a deep structure or tectogrammatical representation (see section 2 below) that relates that content in a linguistic fashion. Based on this representation, a surface structure can be generated, whose elements in turn can be subsequently processed (*transformed*) through the phonological and -finally- phonetic layers to arrive at an 'audible/graphemic output'.

Here, we are primarily concerned with relation between the 'surface form' and the deep structure. Thereby, the surface form is positioned at the stratum of morphonemics, and is conceived of as a sequence of strings. For future discussions it is important that we make explicit here that we do not conceive of a global notion of surface syntax. That is, the relation between the surface form and the deep structure is really one of realization/interpretation

[1] The notions of 'lower' and 'higher' are to be understood as follows: Deep structure is the highest level, phonetics the lowest.

- we do not first transform the deep structure into a 'surface structure' with its own syntax, and subsequently transform that 'surface structure' into a surface form [2].

To give an example of the approach, consider that we want to convey (in Czech) that the little girl is beautiful. So, to begin with, we have something like the following as our deep structure:

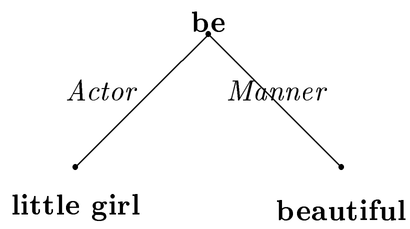


Figure. Deep structure little girl-be-beautiful

Subsequently, to realize this as a surface form, we take the strings corresponding to the items in the tectogrammatical representation, and transform them into proper forms showing their respective functions at the deep level. Thus, the string corresponding to "little girl" gets declined as a nominative, expressing its ACTOR-function.

What is interesting to note is that there are several possibilities to realize "little girl" in Czech. We have "dívka" and "děvče" (and accordingly, "krásná" and "krásné" as the respective realizations of "beautiful"). With regard to the tectogrammatical representation, we need not be concerned with those possibilities though - such is a matter of the surface form:

(1.1) Dívka je krásná.

(1.2) Děvče je krásné.

are both realizations of one and the same tectogrammatical representation expressing that the little girl is beautiful. In other words, a tectogrammatical representation is highly economical.

The formal core of the FGD was laid down in as early as the sixties, culminating in Sgall et al's [46]. There, the writers were primarily occupied with providing a "mathematically -thus linguistically- interesting" description of (linguistic) meaning. Extensive empirical support for the developed

[2] Later on, we will see that there are *in DBLG* mechanisms that may be interpreted as embodying a localized notion of surface syntax - particularly where the grammaticality of a sentence depends purely on realization, vz. the use of function words like expletive pronouns. However, rather than saying that a verb like 'rain' takes no Actor but needs something like a "grammatical subject", we specify the verb as needing an expletive pronoun 'it' with its realization.

formalism was provided by Sgall et al in [45] and various articles. Recently, the formalizations provided in [46] and [45] were reformulated by Petkevič in [40], [41], and [42] in terms of contemporary grammatical constructions.

Below a more in-depth description is given of FGD, whereby we focus on the tectogrammatical representation (*TR*) and its relation to a surface form[3]. The two basic reasons for paying special attention to the tectogrammatical representation is that it plays a pivotal role in whatever perspective we take:

- From a generative point of view, a TR is the starting point for generating an audible output.
- From an analytic viewpoint, a TR expresses the exact linguistic meaning of the sentence under analysis (provided the sentence is grammatical).

2 Tectogrammatical Representations

The term 'tectogrammatical representation' was introduced by H.B. Curry in [4] as the representation signifying how expressions represent processes of construction (cf. [5]). Applied to linguistics, we can understand a tectogrammatical representation thus as expressing how the interpretation of the sentence should be construed, or could be reconstrued. The tectogrammatical representation of a sentence *delineates*, or provides "guidelines", how the utterance's sense should be established. Succinctly put, the sentence's TR gives its "meaning potential" as far as structured by the language rather than by the extralinguistic content the sentence may have.

Within a TR, the following dimensions of linguistic meaning are expressed for the entities making up the representation[4]:

(i) Dependency relations;

(ii) Coordination and apposition (if applicable);

(iii) Contextual boundness (CB) or nonboundness (NB);

(iv) Deep word order;

(v) Grammatical coreference.

In the next subsections we will provide more detail.

[3] Other levels, though clearly belonging to FGD, are not treated of in our framework.
[4] That is, these features have been formally treated of by Petkevič in [41] and [42].

2.1 Dependency Relations

The basic entities that make up the TR are called *semantemes*. A semanteme can either be an entry from a lexicon $\mathcal{L}ex$, or a more complex coordination/apposition-construction (see below).

The main structure of the TR arises from how the various semantemes are related to one another via *dependency relations*. In such a relation, one semanteme (the *dependent*) is said to *modify* another semanteme (the *head*) via a specific kind of dependency relation. Because heads can be modified by multiple dependents, whereas a dependent can only be related to one head, a TR can also be depicted as an n-branching tree structure called a *dependency tree*. Distinctive about dependency trees is that it does not contain any nonterminal nodes, like a phrase-structure tree. See also the example tree on page 8. The main reason for the difference with phrase-structure trees is rather simple, though, if we perceive matters from an analytic perspective: A dependency tree is an (economic) representation of the *product* of analysis, whereas a phrase-structure tree is a representation of that very *process* of analysis.

Given the set $\mathcal{D}$ of kinds of dependency relations we discern for a given language, the following distinctions can be made:

- *Inner participant* (IP) versus *free modifier* (FM): an inner participant is a dependency relation via which a head can only be expanded at most once, whereas a head can be expanded any finite number of times via dependency relations classified as free modifiers.

- *Obligatory* versus *optional*: An obligatory dependency relation for a specific word indicates that it *must* be expanded via this dependency relation, whereas an optional dependency relation only indicates the *possibility* for expansion.

The IP/FM-distinction partitions our set $\mathcal{D}$, whereas the distinction obligatory/optional is relative to a specific word. The two distinctions are related in that, for a word w, the set of dependency relations along which w must be expanded is a subset of the union of the inner participants and free modifiers applicable for w. Which brings us to how words are specified in the lexicon - because it is there that words and their individual information are given.

2.2 Lexical Information

An entry in the lexicon specifies for a given word its *graphemic form*, its *wordclass*, and its *valency frame*. The graphemic form given is the uninflected form of the word. Regarding the word's wordclass, we have to make a distinction between words that *do* get represented in a TR, and those that *don't*. Those words that do get represented have one of the following wordclass: Verb (v), Noun (n), Adjective (adj), Adverb (adv) or Pronoun (pro)[5]. These words are also called *auto-semantic*.

Yet, there are also words are found in the sentence, but that do *not* get represented in the sentence's TR (as individual semantemes/nodes in a tree). Among these are *function words* and *auxiliary verbs*. Function words, like prepositions, can be perceived of as means to realize dependency relations: For example, if we have a noun **Prague** expanding a noun **house** via a LOCATION dependency relation, then we could have a TR looking as follows:

(1.3) **house** LOCATION ⟨**Prague**⟩

and realizations

(1.4) (Czech) "byt v Praze"

(1.5) (English) "(a/the) house in Prague"

Therefore, resulting from the distinction between deep structure and realization as a surface form, function words are not represented[6] Similarly, auxiliary verbs are not represented, since they can -essentially- be perceived of as means to realize a verb's Tense, Modality, et cetera[7]

Finally, the lexical entry for a word provides its *valency frame*. A word's valency frame specifies via which dependency relations a word can be expanded. It is possible that the valency frame is empty; if it is not, then for each dependency relation it is specified whether it is obligatory or optional. Usually, we also specify whether a dependency relation is an inner participant or a free modifier. As an example, consider the following (simplified) lexical entry for 'bought'

$$\text{'bought': } \textbf{buy} \text{ - v - } \mathrm{ACTOR}^{IP}_{OBL}, \mathrm{PATIENT}^{IP}_{OBL}, \mathrm{LOCATION}^{FM}_{OPT}.$$

The entry specifies that the *verb must* have an *Actor* and a *Patient*, and *may* have one or more modifiers expanding it as *Location*. Thus, what this lexical entry 'specifies' are realizations like the following, with the *'d ones being incorrect:

(1.6) I bought a ticket.

(1.7) *I bought.

(1.8) I bought a ticket at the travel agent's.

(1.9) I bought a ticket at the travel agent's in Prague.

[5] Whereby a further division of Pronoun can be made (cf. [42], p.17, where Petkevič refers to work by Machová).

[6] Below we will see more examples of function words, like connectives in coordination.

[7] An issue to be resolved is how modifications of auxiliary verbs should be represented.

(1.10) *I bought in Prague.

The reader may observe here a similarity between FGD's valency frames, and the θ-frames of Government & Binding-theory (cf. [7]). *theta*-frames specify the argument-places or *θ-roles* of the verb, conceived of as a predicate. Yet, the difference between *theta*-frames and valency frames is that the former only include the argument places that *must* be filled, whereas the latter also includes argument places which *may*, but need not, be filled. Such may be seen as a consequence of the two-dimensional classification of dependency relations[8].

2.3 Coordination and Apposition

As we already mentioned above, a semanteme need not correspond to a lexical entry; it may also be a more complex *coordination/apposition-unit* (c/a-unit). A c/a-unit is a semanteme on its own: it symbolizes the *whole*, arising from the members coordinated in a specific way. There are various reasons to make the whole as such identifiable. A c/a-unit can itself function as head, or as modifier, and it is particularly in the latter case that we can observe the need for a linguistic meaning of the whole, distinct from its members. Namely, take for example the sentence

(1.11) The cat and the dog are playing together.

We have a c/a-unit consisting of **cat** and **dog**, whereby the number of the whole is *plural* even though the members are each singular. Since the 'are' requires a plural actor, it is by reference to the coordinated members *as a whole* that we can judge this sentence grammatical[9].

2.4 Contextual Boundness/Nonboundness, and Deep Word Order

Contextual boundness and nonboundness are primary linguistic notions used to classify semantemes in a tectogrammatical representation as reflecting a speaker's disposition towards the actual state of affairs talked about, and his efforts to accommodate the hearer's needs as to be able to interpret what the speaker intents to convey (cf. [45], p.177). Thereby, contextual boundness can loosely be compared to indicate what is salient, 'given', recoverable from the already established discourse context; whereas contextual

[8]A note should be made here on "the" Praguian notion of 'valency frame'. The remark made here holds for 'valency frame' as Sgall puts it forward. Panevová employs a notion of 'valency frame' in which only the obligatory slots are represented, which thus mirrors GB's notion of 'θ-frame'.

[9]Yet, as Petkevič notes in [42] (p.18), representing a c/a-unit as a special complex semanteme consisting of its main members does not come without a price. Various formal and internally linguistic difficulties have to be overcome to make it work - cf. (*ibid*).

nonboundness is similar to 'novelty', not indicating a reference to something established but signalling the *introduction* something new into the context, or the *modification* of something recoverable.

Whether an element of the tectogrammatical representation is CB or NB depends on the placement of the modifier relative to the head, and other modifiers of that head. More specifically, we can define for each language a standard ordering in which modifiers (dependency relations) are to be arranged - the *systemic ordering.* The systemic ordering is a total ordering over all possible dependency relations, and is also reflected in a valency frame: for all dependencies D_i, D_j in that valency frame, it holds that $D_i \prec_{so} D_j$ implies that D_i precedes D_j in the valency frame. Now, roughly put, whenever a modifier occurs in a position different from the position in the systemic ordering, it is judged CB; only if the position complies with the systemic ordering, it can be judged NB[10].

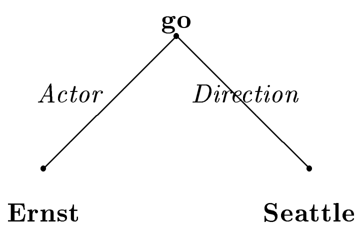
In the tectogrammatical representation we note what elements are CB or NB by simply labelling them as such. If we view the tectogrammatical representation as a dependency tree, we put elements that are CB relative to a head and its modifiers to the *left* of that head, and the NB elements to the *right*, all the while maintaining projectivity. For example, consider the following two sentences (capitals indicating stress):

(1.12) *Ernst šel do Seattle.*
Ernst went to SEATTLE.
En. Ernst went to Seattle.

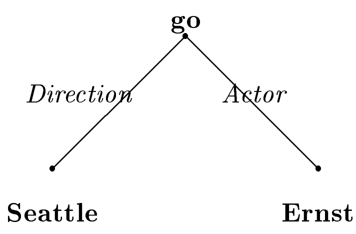
(1.13) *Do Seattle šel* ERNST.
To Seattle went Ernst.
En. ERNST went to Seattle.

The corresponding tectogrammatical representations and dependency trees would be:

(Ernst:Actor)^{CB} go (Seattle:Direction)^{NB}



(Seattle:Direction)^{CB} go (Ernst:Actor)^{NB}



[10]Of course, reality is more complex. In chapter 5 we discuss the relation between systemic ordering and CB/NB-ness in more detail. See also [45] and [8].

Figure. Displaying CB/NB-ness in dependency trees/TRs

Finally, we should make some notes concerning the CB/NB-ness of verbal heads. The boundary between the CB elements and the NB elements is always posited around the head of the clause: Either the head is deemed CB, resulting in a boundary right after the head, or the head is NB, meaning that the boundary is right before the head. In the above examples, we did not talk about the head's CB/NB-ness - such involves a more elaborate discussion which we reserve for chapter 5.

As for the relation between the deep word order and the surface word order is that the instantiation of a tectogrammatical representation in which all modifiers are ordered according to the systemic ordering, results in a surface form which corresponds to the standard surface word order. In chapter 4 we will discuss this in more detail.

2.5 Grammatical Coreference

Grammatical coreference concerns phenomena like the reference of a relative pronoun to an antecedent noun (or, possibly, pronoun), control, and reflexive pronouns. FGD formalizes the identity between the relative pronoun and its antecedent by means of a *relative path* between the two. From the generative perspective, such a path should serve for the transducers that create the proper surface forms to transfer the antecedent's grammatical categories of number and gender to the relative pronoun during the transduction ([42], p.22).

The following examples provide arguments for viewing such a path necessary (for generation):

(1.14) Ich sah einen Politiker, welcher klug war.

(1.15) *Ich sah ein Mädchen, welcher hübsch war.

In example (1.14) the relative pronoun "welcher" refers to "Politiker" and has the proper number (singular) and gender (masculine). Example (1.15) is incorrect in that "welcher" has an improper gender - "Mädchen" is neuter, not masculine.

Petkevič discusses in [42] and [41] more complex forms of grammatical coreference as well. Because we are not concerned with grammatical coreference in this manuscript, we will reserve discussion for a future occasion and leave the issue by noting its basic construction.

Chapter 2

Multi-Modal Logical Grammar

1 Introduction

Multi-modal logical grammar (MMLG) is a grammar in which we model grammaticality using not just *one* mode of composition, but *several* modes. Each mode can be used to model a specific, syntactic phenomenon. A mode is defined using a small logic, that formalizes the mode's behavior. Consequently, a MMLG can be conceived of as a *hybrid system* of small logics.

When performing syntactic analysis with such a grammar, several issues are of importance: How do we know how different modes (i.e. different syntactic phenomena) can be combined, and how do we know *when* a mode can be employed?

The first issue concerns the architecture of the grammar as a hybrid system of logics. Rather than having a collection of individual, isolated logics, we should make sure that logics can *communicate*. We can make logics communicate by including rules that specify how modes can co-occur or *interact*, and rules that enable us to "move" between logics. By moving between logics we will understand the possibility to employ a mode X instead of a mode Y, whereby X allows a slightly different logical behavior than Y. An example of this is the move from a mode of composition that only allows for canonical-order composition to a mode allowing for non-canonical-order composition.

Of course, we should make sure that not every mode can be employed in just about every situation. A more linguistic perspective on this issue may be illuminating. Recall that modes are used to model specific structural phenomena. Now, how to interpret a phenomenon observable in a sentence as being a specific syntactic construction is usually taken to depend on the *context* in which the observed phenomenon occurs. A similar viewpoint is taken in in MMLG.

Namely, while analysing a sentence, we slowly build up a structure that is a (partial) recognition of the sentence. The structure shows the sentence's words that are covered, and the modes by which these words have been combined (i.e. the actual structure). It is with respect to this already formed structure that any further step in an analysis is to be made: Whether a phenomenon can be analysed as a specific syntactic construction, thereby thus employing a particular mode, will depend on whether the mode can be used in the context of the already employed modes. Thus, as long as the mode to be used can properly interact with the available structure, we can continue our analysis. Otherwise, we can try to see whether the grammar allows us to move to another mode of composition, and if so, whether that moved-to mode would provide an outcome[1].

Recapitulating, an important observation is that the use of modes is sensitive to, *controlled* by, the already established context. The already available structure, establishing a context in which further analysis is to proceed, has the more general (logical) name "resource", and logics like MMLG are said to be "resource-sensitive" due to the role of resources in analysis.

The reader may still wonder where we get these "modes" from in the first place. An answer to that is that, as is common in categorial grammars, we have a lexicon that contains entries giving what categories words have. In essence, a word's category either simply gives a simple wordclass, or it elucidates how the word may be combined with other words in a well-formed (grammatical) fashion. In standard (applicative) categorial grammar [2] we can encounter categories like $np \setminus (s/np)$, saying that the verb (s) needs to combine to its right with a noun phrase ("s/np"), and to its left with a noun phrase. If we take the right noun phrase as the object, and the left noun phrase as the subject, $np\setminus(s/np)$ thus gives the category of a transitive verb. However, whereas applicative categorial grammar only knows one mode of composition ($\circ$ and its left- and right-residuals, $\setminus$ and $/$), MMLG has multiple modes. Therefore, a lexical entry's category will not only specify with what a word will combine, but also *how* - thus, using what mode of composition. The name of the mode will be subscription to the slashes, so $np \setminus_i (s/_j np)$ says that combination to the right is to use a mode j whereas combination to the left uses a (possibly different) mode i.

The next section describes in more detail the general architecture of a MMLG.

1.1 General Architecture

Essentially a MMLG consists of two parts: A model theory, and a proof theory. The model theory describes, in mathematical sense, the structures we want to consider as valid - well-formed. Modes of composition are interpreted on the model, an interpretation defining the intended semantics of a mode by specifying how it can take one or more valid structures and turn it into another valid structure. Or more precisely: a model is a set of possible worlds, each world being a grammatical structure, and modes are defined by accessibility relations between worlds (and conditions on the accessibility relations).

[1] If we find, in the end, no possibility to continue our analysis towards complete coverage of the sentence, we conclude that the sentence is ungrammatical.

The proof theory defines the actual mechanisms how to employ the modes in analysis, whereby we ensure that that employment exactly follows the intended semantics of the modes. The relation between an inference carried out within the proof theory and the *grammaticality* of the structure that is formed by that inference is then follows. By exactly following the intended semantics of the modes, every step in the proof will result in a structure that is interpretable in the model theory - thus, a grammatical structure. Being able to carry through a proof until the entire sentence has been analysed thus means, in other words, that the sentence is grammatical. The kind of proof system generally used is that of a *labelled deductive system* [6]. A labelled deductive system is a system for deduction where we put labels to the formulas, and let our inferences manipulate pairs $\langle label, formulas \rangle$ rather than just formulas. In the case of MMLG, we use the form $R \vdash C : S$ with R the resources (label), C the category (formula), and S the interpretation of the inference process.

In the introduction we already mentioned that a MMLG not only defines individual logics for the modes of composition, but also -by necessity- needs to specify how these modes work together. This we do in the proof theory. From a more logical perspective, the proof theory can be broken up into different kinds of rules

1. The logical rules - defining the basic proof procedure.
2. The structural rules - defining additional operations for use in a proof, like associativity, permutativity, weakening, contraction, etcetera (see below).
3. Rules defining how structures composed using the same mode can be combined.
4. Rules defining how one may move from one mode to another mode.
5. Rules defining how structures composed using different modes can be put together (i.e. how modes can interact).

Abstractly put, a hybrid system of logics (a MMLG) arises if we define the logical behavior of a mode in terms of what structural rules we may use when employing that mode, and how it co-exists with other modes available in the grammar.

People differ in their opinion, though, about the exact nature of these definitions, and what the intuitions are behind their formalizations. Frameworks for multimodal logical grammars have been separately proposed by Moortgat & Oehrle, Morrill, and Hepple. Below we discuss in more detail what Moortgat & Oehrle's and Hepple's frameworks amount to. For Morrill's theory, we refer to [35].

2 Moortgat & Oehrle's Approach

Moortgat & Oehrle ([32], [34]) propose one approach to resource-sensitive logical grammars. On their approach, one starts with a rather strong base logic (defining the basic proof procedure) and a set of *structural rules* defining logical behavior unavailable in the base logic itself.

For example, as a base logic one can take the system known as NL, the non-associative Lambek calculus. In that logic, associativity is unavailable, thus:

$$(A \circ B) \circ C \text{does not derive} A \circ (B \circ C) \tag{2.1}$$

However, we can define associativity as a *structural rule*. The idea is now *not* to simply add associativity to the base logic, since this would lead to another logic (namely, Lambek's L). Instead, *structural modalities* are defined that regulate the applicativity of structural rules, providing the possibility to make the 'extra' behavior (associativity) available in a controlled fashion. Once a term is decorated with a particular structural modality, the structural rule(s) accompanying the modality become available.

For example, we could formulate associativity to hold for those terms decorated with a $\diamond_a$:

$$(A \circ B) \circ C^{\diamond_a} \text{is the same as} A \circ (B \circ C^{\diamond_a}) \tag{2.2}$$

Now we would have a controlled access to associativity. We can do a similar thing for permutation, i.e.

$$D \circ C^{\diamond_p} \text{is the same as} C^{\diamond_p} \circ D \tag{2.3}$$

making permutation available whenever a term is decorated with $\diamond_p$.

The picture that evolves this way is that modes are defined in terms of a (strong) base logic like NL, and we can provide limited additions to the rather restrictive behavior of NL by means of structural modalities, allowing access to various structural rules. Logically speaking, the structural modality $\diamond_a$ introduces behavior that is available in L, and $\diamond_p$ behavior from the Van Benthem-Lambek calculus NLP (cf. [51]). Furthermore, if we would define interaction between $\diamond_a$ and $\diamond_p$ then the associative, permutative Lambek calculus LP also comes in reach[2].

Recapitulated, starting with a base logic, and adding structural rules defining additional logical behavior made accessible in a controlled way,

[2] A specification of the interaction between $\diamond_a$ and $\diamond_p$ could be given using Moortgat's [32] rules of mixed commutativity (MC) and mixed associativity (MA).

this leads to an entire landscape of logics through which one can travel along the roads opened up by decorations. If we add (sensitivity to) the head/dependent asymmetry ('dependency') to associativity and permutation, we obtain the following picture (cf. [32])

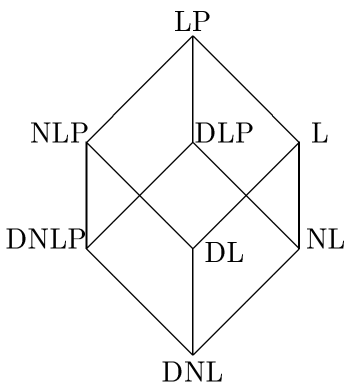


Figure. Resource Logical Landscape (Moortgat & Oehrle)

A particular detail of Moortgat & Oehrle's approach is that the transition from one mode of description to another mode, like $A \circ_i B \Rightarrow A \circ_j B$, is conceived of as a *loss* of structural information. There are good arguments for seeing matters that way. For example, it enables one to "forget" information, so that only that information is taken into account which is relevant to treating the phenomenon specified by the mode transfered to. A downside of this approach is that, however, a large number of modes are needed, each with their own axioms, to deal with different 'information needs'. A description of a reasonably complex linguistic phenomenon may thus easily become daunting - if not to construct, then to decipher.

3 Hepple's Approach

Hepple, on the other hand, proposes an approach in which he focuses on a *hybrid system of logics.* Instead of using a particular base logic, one commences with defining the logical behavior common to all logics, and completes this basic set of rules by structural rules the application of which is controlled by *structurally atomic modes.*

For example, one may consider modes n and c that are non-associative/non-permutative and associative/permutative, respectively. One major difference with Moortgat & Oehrle's approach is thus that here a logic (in the sense of **L**, **NLP**, etc.) is defined as the set of basic rules *plus* a subset of the structural rules. There is no real base logic: All logics "peacefully coexist" in this system, and therefore any logic can be used to encode lexical information.

Whereas movement between logical systems was achieved in Moortgat & Oehrle's approach by making available 'extra-logical' behavior with respect to some base logic, we move -in Hepple's system- between logics by so-called linkage axioms. These are structural rules that specify the linkage between modes[3]. Consider for example the following three modes: n (no access to associativity nor to permutativity), a (access to associativity, but not to permutativity), and p (access to both associativity and to permutativity). Then, the next two axioms would enable us to use a instead of n (and vice versa)

$$\frac{A \circ_n B}{\overline{A \circ_a B}}$$

and to employ p instead of a (and vice versa)

$$\frac{A \circ_a B}{\overline{A \circ_p B}}$$

Thus, we could move from the $base + n$ to $base + a$ to $base + p$ to obtain more freedom, and from $base + p$ all the way back to $base + n$ to regain a more constrained regime.

Important about this movement is that Hepple understands the transition between modes as providing *more* information (cf. [14, 15]). $X \circ_o Y \Leftrightarrow Y \circ_o X$ is conceived of as indicating that both orderings are possible using a mode $\circ_o$ rather than that the ordering is unknown. Thus, modal interaction increases our information by elucidating *alternative possibilities*. What is particularly compelling about this view is that, consequently, multiple modes need not be individually stipulated, but can be made derivable from a small set of structural atomic modes.

Some final remarks concern modes, and the equivalence of the two approaches presented. When conceiving of modes, Hepple's approach stresses structure and combination, whereas Moortgat & Oehrle rather phrase their modes in terms of use of resources. However, logically speaking, both approaches can be proven to be equivalent - see Kurtonina's [26] and Kurtonina and Moortgat's [27].

[3]This is one subset of structural rules. The other structural rules, called interaction axioms, describe possible interactions between elements combined by the same modality; for example, associativity and permutativity are interaction axioms.

Chapter 3

Basic Dependency-Based Logical Grammar

1 Introduction

My aim in this chapter is to develop the basics of Dependency-Based Logical Grammar (DBLG), a MMLG-framework for dependency-based descriptions of natural language syntax. Succinctly put, dependency-based theories of linguistic grammar describe syntax of natural language in terms of different kinds of semantically motivated *dependency relations*, and *heads* and *dependents* that are relations by such dependency relations. The distinction between heads and dependents expresses a (directional) asymmetry: When standing in a dependency relation, a head is said to *govern* the dependent; or, conversely, the dependent *modifies* the head.

The head/dependent-asymmetry as such is found in various formal theories of linguistic grammar, not all being dependency-based: For example, Head-Phrase Structure Grammar (HPSG, [43]) takes the asymmetry as fundamental, and there have been a number of proposals to add the asymmetry to Categorial Grammar ([1, 33, 16]). Thus, only the distinction the head/dependent-asymmetry does not make a grammar dependency-based - we need to combine it with semantically motivated relations like **Actor, Manner**, etcetera, used instead of constituents/phrases to characterize sentential structure.

To specify how a word may act as a head, it is given a *valency frame* that specifies along what dependency relations it may be (or must be) modified. The notion of valency frame can be compared to for example Government & Binding's notion of θ-frame [7] or HPSG's subcategorization-list (though see section 4 below). What the valency frame essentially expresses is an n-ary relation that subcategorizes for dependents that can (be interpreted to) modify the head along the specified dependency relations.

Tesnière specified in his 1959 *Eléments de syntaxe structurale* dependency relations as acyclic, binary relations with the condition that each dependent modifies one and only head. A *dependency structure*, elucidating how dependents and heads are all connected by dependency relations (with the possibility that heads themselves act as dependents of an other head), is of the form of a *tree* - a *dependency tree.* Although a syntactic tree, a dependency tree is different from a phrase-structure tree in that there are no nonterminals/intermediate nodes in a dependency tree (which is an immediate consequence of the fact that a valency frame subcategorizes directly for dependents/dependency relations)[1].

In a *stratificational* approach to dependency-based grammar like FGD (Chapter 1), a description of syntax proceeds by explaining the relation between the *morphological form* of a word (or group of words) and the *function* it might be interpreted to have, and how functions fit together into valency frames. What is particularly stratificational here is that a distinction is made between the linguistic level at which morphological forms are observed, and the level at which functions are composed in dependency structures.

Let me fix some terminology for the remainder of the discussion:

- By a 'form' or 'wordform' I will understand a morphological form of a word, and the level at which we observe forms will be termed the 'surface dimension'. A sequence of forms (and possibly, function words like prepositions, [45]) I will call a 'sentence'.

- By a 'function', or a 'dependent', of kind δ I will understand a dependent that modifies a head along a dependency relation δ.[2] The level at which functions reside will be called the 'deep dimension', and it is there that dependency structures are composed.

If we want to formalize this conception of "stratificational dependency-based theory of natural language syntax" as a multimodal logical grammar, how do we proceed? Starting with the surface dimension, I will define the basics of a logical grammar enabling one to define simple (binary) modes ('surface modes') for forming larger groups of wordforms, and to express morphological form in terms of features. Categories found at this level are defined purely in terms of simple wordclasses and the residuals (slashes) of the surface modes.

Subsequently, I define the basics of a logical grammar for the deep dimension. Because I am interested in composing dependency structures using n-ary valency frames, the modes of composition here ('deep modes') will not be binary, but n-ary. Furthermore, the categories we are dealing with are formed around kinds of functions or dependents (as above) and the residuals

[1] I would like to argue that the difference is conceptual rather than fundamental: A phrase-structure tree is a derivation tree, showing how the tree can be constructed as a *process* of analysis, with all its intermediate steps - a dependency-tree on the other hand represents the *product* of an analysis, not the derivation itself. However, such a derivation tree can easily be obtained from a proof constructing a dependency-tree.

[2] Thus, whereas traditionally the dependency relation is used as a label to an arc in the dependency tree, we now use the dependency relation as a label (classification) of a node in the tree. Formally this makes no difference.

of the deep modes. It is the combination of these categories with composition modeling composition-by-valency-frame (i.e. head-oriented) that makes the approach dependency-based.

However, up to this point we have two separate logical grammars - one logical grammar defining well-formed structures of wordforms, and the other logical grammar defining well-formed dependency structures. The first grammar will not be able to form structures spanning the entire sentence (because such would require the presence of one or more valency frames - which are found in the deep dimension), whereas the functions that the second grammar manipulates have no relation (yet!) to the forms in the sentence. Consequently, even though we can talk about the well-formedness of structures found in each dimension, we are as yet incapable of analysing a sentence in terms of its dependency structure.

For that, we need to be able to relate forms and functions. Of importance thereby is that we make sure that the relation is *valid*: if the relation is valid, we obtain a valid way to relate valid structures of the surface dimension to valid structures of the deep dimension. Then, the *whole* process can be made valid - enabling us to make a judgment of the grammaticality of a sentence in terms of an underlying dependency structure. Multi-dimensional modal logic (MDML, [28]) provides us the tools to formalize this relation.

An overview of the chapter is as follows. In section 3 I define the logical grammar for the surface dimension, and in section 4 the logical grammar for the deep dimension. I propose a formalization of the relation between form and function in section 5, and discuss the notion of grammaticality that appears to arise from the approach.

Up to that point, I will have been concerned with inferring grammaticality rather than building representations (i.e. dependency structures). In the next chapter I discuss *term-assignment* in DBLG, which enables me to construct a dependency structure in parallel with the way in which the proof of a sentence's grammaticality proceeds.

To give the reader some insight in what DBLG will look like in terms of a calculus for analysis, let me present an example. The idea is more to give the reader a flavor of what is to come rather than to explain everything in meticulous detail.

2 Example

The Czech sentence to be analysed is the following one:

(3.1) *Ta* *malá* *kočka* *spala.*
The little kitten slept.
English: "The little kitten slept."

- Signature: n (noun), adj (adjective), s (verb), $DescrPr$ (Descriptive Property), and $Actor$ (Actor) are basic categories; $nom, fem, sg, defin, def, DEFIN$ and $past$ are features (modelled as unary modal operators). I presuppose a binary connective named s for composition at the surface, $\{\backslash, \bullet, /\}$, and an n-ary connective named d for composition in the deep dimension (dependency structures), $\{\times, \div\}$. Functional interpretation is written as $\otimes$.

- Lexicon: $\{$
  $(ta,\ [nom][fem][sg][def](n/_s n))$,
  $(malá,\ [nom][fem][sg]adj)$,
  $(kočka,\ [nom][fem][sg][DEFIN] \div \{DescrPr, \boxed{\mathrm{n}}\})$,
  $(spala,\ [fem][sg][past] \div \{Actor, \boxed{\mathrm{s}}\})$
  $\}$

1. The step-by-step derivation starts with the adjective, *malá*, that can be functionally interpreted as a Descriptive Property, given its form:

$$\frac{\text{malá} \vdash [nom][fem][sg]adj}{\text{malá} \vdash [nom][fem][sg]adj \otimes DescrPr}\ \otimes$$

2. We would like to see *malá* as a dependent of *kočka*, modifying *kočka* as a Descriptive Property. In order to do so, we first move the features to the resource-side, where they come to act as 'tags' that we use later to check agreement:

$$\frac{\text{malá} \vdash [nom][fem][sg]adj \otimes DescrPr}{(\text{malá})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle} \vdash adj \otimes DescrPr}\ *\Box^{\downarrow}E$$

3. Similarly, we move the features from the head-category in *kočka*'s 'valency frame' to the resource-side:

$$\frac{\text{kočka} \vdash [nom][fem][sg][DEFIN] \div \{DescrPr, \boxed{\mathrm{n}}\}}{(\text{kočka})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle} \vdash \div\{DescrPr, \boxed{\mathrm{n}}\}}\ *\Box^{\downarrow}E$$

4. Following n-ary residuation, the conclusions arrived at in steps 2 and 3 can be combined:

$$\frac{(\text{malá})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle} \vdash adj \otimes DescrPr \quad (\text{kočka})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle} \vdash \div\{DescrPr, \boxed{\mathrm{n}}\}}{((\text{malá})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle}, (\text{kočka})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle})^{\times} \vdash n}\ \div E$$

A remark should be made concerning the category derived (so far). Within a constituency-based grammar, we would have derived an np (noun phrase). However, DBLG is a *dependency*-based grammar. Consequently, the category only indicates the category of the *head* of the structure.

5. Now we should check whether the features of *malá* and *kočka* agree. Agreement in DBLG is modelled in terms of (valid) structures, rather than unifiability (as in constraint-based frameworks). Slightly rephrased, if two words carry the same feature, then (in the n-ary setting) we can move that feature back, onto the head (*head-dependent agreement*).

$$\frac{((\text{malá})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle}, (\text{kočka})^{\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle})^{\times} \vdash n}{(\text{malá}, (\text{kočka})^{\langle DEFIN\rangle})^{\times} \vdash [nom][fem][sg]n} *nAgr$$

Up to this point, however, we are left with a feature $DEFIN$ that does not appear on *malá*. This is not surprising: adjectives do not, linguistically speaking, have anything to do with definiteness, and therefore one could opt for simply not giving them such a feature. For the agreement between head and dependent, in other words, the feature does not matter. Since the feature belongs to the head, we can therefore move the feature back into the category without further ado.

$$\frac{(\text{malá}, (\text{kočka})^{\langle DEFIN\rangle})^{\times} \vdash [nom][fem][sg]n}{(\text{malá}, \text{kočka})^{\times} \vdash [nom][fem][sg][DEFIN]n} nHeadAD$$

6. We should now do something with the determiner. Determiners, in dependency-based grammar, are so-called *function words*. Function words themselves do not receive an individual node in a dependency tree - instead, they specify a feature of a node in that tree. In the case of a determiner, the noun's definiteness is specified.

In DBLG, [DEFIN] stands for an underspecified feature/value, which can be specified to either [def] (definite) or [indef] (indefinite). To have the noun agreeing with the determiner, we should therefore move (again...) the features to the resource-side, and then specify [DEFIN] as [def] to match with the determiner's [def].

$$\frac{(\text{malá}, \text{kočka})^{\times} \vdash [nom][fem][sg][DEFIN]n}{(\text{malá}, \text{kočka})^{\times}\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle \vdash n} *\Box^{\downarrow}E$$

The composition with the determiner,

$$\frac{\dfrac{\text{ta} \vdash [nom][fem][sg][def](n/_s n)}{(\text{ta})\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n/_s n} *\Box^{\downarrow}E \quad (\text{malá}, \text{kočka})^{\times}\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle \vdash n}{(\text{ta})\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \circ_s (\text{malá}, \text{kočka})^{\times}\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle \vdash n} /E$$

By inclusion we can change $DEFIN$ into def,

$$\frac{(\text{ta})\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \circ_s (\text{malá}, \text{kočka})^{\times}\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle DEFIN\rangle \vdash n}{(\text{ta})\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \circ_s (\text{malá}, \text{kočka})^{\times}\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n} Incl$$

After which all the features can be distributed (binary form of agreement),

$$\frac{(\mathrm{ta})\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \circ_s (\mathrm{malá,\ kočka})^\times\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n}{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n}\ Agr$$

and moved back to the category of the head,

$$\frac{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n}{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times) \vdash [nom][fem][sg][def]n}\ *\Box^\downarrow I$$

7. In terms of dependency trees, the tree formed so far has two nodes, corresponding to *malá* and *kočka*, whereby the latter governs the former. Within the resources, we find this back as the tuple (malá, kočka)$^\times$; the determiner *ta* is combined using a surface mode of composition and hence does not appear (as such) in the dependency structure (only its effect, namely the definite-ness of the nominal head, is noted in the structure).

The next step we take is to interpret the structure as an Actor, by the form of its head:

$$\frac{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times) \vdash [nom][fem][sg][def]n}{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times) \vdash [nom][fem][sg][def]n \otimes Actor}\ \otimes$$

8. The last part of the analysis is concerned with binding the Actor to the verbal head. Again we have to check for agreement, whereby this time we -also- find a feature on the *dependent* that is irrelevant for head-dependent agreement. An asymmetric distribution rule like $[nHeadAD]$ will be used - $[nDepAD]$.

$$\frac{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times) \vdash [nom][fem][sg][def]n \otimes Actor}{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n \otimes Actor}\ *\Box^\downarrow E$$

$$\frac{spala \vdash [fem][sg][past] \div \{Actor, \boxed{s}\}}{(spala)\langle fem\rangle\langle sg\rangle\langle past\rangle \vdash \div\{Actor, \boxed{s}\}}\ *\Box^\downarrow E$$

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{(\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle \vdash n \otimes Actor \quad (spala)\langle fem\rangle\langle sg\rangle\langle past\rangle \vdash \div\{Actor, \boxed{s}\}}{((\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle nom\rangle\langle fem\rangle\langle sg\rangle\langle def\rangle, (spala)\langle fem\rangle\langle sg\rangle\langle past\rangle)^\times \vdash s}\ \div E}{((\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle fem\rangle\langle sg\rangle\langle def\rangle, (spala)\langle fem\rangle\langle sg\rangle\langle past\rangle)^\times \vdash s}\ nDepAD}{((\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times)\langle def\rangle, (spala)\langle past\rangle)^\times \vdash [fem][sg]s}\ *nAgr}{((\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times), (spala)\langle past\rangle)^\times \vdash [fem][sg]s}\ nDepAD}{((\mathrm{ta} \circ_s (\mathrm{malá,\ kočka})^\times), (spala))^\times \vdash [fem][sg][past]s}\ nHeadAD$$

9. A labelled deductive system can be defined, where propositions of the form $R \vdash C$ get labelled with a term S. The term is an interpretation of the derivation over the categories, making use of a formula-as-types correspondence via a suitably defined lambda-calculus. (Namely, a directional lambda calculus in which binary and n-ary slashes are distinguished.)

Without going into detail how term-assignment works exactly: The term derived for *malá kočka* is $(\lambda^l y.\{DescrPr : y\}\text{kočka})(\text{malá})$, and the term assigned to *spala* is $(\lambda^l x.\{Actor : x\}\text{spala})$. Subsequently, when composing *ta malá kočka spala*, the entire term becomes

$$(\lambda^l x.\{Actor : x\}\text{spala})((\lambda^l y.\{DescrPr : y\}\text{kočka})(\text{malá}))$$

which reduces to

$$(\{Actor : ((\lambda^l y.\{DescrPr : y\}\text{kočka})(\text{malá}))\}\text{spala})$$

which finally reduces to

$$(\{Actor : (\{DescrPr : \text{malá}\}\text{kočka})\}\text{spala})$$

which is what we call the *dependency structure.*

3 The Surface Dimension

The idea of a logical grammar for the surface dimension is, for one, to provide the means to bind function words like prepositions and determiners to nominal heads, and auxiliaries to verbal heads. Furthermore, I want to be able to to characterize the morphological form of a wordform, and to this aim I introduce "features". The categories that can be formed are defined as follows.

Definition 3.1 (Surface Categories) *The set of* surface categories $\mathfrak{U}_{surf}$ *is defined over a finite, non-empty set of basic categories* $\mathfrak{B}$ *as follows: (1) All the basic categories from* $\mathfrak{B}$ *are categories. (2) If* A *and* B *are categories, and* i *is a surface modes of composition, then* $A/_iB$ *and* $B\backslash_iA$ *are categories. (3) If* f *is a morphological feature and* A *a category, then* $\langle f\rangle A$ *and* $[f]A$ *are categories. (4) Nothing else is a surface category.* ■

Remark 3.1 Because I am developing a dependency-based approach, I take $\mathfrak{B}$ to include categories like n, adj - clearly, phrases like *pp* and *np* do not occur in $\mathfrak{B}$.

3.1 Surface Modes

The framework in which surface modes can be defined is based on the idea that composition at the surface, resulting into (larger) groups of wordforms, is essentially *binary* in nature. Consequently, surface modes follow the basic rules of residuation.

Notation 3.1 *To indicate what mode is being used, the name of the mode will appear as* subscript *to products and slashes - thus,* $A \circ_\alpha B$ *means that* A *and* B *have been combined using a mode* α*. When no mode in particular is meant, subscripts* i *or* j *are usually used.*

Definition 3.2 (Binary Residuation) *Residuation for binary modes of composition is defined as follows:* $A \rightarrow C/_i B$ *iff* $A \circ_i B \rightarrow C$ *iff* $B \rightarrow A \backslash_i C$. *That is, the composition of* A *and* B, $A \circ_i B$, *has category* C, *which means that* A *needs a* B *to its right to form* C $(A \rightarrow C/_i B)$ *or* B *an* A *to its left to result in* C $(B \rightarrow A \backslash_i C)$. ■

Remark 3.2 Binary residuation only works for binary modes of composition. For the deep dimension, where I will use n-ary modes of composition rather than binary modes, the notion of residuation will thus have to be generalized to the n-ary case.

Definition of surface modes then proceeds by giving their intended semantics and the proof rules that model the behavior of modes. Because the proof rules essentially follow the intended semantics of modes, the semantics lend a seal of *validity* to inferences employing the proof rules.

Definition 3.3 (Semantics for surface modes) *To define the intended semantics of surface modes, first of all frames of the kind* $\mathfrak{F}_{surf} = \langle \mathfrak{U}_{surf}, \mathcal{R}^3 \rangle$ *are introduced. A frame* $\mathfrak{F}$ *takes the universe of surface categories as domain, and an accessibility relations* $\mathcal{R}^3$ *that models composition in the surface dimension. Subsequently, a model* $\mathfrak{M}_i$ *is defined for a surface mode* i, *taking* $\mathfrak{F}_{surf}$ *and a valuation function* $\mathcal{V}$. *This model defines the intended semantics of the mode* i *by specifying the valuation of structures built using* i *(using pre- and postconditions on the accessibility relation* $\mathcal{R}^3$*):*

(3.2) $\mathcal{V}(A \circ_i B) = \{x | \exists x \exists y [\mathcal{R}xyz \wedge y \in \mathcal{V}(A) \wedge z \in \mathcal{V}(B)]\}$

(3.3) $\mathcal{V}(C/_i B) = \{y | \forall x \forall z [(\mathcal{R}xyz \wedge z \in \mathcal{V}(B)) \Rightarrow x \in \mathcal{V}(C)]\}$

(3.4) $\mathcal{V}(A \backslash_i C) = \{z | \forall x \forall y [(\mathcal{R}xyz \wedge y \in \mathcal{V}(A)) \Rightarrow x \in \mathcal{V}(C)]\}$

For basic categories $b \in \mathfrak{B}$, $\mathcal{V}(b)$ *assigns subsets of* $\mathfrak{U}_{surf}$. ■

Where it concerns the formulation of the proof theory, a choice can be made for Hepple's approach ("hybrid categorial logics", [14, 15]) or the approach advocated by Moortgat & Oehrle ("multimodal categorial grammar",[32]). Although Kurtonina proved that both approaches are formally equivalent [26], they bring about different takes on linguistic description.

I shall employ Hepple's ideas here. As such, the proof theory will consists of a logic that specifies the behavior common to all modes of composition, and structural rules that define additional behavioral characteristics. The structural rules themselves are divided into *mode-internal axioms* that define how two structures composed using the same mode interact, *interaction axioms* that define interaction between two structures composed using different modes, and *linkage axioms* that specify how one mode can be replaced by another mode. As a matter of fact, the structural rules are *schemata* that need to be instantiated for individual modes or combinations of modes. The behavior of a mode is then defined in terms of the common logic plus which structural rules are instantiated for this mode.

The take on linguistic description that arises out of Hepple's approach is that the category of a wordform, as found in the lexicon, already specifies which mode is (or modes, are) to be employed when trying to combine the word into a larger structure. A result of this is that the proof theory is a system in which only the behavior of the individual modes needs to be described, a system made hybrid by defining how modes interact with one another[3].

Notation 3.2 *The format of the proof rules is that of a natural deduction system (later I will turn this into a proper, labelled natural deduction system in the sense of [6], see Chapter 4) using propositions of the form $R \vdash C$, with R called the* resources *and C the* category. *Essentially R is a syntactic structure in terms of words and their modes of composition, whereas C is the category assigned to that structure. Peculiar to a natural deduction system is that it allows for assumptions (or hypotheses) to be used in an inference [48]. Whenever an $R \vdash C$ is assumed, it is enclosed in square brackets: $[R \vdash C]$.*

Definition 3.4 (Proof Theoretical Syntax) ***Common Behavior***

(1) $$\frac{s \vdash A/_i B \quad t \vdash B}{(s \circ_i t) \vdash A}\ /_i E \qquad \frac{[v \vdash B] \quad (s \circ_i v) \vdash A}{s \vdash A/_i B}\ /_i I$$

(2) $$\frac{t \vdash B \quad s \vdash B \backslash_i A}{(t \circ_i s) \vdash A}\ \backslash_i E \qquad \frac{[v \vdash B] \quad (v \circ_i s) \vdash A}{s \vdash B \backslash_i A}\ \backslash_i I$$

(3) $$\frac{\begin{array}{c}[v \vdash B] \quad [w \vdash C] \\ s[v \circ_i w] \vdash A\end{array} \qquad t \vdash B \circ_i C}{s[t] \vdash A}\ \circ_i E \qquad \frac{s \vdash A \quad t \vdash B}{(s \ \circ_i \ t) \vdash A \circ_i B}\ \circ_i I$$

Additional structural rule schemata can concern mode-internal behavior, linkage, and interaction. The most common mode-internal rule schemata are those defining associative and commutative behavior, and are given below.

[3]Without going into too much detail -see [14] for more discussion-, the difference with Moortgat & Oehrle's approach is that in their approach, a (relatively strong) base logic is favored in which the lexical categories are primarily formulated. In order to bring about the possibility for different kinds of composition then, the proof theory needs to specify all sorts of distinguishable contexts in which a move from the base logic can be made to other logics, defining different behavior of composition. Consequently, lexical categories are relatively simple, but the proof theory becomes rather complex due to the need to define the various contexts.

Mode-Internal (Associativity (a) and Commutativity (p)

$$(4)\quad \frac{s[((x \circ_i y) \circ_i z)] \vdash A}{s[(x \circ_i (y \circ_i z))] \vdash A}\, a$$

$$(5)\quad \frac{s[(x \circ_i y)] \vdash A}{s[(y \circ_i x)] \vdash A}\, p$$

Linkage

$$(6)\quad \frac{s[(x \circ_i y)] \vdash A}{s[(x \circ_j y)] \vdash A}\, i/j$$

■

Remark 3.3 Note that I have left out interaction rules from the definition. The reason is that these rule schemata are usually of a more specific form than the rule schemata given above. Furthermore, nothing stops us of course from defining, in a particular fragment, more structural rule schemata.

3.2 Features

Heylen presents in [18] an approach to encode morphological features in a multimodal logical grammar, using unary modal operators. The basic idea is to mark lexical categories with 'boxes' identifying feature-values. For example, the Czech noun "kobliha" (*En.*donut) could get as lexical category $[sg][fem][nom]n$, meaning a *singular, feminine, nominative* noun[4].

The way these 'boxed' categories are used in a proof can be sketched as follows. Recall that DBLG -so far- employs a labelled deductive system of the kind $R \vdash C$, meaning that a surface structure R implies a particular category C. As a proof proceeds by combining categories, we effectively "enlarge" the surface structure that is being covered. The idea is of course to continue all the way to having covered the entire sentence.

Logical rules for 'boxes' appearing in categories allow us to transfer the feature classification from the category to the surface form. Thus, the surface form gets an explicit, morphological tag. Other logical rules then control the combination of wordforms by means of these tags: Only wordforms of "agreeing" morphology can be combined.

An important point regards the "agreement", though. DBLG is a logical grammar, in which we are trying to prove a structure - turning around a notion of *validity*. Such is distinct from the usual way in which we deal with

[4]See the appendix for a list of all the dependency relations and features used in this manuscript

features, which is by *satisfaction*. The way we deal with features here is by means of proving the validity of putting two (morphologically tagged) structures together[5].

Definition 3.5 (Unary Modals for Features) *A feature $\mathcal{F}$ appears in a lexical category as a box $[\mathcal{F}]$. A box $[\mathcal{F}]$ follows the more general logical definition of $\Diamond_i$ and $\Box_i^{\downarrow}$: by residuation, $\Diamond_i A \rightarrow B$ iff $A \rightarrow \Box_i^{\downarrow} B$.* DBLG *employs subsequent kinds of rules to deal with features:*

$$\frac{A \vdash [i]B}{(A)^{\langle i \rangle} \vdash B}\ \textit{Box elimination} \qquad \frac{(A)^{\langle i \rangle} \vdash B}{A \vdash [i]B}\ \textit{Box introduction} \qquad \frac{(A)^{\langle i \rangle} \vdash B}{(A)^{\langle j \rangle} \vdash B}\ \textit{Inclusion} \qquad \frac{(A)^{\langle i \rangle} \circ_k (B)^{\langle i \rangle} \vdash C}{(A \circ_k B)^{\langle i \rangle} \vdash C}\ \textit{Distribution} \qquad \frac{(A)^{\langle j \rangle \langle i \rangle} \vdash B}{(A)^{\langle i \rangle \langle j \rangle} \vdash B}\ \textit{Commutativity}$$

■

Remark 3.4 These rules (and schemas[6]) are easily explained. Box elimination enables the transfer of a feature appearing in the category to be used as a tag in the surface form - whereas box introduction transfers a tag (constructed in the surface form) into the category. Inclusion allows for specifying generalizations over features in the categories. Commutativity makes the order in which tags appear, irrelevant. And distributivity enables us to say that if two wordforms (or groups of wordforms) each have the same tag, they "share" that tag - which is how we model *agreement*.

3.3 Agreement

Above it was already mentioned that, due to the proof-oriented nature of logical grammar, we are oriented at validity of (inferences over) structures. Consequently, agreement is formalized in a structural way: Two terms are in agreement if their structures have the same logical form. That does not make the approach particularly different from unification formalisms, though: By distributivity we obtain the same effect as what is otherwise known as "feature percolation".

In an abstract form, agreement is:

$$\frac{(A)^{\langle agr \rangle} \circ_j (B)^{\langle agr \rangle} \vdash C}{(A \circ_j B)^{\langle agr \rangle} \vdash C}$$

Subsequently, we can instantiate the Distributivity schema for the features we want to distinguish, for example the cases *nom,acc*, number *sg,pl*,

[5]That is not to say that one cannot employ satisfaction-by-unification as a means to control features in a categorial grammar: cf. Steedman's CCG or Kraak's [25].

[6]A note: The inclusion and the distribution "rules" are in fact a schemata that need to be instantiated for couples of features (for example, *case* includes *nom* (nominative), but not *fem* (feminine)).

and gender *fem,mas*. An instantiation for *fem*

$$\frac{(A)^{\langle fem\rangle} \circ_j (B)^{\langle fem\rangle} \vdash C}{(A \circ_j B)^{\langle fem\rangle} \vdash C}$$

can then be used, for example, in the combination of the determiner "la" with the noun "casa".

Remark 3.5 The context in which distribution should take place can be restricted by instantiating the schema for particular modes of composition only - enabling one to exclude, for example, coordination constructions (that would be formed by a particular mode of composition, for which the distribution schema thus would *not* be instantiated).

3.4 Lexical Underspecification

Lexical *underspecification* is used in grammars like HPSG to reduce the number of lexical assignments - thereby expressing, in a sense, generalizations. Besides the possibility to reduce the number of entries needed in a lexicon, Heylen also observes in [18] that not all morphological distinctions are relevant in every context. For example, the number of a verb's object is usually irrelevant, so that both the combination of the verb with a singular object and with a plural object will be judged grammatical.

Part of what we need for lexical underspecification has already been introduced above, namely the Inclusion schema. The Inclusion schema enables us to specify how we can go from a more general feature to a more specific feature. Thus, what we are left with is providing the general features, and the proper instantiations of the Inclusion schema.

Obvious candidates for general features are *case*, *num*, and *gen* for case, number, and gender, respectively. Proper instantiations, given the features *nom, acc, sg, pl, fem*, and *mas* introduced earlier, would then be: *nom* or *acc* from *case*, *sg* or *pl* from *num*, and fem or *mas* from *gen*. For example, specified in rule-format, the inclusions

$$\frac{(A)^{\langle case\rangle} \vdash C}{(A)^{\langle nom\rangle} \vdash B} \quad \frac{(A)^{\langle case\rangle} \vdash C}{(A)^{\langle acc\rangle} \vdash B}$$

enable us to use just one entry for "Frau", which is the morphological form of both the nominative as well as accusative, feminine singular noun: $[case][fem][sg]n$. In the proper context, the $[case]$ feature can be employed as an underspecified tag and subsequently specified to either *nom* or *acc* as needed.

4 The Deep Dimension

In the previous sections I defined the necessary apparatus to deal with features and the construction of groups of wordforms. In the current section I will be concerned with developing the part dealing with building dependency structures - *deep composition.* To do so I take an approach of gradual refinement, starting with a simplified notion of valency frame (equalling GB's θ-frame). After having defined the basic ideas of deep composition, I extend the notion of valency frame to cover the Praguian intuitions [45]. This will lead to additional structural rules for the proof syntax, and the introduction of decorations to constrain the applicability of structural rules. Finally, agreement at the deep level is discussed.

4.1 Valency Frames as θ-frames

The introduction already mentioned the notion of valency frame. A valency frame for a word specifies how that word may act as a head, in the sense of by what dependency relations it may be modified. Similar notions are GB's θ-frame and HPSG's subcategorization list. For the moment I will understand a valency frame to be exactly like a θ-frame. Thus, it specifies by what dependency relations the word-as-head *must* be modified, and that it must be modified by each dependency relation *once and only once*[7].

To model a valency frame in a categorial setting, several things need to be done. First, we need categories that mirror the idea of a dependency relation. As I already pointed out in the introduction, a dependent modifying a head by dependency relation δ can be taken to be a dependent of kind (i.e. category) δ, without loss of generality. Second, since a valency frame is n-ary, (and -as a result- dependency trees n-ary branching rather than binary branching by definition), we need an n-ary mode of composition to model a valency frame in a categorial logic.

To begin with the first point, let me give a definition of the deep categories. The definition is preliminary in that features are not yet included, nor decorations to be introduced later (see sections 4.3 and 4.4 below, respectively).

Definition 4.1 (Preliminary Deep Categories) *The set of* deep categories $\mathfrak{U}_{deep}$ *is defined over a finite, non-empty set of basic categories* $\mathfrak{B}_{deep}$ *as follows: (1) All the basic categories from* $\mathfrak{B}_{deep}$ *are categories. (2) If* A *and* B *are categories, and* i *is a deep modes of composition, then* $A/_iB$ *and* $B\backslash_i A$ *are categories. (3) Nothing else is a deep category.* ■

Remark 4.1 The set of basic categories $\mathfrak{B}$ in the above definition includes

[7]The valency frame, under this perspective, is exactly like a predicate, with relations as arguments.

the category s for verbal head, and semantically motivated dependency relations like Actor, Patient, Location, etcetera. A complete list of dependency relations is given in [42]. (See also the appendix.)

4.2 Deep Modes

The idea of an n-ary mode of composition, based on n-ary residuation, can be formalized by a rather straightforward generalization from the binary case, as Moortgat showed in [31].

Definition 4.2 (n-ary Composition and Residuation) *We can write an n-ary mode of composition i as $\times_i^n\{\mathcal{C}_1, ..., \mathcal{C}_n\}$ and its n residuals (for each place j) as $\div_i^n\{\mathcal{C}_1, ..., \mathcal{C}_{j-1}, \boxed{\mathcal{C}}, \mathcal{C}_{j+1}, ..., \mathcal{C}_n\}$. A residual has the resulting category in its j-th place (the $\boxed{\mathcal{C}}$) whereas the other categories are the arguments. Given the location j of the resulting category $\boxed{\mathcal{C}}$, the arguments $\mathcal{C}_1, ..., \mathcal{C}_{j-1}$ are understood to be filled by matching categories occurring to the left, and the arguments $\mathcal{C}_{j+1}, ..., \mathcal{C}_n\}$ by matching categories to the right. The n-ary residuation laws are defined as follows:*

$$\times_i^n\{\mathcal{C}_1, ..., \mathcal{C}_n\} \textit{ if and only if } \div_i^n\{\mathcal{C}_1, ..., \mathcal{C}_{j-1}, \boxed{\mathcal{C}}, \mathcal{C}_{j+1}, ..., \mathcal{C}_n\}$$

■

All the deep modes of composition will be of the above defined n-ary kind. Analogously to their binary surface brethren, they are given an intended meaning in terms of a frame-based semantics.

Definition 4.3 (Semantics of Deep Modes) *To define the intended semantics of deep modes, first of all frames of the kind $\mathfrak{F}_{deep} = \langle \mathfrak{U}_{deep}, \{\mathcal{R}^n\}\rangle$ are introduced. A frame $\mathfrak{F}$ takes the universe of deep categories as domain, and has a family of n-ary accessibility relations $\mathcal{R}^n$ that model composition in the deep dimension (such that for a k-ary mode of a composition, $k \geq 1$, there is an accessibility relation $\mathcal{R}^k$). Subsequently, a model $\mathfrak{M}_i$ is defined for a deep mode i, taking $\mathfrak{F}_{deep}$ and a valuation function $\mathcal{V}$. This model defines the intended semantics of the k-ary mode i by specifying the valuation of structures built using i (using pre- and postconditions on accessibility relation $\mathcal{R}^k$, which has an inverse $\mathcal{R}^{-k}$ defined such that $\mathcal{R}^{-k}y_iy_1...x...y_k$ if and only if $\mathcal{R}^kxy_1...y_i...y_k$). For basic categories $b \in \mathfrak{B}$, $\mathcal{V}(b)$ assigns subsets of $\mathfrak{U}_{deep}$.*

(3.5) $\mathcal{V}(\times_i^k\{A_1, \ldots, A_k\}) = \{x | \exists y_1 \ldots y_k(R^kxy_1 \ldots y_k \;\&\; y_1 \in \mathcal{V}(A_1) \;\&\; \ldots \;\&\; y_k \in \mathcal{V}(A_k)\}$

(3.6) $\mathcal{V}(\div_i^k\{A_1, \ldots, A_k\}) = \{\forall y_1 \ldots y_k((R^{-k}xy_1 \ldots y_k \;\&\; y_{j(j\neq i)} \in \mathcal{V}(A_j)) \Rightarrow y_i \in \mathcal{V}(A_i)\}$

■

Finally, I give the proof theoretic syntax for deep modes. For the moment, I only present the proof rules describing common behavior (i.e. introduction and elimination of $\times$ and $\div$). Further extensions will be presented in subsequent chapters, after term assignment has been introduced in Chapter 4, and the relation between the surface dimension and deep dimension has been worked out in further detail.

Notation 4.1 *Because the modes of composition considered here are n-ary, I use a slightly different notation in the resources to indicate that words s_1 .... s_n have been combined into a structure using mode i: $(s_1, ..., s_n)^{\times_i^n}$.*

Definition 4.4 (Basic Proof Syntax for Deep Modes) *The **Common Behavior** of the deep modes of composition is defined by the following proof rules, specifying introduction and elimination of $\times_i$ and $\div_i$ for a mode i.*

$$\frac{s_1 \vdash C_1 \cdots s_{m-1} \vdash C_{m-1} \quad t \vdash \div_i^n\{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\} \quad s_{m+1} \vdash C_{m+1} \cdots s_n \vdash C_n}{(s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_i^n} \vdash B} \div_i E$$

$$\frac{[s_1 \vdash C_1 \cdots s_{m-1} \vdash C_{m-1}] \quad (s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_i^n} \vdash B \quad [s_{m+1} \vdash C_{m+1} \cdots s_n \vdash C_n]}{t \vdash \div_i^n\{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\}} \div_i I$$

$$\begin{array}{l}[s_1 \vdash C_1] \cdots [s_n \vdash C_n] \\ \dfrac{s[(s_1, ..., s_n)^{\times_i^n}] \vdash A \quad t \vdash \times_i^n(C_1, .., C_n)}{s[t] \vdash A} \times_i E\end{array}$$

$$\frac{s_1 \vdash C_1 \cdots s_n \vdash C_n}{(s_1, , ..., s_n)^{\times_i^n} \vdash \times_i^n(C_1, ..., C_n)} \times_i I$$

■

4.3 Extending the Notion of Valency Frame

Understanding a valency frame as being a θ-frame is very restrictive, and certainly does not correspond to the more elaborate ideas prevalent in FGD [45]. My motivation for initially employing a restricted notion was that the simple proof syntax given above is capable of dealing with such valency frames. In the current section I will extend the understanding of a valency frame in DBLG so as to cover intuitions found in FGD, and present the more complex proof syntax needed to model that extended understanding.

Recall that in FGD, dependency relations can be classified (with respect to a head) along two dimensions (cf. Chapter 1):

1. **Obligatory/optional**: whether the head *must* (obligatory) or *may* (optional) be modified along this dependency relation;

2. **Inner Participant/Free Modifier**: whether the head can be modified by the dependency relation at most once (inner participant), or more than once (free modifier).

The distinction inner participant/free modifier should perhaps be specified more precisely. If a dependency relation δ is classified as an inner participant of a head, then there can only be one dependent modifying the head along that dependency relation. Or, in terms of DBLG - there is at most one δ-dependent. On the other hand, if the dependency relation δ is a free modifier, then there may be *several* δ-dependents. (One can easily visualize this as a tree in which there are several edges labelled δ, all related to the head-node.)

Consequently, how are valency frames presented in FGD? In [45], a valency frame gives *all* the dependency relations by which a head can be modified, and specifies for each dependency relation whether it is an inner participant or a free modifier, and whether it is obligatory or optional. A slightly different formulation is presented by Panevová in for example [38] (recapitulating on her work during the last two decades). On her account, a valency frame only includes the obligatory and optional inner participants, and the obligatory free modifiers. The optional free modifiers are left out, the reason being that (notably) for verbal heads these are common to large classes of verbs. The proof syntax formulated here will model the kind of valency frames of [45].

Given the proof syntax of definition 4.4, which -as said- only models the idea of a valency frame as θ-frame, the following extensions need to be made to the proof syntax. First of all, the idea of modelling a valency frame using an n-ary mode of composition remains of course, but the categories appearing as arguments will be decorated in the following way: if a category (dependency) is a free modifier, it receives the decoration $\circledast$, whereas an inner participant is decorated as $\circledcirc$, and if a category (dependency) is optional it is decorated with a $\boxminus$, whereas an obligatory category is decorated with $\boxplus$. To make sure decoration is done properly, we require that for every category (dependency) Δ appearing in an n-ary mode (modelling a valency frame)

$$\Delta^{ij} \text{ whereby } i \in \{\boxminus, \boxplus\} \text{ and } j \in \{\circledast, \circledcirc\}.$$

Furthermore, by definition , $\Delta^{ij} \equiv \Delta^{ji}$, so it does not matter in which order the decorations appear on a category.

Secondly, rules are needed that enable one

1. to modify a head multiple times by one and the same dependency relation, if and only if that dependency relation has been marked as

a free modifier (and there are indeed multiple dependents that can be taken to modify the head by that dependency relation)

2. to get rid of optional, unused category (dependency).

Remark 4.2 It is easy to see why these two rules suffice: The basic proof syntax models obligatory inner participants, the rule under (1) models free modifiers (either obligatory or optional), and the rule under (2) models optionality. The observant reader will notice that the rule under (2) overtly changes the arity of the mode of composition: "getting rid" of a category makes the n-ary mode into an $(n-1)$-ary mode[8].

Definition 4.5 presents these two rules. I first fix some notation to facilitate a slightly more perspicuous formulation.

Notation 4.2 *As is common in natural deduction system, $[s_i \vdash C_i]$ stands for the assumption of $s_i \vdash C_i$ in the antecedent. I extend this notation to the resources, where I will write $[s_i]$ to indicate that s_i was included in the resources on assumption.*

Furthermore, $\mathcal{A}_{left}$ and $\mathcal{A}_{right}$ stand for the sets of assumptions $\{[s_i \vdash C_i]\}$ and $\{[s_j \vdash C_j]\}$ whereby $i \in [1..m-1], j \in [m+1..n]$ (for the meaning of m-1 and m+1, refer back to definition 4.4). For the antecedents that are appear as not *assumed, it holds that they are indeed not in $\mathcal{A}_{left}$ nor in $\mathcal{A}_{right}$.*

Finally, by $\mathcal{A}_x^{\vee}$ I mean the set of all the resources that appear assumed, and by a resource of the form $(Lft, t, Rght)^{\times_i^n}$ a resource where t is combined to the left with Lft and to the right with $Rhgt$ (both possibly including assumed resources) by means of mode $\times_i^n$. Then, a resource of the form $(Lft - \mathcal{A}_{left}^{\vee}, t, Rght - \mathcal{A}_{right}^{\vee})^{\times_i^{n-k}}$ means the resource with all k assumed elements removed.

Definition 4.5 (Extended Proof Syntax) *To the basic proof syntax of the deep dimension, as given in definition 4.4, the following structural rules are added.*

Optional Category Deletion
Given an proof step employing $[\div_i^n E]$ using one or more assumptions,

$$\frac{\overset{\mathcal{A}_{left}}{\{s_i \vdash C_i\}} \quad t \vdash \div_i^n\{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\} \quad \overset{\mathcal{A}_{right}}{\{s_j \vdash C_j\}}}{(Lft, t, Rght)^{\times_i^n} \vdash B} \div_i E$$

where for every assumption of the form $s_h \vdash C_h$ it holds that C_h is decorated with $\boxminus$ in the category of t. Then, structural rule $[OptDel]$ allows the

[8] Similarly, we may want to say that the rule under (1) *covertly extends* the arity.

deletion of the assumptions from the resource:

$$\dfrac{\begin{matrix}\mathcal{A}_{left} \quad \cdots \quad \mathcal{A}_{right} \\ \vdots \\ (Lft, t, Rght)^{\times_i^n} \vdash B\end{matrix}}{(Lft - \mathcal{A}^{\vee}_{left}, t, Rght - \mathcal{A}^{\vee}_{right})^{\times_i^{n-k}} \vdash B}\ OptDel$$

$\div E$ ***with Free Modifier Extension***

$$\dfrac{\begin{matrix}s_1 \vdash C_1 \cdots s_{m-1} \vdash C_{m-1} \quad t \vdash \div_i^n\{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\} \\ s_{m+1} \vdash C_{m+1} \cdots s_n \vdash C_n\end{matrix}}{(s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_i^{n+k}} \vdash B}\ FM - \div E$$

where, for every $s_i \vdash C_i$ *with* C_i *marked as* $\circledast$ *in the category of* $t \vdash \div_i^n\{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\}$, *there may be another* $s_i' \vdash C_i$ *appearing next to* s_i.

■

Remark 4.3 The idea behind [*OptDel*] is to complete a category by assuming the optional categories as present, and then drop the assumptions without that having an effect on the category. (Compare that to [$\div I$] in definition 4.4.) The Free Modifier Extension is, in fact, an extended version of [$\div E$]. The difference is that there may be a sequence of occurrences of resources implying a category C_i, (all *directly* following one another), rather than a single resource $s_i \vdash C_i$ - with C_i decorated with a $\circledast$ in the category of the head.

4.4 Agreement Between Heads and Dependents

4.5 Features and Agreement

In [18, 19] and (p.c.) Heylen discusses how features can be modelled in a categorial framework by means of unary modalities (in the sense of [32]). Agreement of features is formalized by means of distribution laws - for the simplest case, this says intuitively that two structures agree on a feature if they both have that feature, so that the feature can be distributed over (assigned to) the composition of the two structures. For example, consider $\langle f \rangle$ a feature, then

$$\dfrac{A^{\langle f \rangle} \circ_i B^{\langle f \rangle} \vdash C}{(A \circ_i B)^{\langle f \rangle} \vdash C}\ Dist$$

after which the feature $\langle f \rangle$ can be attached to the resulting category C by the standard definition of unary modalities [32]:

$$\frac{(A \circ_i B)^{\langle f \rangle} \vdash C}{(A \circ_i B) \vdash [f]C}\ Check$$

The above laws consider the case which is *symmetric* in that features from both A and B are distributed, and then checked. Similarly, we can consider cases which are *asymmetric*, in that the features of either A or B are distributed and then checked on the resulting category. Examples of the symmetric case (3.7) and two asymmetric cases (3.8-3.9) are the following (by Dirk Heylen):

$$(3.7)\qquad \cfrac{\cfrac{\cfrac{\ldots}{(\Box^{\downarrow}(X/Y))^{\diamond} \circ (\Box^{\downarrow}Y)^{\diamond} \Rightarrow X}}{(\Box^{\downarrow}(X/Y) \circ \Box^{\downarrow}Y)^{\diamond} \Rightarrow X}\ Dist}{\Box^{\downarrow}(X/Y) \circ \Box^{\downarrow}Y \Rightarrow \Box^{\downarrow}X}\ Check$$

$$(3.8)\qquad \cfrac{\cfrac{\cfrac{\ldots}{(\Box^{\downarrow}(X/Y))^{\diamond} \circ Y \Rightarrow X}}{(\Box^{\downarrow}(X/Y) \circ Y)^{\diamond} \Rightarrow X}\ Dist}{\Box^{\downarrow}(X/Y) \circ Y \Rightarrow \Box^{\downarrow}X}\ Check$$

$$(3.9)\qquad \cfrac{\cfrac{\cfrac{\ldots}{X/Y \circ (\Box^{\downarrow}Y)^{\diamond} \Rightarrow X}}{(X/Y \circ \Box^{\downarrow}Y)^{\diamond} \Rightarrow X}\ Dist}{X/Y \circ \Box^{\downarrow}Y \Rightarrow \Box^{\downarrow}X}\ Check$$

The interest in asymmetric distribution is that, linguistically speaking, not all features are always relevant to consider for agreement - or there need not be any features at all on a complement, whereas (still) the resulting category should carry the features we started out with.

Finally, let us consider the following example:

$$(3.10)\qquad \cfrac{\cfrac{\cfrac{\cfrac{\overline{X \Rightarrow X} \quad \cfrac{\cfrac{\overline{Y \Rightarrow Y}}{(\Box^{\downarrow}Y)^{\diamond} \Rightarrow Y}}{\Box^{\downarrow}Y \Rightarrow \Box^{\downarrow}Y}}{X/\Box^{\downarrow}Y \circ \Box^{\downarrow}Y \Rightarrow X}}{(\Box^{\downarrow}(X/\Box^{\downarrow}Y))^{\diamond} \circ \Box^{\downarrow}Y \Rightarrow X}}{(\Box^{\downarrow}(X/\Box^{\downarrow}Y) \circ \Box^{\downarrow}Y)^{\diamond} \Rightarrow X}}{\Box^{\downarrow}(X/\Box^{\downarrow}Y) \circ \Box^{\downarrow}Y \Rightarrow \Box^{\downarrow}X}$$

As Heylen notes, this is a combination of (3.8) and the following:

$$(3.11)\qquad \cfrac{\overline{X \Rightarrow X} \quad \cfrac{\cfrac{\overline{Y \Rightarrow Y}}{(\Box^{\downarrow}Y)^{\diamond} \Rightarrow Y}}{\Box^{\downarrow}Y \Rightarrow \Box^{\downarrow}Y}}{X/(\Box^{\downarrow}Y) \circ \Box^{\downarrow}Y \Rightarrow X}$$

In (3.10) it appears that Y is specified twice for the feature $\Box^{\downarrow}$ - namely, once by the functor $\Box^{\downarrow}(X/\Box^{\downarrow}Y)$ as a whole, and once as the argument within that functor, $\Box^{\downarrow}Y$. Heylen argues that such double specification is not particularly elegant, and proposes to use (3.7) instead to achieve the same effect[9]. As we will see in the next section, the example illustrated by (3.10) does become an issue in case of n-ary composition, in which it is undesirable to revert to (3.7).

4.6 Agreement and n-ary Composition

The problem of feature agreement by distribution for the case of n-ary composition is that it may be that a feature, appearing on the head, needs to be checked for some but not all complements. For example, consider a simplified category for a verb, $\div\{Actor, \boxed{\mathrm{s}}, Patient\}$ to which we want to add the (verbal) feature $\Box^{\downarrow}_{s}$ of "singular". Thus, distribution as in the binary case would, when generalized to the n-ary case, yield a too strong requirement for agreement since *all* complements would have to share the (applicable) features as noted for the construction. In this section, I want to consider several possibilities for formulating a "weaker" version of agreement, which is more realistic for the n-ary case.

Clearly, when the verb is singular then so should be the Actor; whereas, at the same time, the Patient can be singular or plural irrespective of the verb's number. Given the examples above, we could perhaps opt for trying either of the following two possibilities:

(3.12) $\Box^{\downarrow}_{s}\{\Box^{\downarrow}_{s}(Actor)^{\diamond_s}, \boxed{\mathrm{s}}, \Box^{\downarrow}_{n}(Patient)^{\diamond_n}\}^{\times}$

(3.13) $\Box^{\downarrow}_{s}\{\Box^{\downarrow}_{s}(Actor)^{\diamond_s}, \boxed{\mathrm{s}}, Patient\}^{\times}$

That is, in (3.12) we specify for the Patient that it can be of either number (n as underspecified feature) whereas in (3.13) we only specify for the Actor that it should be singular - the Patient is not constrained. Observe that in both cases we use the setup of (3.10). Now the question is whether we can make either of them work.

First, let us leave the Patient out of the equation and see how agreement between the verb and the Actor can be brought about. Employing essentially the n-ary analogon of the last step of (3.10), we get

$$(3.14)\quad \cfrac{\cfrac{\cfrac{\ldots}{\{\Box^{\downarrow}_{s}(Actor)^{\diamond_s}, \Box^{\downarrow}_{s}\{\Box^{\downarrow}_{s}Actor, \boxed{\mathrm{s}}, Patient\}^{\diamond_s}, Patient\}^{\times} \Rightarrow s}}{\{\Box^{\downarrow}_{s}Actor, \Box^{\downarrow}_{s}\{\Box^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}, Patient\}^{\diamond_s}\times \Rightarrow s}}{\{\Box^{\downarrow}_{s}Actor, \Box^{\downarrow}_{s}\{\Box^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}, Patient\}_{\times} \Rightarrow \Box^{\downarrow}_{s}s}$$

[9]Supposing that we are dealing with one and the same feature, of course.

That is, we make use of (3.11) to relate the $\Box_s^{\downarrow}Actor$ with the argument $\Box_s^{\downarrow}Actor$ of the n-ary residual, and use (3.8) to distribute the feature over the entire construction, so that we end up with the resulting category (the head) being marked with that feature as well. This is what happens in (3.10) as well, and if it were only for this, it would indeed seem reasonable to apply an n-ary version of the symmetric distribution law instead.

However, let us have a look now at what would happen in case we take proper care of the Patient as well. As noted, we could either mark the Patient-argument in the verbal frame with an underspecified feature $\Box_n^{\downarrow}$ which we can specialize later on, or we can leave the Patient-argument unmarked (indicating the absence of a constraint). Thus we either try to prove

$$(3.15)\quad \{\Box_s^{\downarrow}Actor, \Box_s^{\downarrow}\{\Box^{\downarrow}Actor, \boxed{\mathrm{s}}, \Box_n^{\downarrow}Patient\}, \Box_p^{\downarrow}Patient\}_{\times} \stackrel{?}{\Rightarrow} \Box_s^{\downarrow}s$$

or

$$(3.16)\quad \{\Box_s^{\downarrow}Actor, \Box_s^{\downarrow}\{\Box^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}, \Box_p^{\downarrow}Patient\}_{\times} \stackrel{?}{\Rightarrow} \Box_s^{\downarrow}s$$

whereby $\Box_p^{\downarrow}$ is the feature for plural, and $\Box_n^{\downarrow}$ (as said) the underspecified feature for number (specializable to either singular or plural).

Consider the following chain of reasoning involving (3.15) first.

$$(3.17)\quad \cfrac{\cfrac{\cfrac{\cfrac{\cdots}{\{\Box_s^{\downarrow}(A)^{\diamond_s}, \Box_s^{\downarrow}\{\Box_s^{\downarrow}A, \boxed{\mathrm{s}}, \Box_n^{\downarrow}P\}^{\diamond_s}, \Box_p^{\downarrow}(P)^{\diamond_p}\}_{\times} \Rightarrow s}\,incl}{\{\Box_s^{\downarrow}(A)^{\diamond_s}, \Box_s^{\downarrow}\{\Box_s^{\downarrow}A, \boxed{\mathrm{s}}, \Box_p^{\downarrow}P\}^{\diamond_s}, \Box_p^{\downarrow}(P)^{\diamond_p}\}_{\times} \Rightarrow s}\,?forget?}{\{\Box_s^{\downarrow}(A)^{\diamond_s}, \Box_s^{\downarrow}\{\Box_s^{\downarrow}A, \boxed{\mathrm{s}}, \Box_p^{\downarrow}P\}^{\diamond_s}, \Box_p^{\downarrow}P\}_{\times} \Rightarrow s}}{\cfrac{\{\Box_s^{\downarrow}A, \Box_s^{\downarrow}\{\Box_s^{\downarrow}A, \boxed{\mathrm{s}}, \Box_p^{\downarrow}P\}, \Box_p^{\downarrow}P\}_{\times}^{\diamond_s} \Rightarrow s}{\{\Box_s^{\downarrow}A, \Box_s^{\downarrow}\{\Box_s^{\downarrow}A, \boxed{\mathrm{s}}, \Box_p^{\downarrow}P\}, \Box_p^{\downarrow}P\}_{\times} \Rightarrow \Box_s^{\downarrow}s}}$$

This chain of reasoning does not constitute a proper inference - because look at the situation handled at the $?forget?$ step. Intuitively, in that step we "forget" about the diamond marking the Patient, leading us to a situation in which there is no diamond to be distributed. Which amounts to "forgetting" the feature, it being irrelevant in any check for agreement. However, compare the configuration to the one involving the Actor. Clearly they are the same. In case of the Actor we *do* distribute the feature, though. Consequently, we could end up having one and the same structure marked with both a singular and a plural feature, which smells after inconsistency and should therefore be avoided.

4.7 Proposed Solution

I would like to argue that the solution lays in adopting (3.16) as category for the verb, and taking the absence of any constraining feature in the category as a context in which to apply a rule to deal with the irrelevant feature of -in this case- the Patient. The idea behind the rule, as formulated below, is that the feature on a particular complement, is irrelevant with respect to checking a feature to appear on the head and checked against an other complement (*preferred/relevant distribution*):

$$\frac{\{\Box_i^{\downarrow}(A)^{\diamond_i}, \Box_i^{\downarrow} \div \{..\Box_i^{\downarrow}A.., \boxed{\mathrm{h}}, ..B..\}^{\diamond_i}, \Box_j^{\downarrow}(B)^{\diamond_j}\}_{\times} \Rightarrow h}{\{\Box_i^{\downarrow}(A), \Box_i^{\downarrow} \div \{..\Box_i^{\downarrow}A.., \boxed{\mathrm{h}}, ..B..\}, \Box_j^{\downarrow}(B)\}_{\times}^{\diamond_i} \Rightarrow h}\ PDist \tag{3.18}$$

(For good order, please note that the exact placement with respect to the head of A and B is irrelevant in this rule.)

Consequently, the following inference runs without problem:

$$\frac{\dfrac{\dfrac{\cdots}{\{\Box_i^{\downarrow}(Actor)^{\diamond_i}, \Box_i^{\downarrow} \div \{\Box_i^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}^{\diamond_i}, \Box_j^{\downarrow}(Patient)^{\diamond_j}\}_{\times} \Rightarrow s}}{\{\Box_i^{\downarrow}(Actor), \Box_i^{\downarrow} \div \{\Box_i^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}, \Box_j^{\downarrow}(Patient)\}_{\times}^{\diamond_i} \Rightarrow s}\ PDist}{\{\Box_i^{\downarrow}(Actor), \Box_i^{\downarrow} \div \{\Box_i^{\downarrow}Actor, \boxed{\mathrm{s}}, Patient\}, \Box_j^{\downarrow}(Patient)\}_{\times} \Rightarrow \Box_i^{\downarrow}s} \tag{3.19}$$

Hence, even though the setting of (3.10) may seem undesirable for the binary setting unless one is dealing with different features, (3.10) does offer an initial idea how to solve a problem concerning agreement in the n-ary setting. Finally, we need a few rules to handle inclusion, and skipping of features during agreement (i.e. features which are irrelevant to the checking agreement at hand - see the example at the beginning of the chapter).

Definition 4.6 (Asymmetric agreement between Heads and Dependents) *In addition, the following proof rules describe the handling of features when concerns agreement between heads and dependents modelled by asymmetric distribution laws. [nInc] is inclusion for the n-ary case, and [nDepAD] and [nHeadAD] are the asymmetric distribution rules for irrelevant dependent respectively head features.*

$$\frac{(...(s_i)^{\langle f\rangle}...(t)^{\langle f\rangle}...)^{\times_i^n} \vdash B : s}{(...(s_i)...(t)...)^{\times_i^n} \vdash [f]B : s}\ nAgr\quad c$$

$$\frac{(....w^{\langle f\rangle}...)^{\times_i^n} \vdash B : s}{(....w^{\langle f'\rangle}...)^{\times_i^n} \vdash B : s}\ nInc$$

$$\frac{(...(s_i)^{\langle f\rangle}...(t)...)^{\times_i^n} \vdash B : s}{(...(s_i)...(t)...)^{\times_i^n} \vdash [f]B : s}\ nDepAD$$

$$\frac{(...(s_i)...(t)^{\langle f\rangle}...)^{\times_i^n} \vdash B : s}{(...(s_i)...(t)...)^{\times_i^n} \vdash [f]B : s}\ nHeadAD$$

■

5 Form & Function, and Grammaticality

In the discussion above, I made a distinction between *surface modes* of composition, and *deep modes* of composition. The surface modes are primarily used to compose groups of wordforms, and in some cases allow for specifications of a local surface syntax[10]. Deep modes are used to compose dependency structures.

The inferences that operate on these modes are valid because they (can be proven to) follow the intended semantics of the modes. These intended semantics are defined using frames on universes. Basically put, a universe is a 'set' of "possible worlds" (grammatical structures). A frame defines an accessibility relation over possible worlds/grammatical structures, stating under what conditions one can create a composed grammatical structure given two grammatical structures one wants to combine (i.e. reach a possible world that is accessible from the worlds at which the grammatical structures-to-be-combined reside).

When defining the modes, I took two universes, $\mathfrak{U}_{surf}$ and $\mathfrak{U}_{deep}$, and interpreted the surface modes on $\mathfrak{U}_{surf}$ and the deep modes on $\mathfrak{U}_{deep}$. The result of this split is that we can now only define grammaticality of surface structures separately from the grammaticality of deep structures. An overall judgement of grammaticality, analyzing a surface structure *in terms of* an underlying deep structure, is unattainable since there is no relation between the two universes.

Employing the tools of multidimensional modal logic (MDML), developed by Venema and Marx in [28], such a relation *can* be fully specified. MDML is a formalism of modal logic in which a frame can be specified over a cartesian product of universes: It is multidimensional in allowing for semantics to be defined over more than one universe. As a consequence, validity of a proposition ψ is no longer defined at a point x, but at a *state* $(x_1, ..., x_n)$ with x_1 in the first universe of the product, and x_n in the last universe of the product.

The idea then is to take the cartesian product of $\mathfrak{U}_{surf}$ and $\mathfrak{U}_{deep}$, $\mathfrak{G} = \mathfrak{U}_{surf} \times \mathfrak{U}_{deep}$, and define a frame $\mathfrak{F} = \langle \mathfrak{G}, I \rangle$, with I is the interpretation function. Let us take $\mathfrak{M}$ as a model based on $\mathfrak{F}$, and $\mathfrak{M}, (sf, df) \models \phi$ stand for the validity of the statement that "the surface structure residing at sf can linguistically be interpreted as the deep structure (function) at df".

Subsequently, to formalize the relation between form and function, I define in I two operators, $\otimes$ and $\oplus$. The interpretation of $\otimes$ is to take the proposition that a certain surface category is grammatical to a proposition saying that the surface category can be interpreted as a particular deep category. The interpretation of $\oplus$ is the converse - it takes a deep category to

[10] For example, a surface mode can be introduced to deal with the requirement of an expletive pronouns as a grammatical subject to appear with Actor-less verbs.

a surface form. More linguistically put, $\otimes$ defines functional interpretation, and $\oplus$ functional realization.

Definition 5.1 (Form and Function) *Given $\mathfrak{U}_{surf}$ and $\mathfrak{U}_{deep}$, with $\mathfrak{G} = \mathfrak{U}_{surf} \times \mathfrak{U}_{deep}$. Define a frame $\mathfrak{F} = \langle \mathfrak{G}, I \rangle$, with I the interpretation function. The definition of I as follows:*

$$I(\otimes) = \{(u,v),(x,y)|x = 0 \wedge v = y\}$$

$$I(\oplus) = \{(u,v),(x,y)|u = x \wedge y = 0\}$$

■

Notation 5.1 *With a bit of abuse of notation, I will write $SC \otimes DC$ for the (functional) interpretation of the surface category SC as the deep category DC, and $DC \oplus SC$ for the (functional) realization of the deep category DC as the surface category SC.*

Remark 5.1 An important remark concerns what $\mathfrak{M}$ will take from the abstract internal structure of the universe $\mathfrak{G}$. Clearly, if we take $\mathfrak{G}$ at face-value, a lot of predictions would be borne out that would be linguistically infelicitous, since in $\mathfrak{G}$ every state is in principle accessible. Hence, a surface form in nominative case would be interpretable as a Manner, which is clearly undesirable. Therefore, we will assume for $\mathfrak{M}$ that we have some means of pruning $\mathfrak{G}$ such that only linguistically 'valid' deductions will indeed be valid on $\mathfrak{M}$. For smaller models, we may go by actual construction. Mostly $\mathfrak{G}$ will be restricted to mapping surface categories including particular morphological features to a dependency kind (as in $\mathfrak{B}_{deep}$).

Remark 5.2 Note that if it is ensured that there is a perfect matrix μ for $\mathfrak{M}$ (in terms of [28], Chapter 2) then consistency is obtained.

Chapter 4

Term-Assignment in DBLG

Before I present a formalization of term-assignment for DBLG, which will be based on Hepple's [13] and Wansing's [52], I discuss the idea of term-assignment, and 'types' in DBLG.

1 The Idea Of Term-Assignment

The idea of term-assignment is to associate formulas in a proof (the categories) with lambda terms, following the Curry-Howard interpretation of proofs: Instead of $R \vdash C$, propositions will now look like $R \vdash C : S$, with S the term associated to category C. In particular, within a natural deduction system, the familiar lambda abstraction corresponds to introduction, whereas application corresponds to elimination. In a true Curry-Howard *isomorphism*, the terms associated with formulas, and the operations on terms (in terms of a lambda calculus) following the proof steps, in fact provide a *record* of how the (natural deduction) proof proceeds. For categorial type logics, it appears hard to establish such an *isomorphism*, due to the hybrid nature of the logic. Moortgat in [32] argues that, rather than an isomorphism, a (weaker) correspondence fits the intentions as well. It remains a topic for further research whether such a correspondence (in absence of a Church-Rosser property) is indeed desirable - cf. for example [36],p.24ff on the necessity of Church-Rosser for a theory of natural language semantics[1].

Here, I will use the idea of correspondence, perceiving of a (complex) term as showing how (less complex) terms, composed so far, can be interpreted as fitting together. What is more, since individual words are assigned categories and can therefore be assigned corresponding terms as well, a complex term can in fact be seen as a *(partial) linguistic object* (cf. [13]) representing how words can be interpreted to fit together (in a grammatical way).

In case the proof concludes in a category s, the term that has been built up is a complete, interpreted representation of the sentence's grammatical

[1]Failure of Church-Rosser means that a term may have more than one β-reductions. As Muskens points out, for a theory of semantics this is somewhat undesirable, because, since each β-reduction spells out a different semantics/meaning, one and the same term can be assigned different meanings, while retaining *one* corresponding syntactic analysis.

structure. Let me rephrase this in terms of how I propose to see this in DBLG. Whereas the inference over the formulas/categories attempts to show that a sentence is grammatical on the basis of a dependency-based syntax, *the complex term that is built up provides the actual dependency structure.*

2 Types in DBLG

In categorial grammar (or, *type-logical grammar*) one usually takes a typed lambda calculus, with Montague-style typing. This is not what I intend to do in DBLG.

There are, essentially, two reasons. For one, assigning truth-semantic types directly to a dependency structure does not exactly square with the idea of linguistic meaning, as outlined in [45]. Therefore, I will simply use kinds of dependency relations as basic types, being the semantic analogues of the kinds found in the (deep) categories.

Secondly, there appears to be a more fundamental issue involved in relating a dependency structure with a Montagovian typing system. Montague based his typing system on the lambda calculus as developed by Church, requiring that each category has one and only one corresponding type. A more flexible approach has been developed by Hendriks in [11], where a category may be assigned several different types.

The relational nature of interpretation in a dependency-based framework, however, seems to call rather for a constructive approach to typing, in which a type is *created* for a formula depending on the context in which the formula appears. Put slightly different, a formula is assigned a type that is *relevant* within context. Curry developed such an approach to typing (cf. [49] for an excellent survey of both Church- and Curry-style lambda calculi).

For the moment I will just have to leave the reader with pondering over this idea, whose main point is "simply" that instead of enumerating the possible types for a category, a type is inferred that would enable one to interpret the category in context. Although I would not want to go as far as to venture any claim, (given the absence of any formalization), such typing might prove to be a step up from flexibility: Like going from an extensionally defined set of possible types to an 'intensional' way of saying what it means for a category to have a particular type[2].

[2]For the historically minded reader, I would like to refer to the discussions between Lambek and Curry in the early sixties, as cited in [32]. For the linguistically minded reader it would perhaps be interesting to observe that the kind of linguistic sign that would underly such interpretation is inherently *not* like the binary Saussurian sign, but like the triadic Jakobsonian-Peircean linguistic sign - cf various articles [20].

3 Formalization

Following Hepple's [13] and Wansing's [52] (particularly Chapter 5), term-assignment for DBLG is defined employing a directional variant of the original (implicational) lambda-calculus. More precisely, higher types can be defined using directional slashes (corresponding to the residuals of the various modes), expressing the direction where the argument(s) for functional application should be found.

Notation 3.1 *For the binary modes, the traditional slashes* $\{\backslash_i, /_i\}$ *can be used. To give n-ary residuation a directional flavor (following the definitions as in definition 4.4), I will use* $\backslash^{\div_i^n}$ *and* $/^{\div_i^n}$ *to indicate the left- and right-arguments in the residual* $\div_i^n$. *The arguments themselves will be enclosed in curly brackets* $\{\}$. *In the definition below,* M^A *stands for a term* M *of type* A, *and* x^B *for a variable of type* B.

Definition 3.1 (Typed Lambda Calculus) *The lambda calculus* $\lambda_{\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}}$ *is a directional variant of the ordinary typed lambda calculus* $\lambda_{\supset}$.

The vocabulary of the term language $T_{\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}}$ *consists of denumerably many variables* $v_1, v_2, ...,$ *every formula in* $\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}$, *the lambda abstractors* λ^l *and* λ^r, *and brackets* $(,)$.

The set $\Lambda_{\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}}$ *of* $T_{\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}}$*-terms is the smallest set* Γ *such that*

(i) $V_{\{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}} \equiv \{v_i^A | 0 < i \in \omega, A \text{ a formula in } \{\backslash_i, /_i, \backslash^{\div_i^n}, /^{\div_i^n}\}\} \subset \Gamma$

(ii) if $N^{B/_iA}, M^A \in \Gamma$, *then* $(NM)^B \in \Gamma$.

(iii) if $M^A, N^{A\backslash_iB} \in \Gamma$, *then* $(MN)^B \in \Gamma$

(iv) if $M_1^{A_1}, ..., M_{m-1}^{A_{m-1}}, N^{\{A_1,...,A_{m-1}\}\backslash^{\div_i^n} B /^{\div_i^n} \{A_{m+1},...,A_n\}}, M_{m+1}^{A_{m+1}}, ..., M_n^{A_n} \in \Gamma$, *then* $(M_1...M_{m-1}NM_{m+1}...M_n)^B \in \Gamma$

(v) if $M^B \in \Gamma, x^A \in V_{\{\backslash_i, /_i\}}$, *then* $(\lambda_i^r x.M)^{(B/_iA)}, (\lambda_i^l x.M)^{A\backslash_iB)} \in \Gamma$

(vi) if $N^B \in \Gamma, x^{A_1}...x^{A_{m-1}}x^{A_{m+1}}...x^{A_n} \in V_{\{\backslash^{\div_i^n}, /^{\div_i^n}\}}$, *then* $(\lambda_i^l x^{A_1}...x^{A_{m-1}}\lambda_i^r x^{A_{m+1}}...x^{A_n}N)^{(\{A_1,...,A_{m-1}\}\backslash^{\div_i^n} B /^{\div_i^n} \{A_{m+1},...,A_n\})} \in \Gamma$

■

Notation 3.2 *Note that* $\lambda^l x_1...x_k.M$ *is per definition equal to* $(\lambda^l x_1(\lambda^l x_2(....(\lambda^l x_k.M)...),$ *similarly for* λ^r *as well as a combination of a sequence of* λ^l*'s with a sequence of* λ^r*'s (as in (vi) above).*

Hereafter, $N[M^B/x^B]$ *stands for the substitution of* M^B *for every variable* x^B *in* N *as bound by a lambda-abstraction.*

Definition 3.2 (Typed β-equivalence and β-reduction, [52]) *The axiom schemas for typed β-equality are*[3]*:*

(β^r) $(\lambda^r x^A.M)N^A = M[N/x]$

(β^l) $N^A(\lambda^l x^A.M) = M[N/x]$

The binary relations $\stackrel{\beta}{\rightarrow}$ *(one-step β-reduction) and* $\stackrel{\beta}{\twoheadrightarrow}$ *(β-reduction) and* $\stackrel{\beta}{=}$ *(β-convertibility) are defined as:*

1. $(\lambda^r x^A.M)N^A \stackrel{\beta}{\rightarrow} M[N/x]$,
   $N^A(\lambda^l x^A.M) \stackrel{\beta}{\rightarrow} M[N/x]$;
2. $(\lambda_i^l x^{A_1}...x^{A_{m-1}}\lambda_i^r x^{A_{m+1}}...x^{A_n} M)^{(\{A_1,...,A_{m-1}\}\backslash^{\div_i^n} B/^{\div_i^n}\{A_{m+1},...,A_n\})}$
   with to the left $N_1^{A_1}...N_{m-1}^{A_{m-1}}$ *and to the right* $N_{m+1}^{A_{m+1}}...N_n^{A_n}$
   $\stackrel{\beta}{\rightarrow} M[N_1/x_1,...,N_n/x_n]$
3. *if* $M^A \stackrel{\beta}{\rightarrow} N^A$*, then* $MG^{(A\backslash B)} \stackrel{\beta}{\rightarrow} NG^{(A\backslash B)}$, $G^{(B/A)}M \stackrel{\beta}{\rightarrow} G^{(B/A)}N$;
   if $M^{(A\backslash B)} \stackrel{\beta}{\rightarrow} N^{(A\backslash B)}$*, then* $G^A M \stackrel{\beta}{\rightarrow} GN$;
   if $M^{(B/A)} \stackrel{\beta}{\rightarrow} N^{(B/A)}$*, then* $MG^A \stackrel{\beta}{\rightarrow} NG$;
   if $M \stackrel{\beta}{\rightarrow} N$*, then* $\lambda_i^r x.M \stackrel{\beta}{\rightarrow} \lambda_i^r x.N$, $\lambda_i^l x.M \stackrel{\beta}{\rightarrow} \lambda_i^l x.N$
4. *Analogously for the n-ary case.*
5. $\stackrel{\beta}{\twoheadrightarrow}$ *is the reflexive and transitive closure of* $\stackrel{\beta}{\rightarrow}$*.*
6. $\stackrel{\beta}{=}$ *is the equivalence relation generated by* $\stackrel{\beta}{\twoheadrightarrow}$

■

Definition 3.3 (β-Redex, β-Normal Form) *The terms* $(\lambda^r x^A.M)N^A = M[N/x]$ *and* $N^A(\lambda^l x^A.M) = M[N/x]$ *are called β-*redexes*, which both have as their* contractum *the term $M[N/x]$. M is a β-*normal form *if it has no β-redex as a subterm. M has a β-normal form if there exists an N such that* $M \stackrel{\beta}{=} N$ *and N is a β-normal form.* ■

Remark 3.1 Wansing proves in [52] (Chapter 5) that the Church-Rosser theorem holds for the directional lambda calculus $\lambda_{\{\backslash,/\}}$ - that is, it can be proven that each M has exactly one (i.e. a unique) β-normal form. It appears that, for the lambda calculus $\lambda_{\{\backslash_i,/_i\backslash^{\div_i^n},/^{\div_i^n}\}}$ this result can be extended, so that also in case of DBLG the terms have unique β-normal forms for individual logics (i.e. the hybrid case still remains open).

[3]Observe that, by the notation above, the axiom schemas by definition extend to the n-ary case.

Finally, the proof syntax should be adapted so as to include term-assignment. Above I already alluded to the different format of the propositions one finds in the proofs: Instead of $R \vdash C$ now a *labelled deductive system*-format is used, $R \vdash C : T$, with T the term assigned to the category C (that is implied by the resources R). The definition below gives the adapted rules for both binary and n-ary modes, as well as the adapted versions of $[OptDel]$ and $[FM \div E]$

Definition 3.4 (Labelled Proof Syntax) *In addition to the resources and the category, a proposition in a proof is labelled with a term from the term-language as defined in definition 3.1. Operations on terms, in parallel with manipulations of the formulas, follows the λ-calculus as defined in definition 3.1.*

Labelled Proof Syntax for Binary Modes

$$(1)\quad \frac{s \vdash C_j/_h C_k : M^A \quad t \vdash C_k : N^B}{(s \circ_h t) \vdash C_k : (M^A\ N^B)}\ /_h E \qquad \frac{[v \vdash C_j : M^B] \quad (s \circ_h v) \vdash C_i : N^A}{s \vdash C_i/_h C_j : \lambda^r_h v^B.N^A}\ /_h I$$

$$(2)\quad \frac{t \vdash C_j : N^B \quad s \vdash C_j \backslash_h C_i : M^A}{(t \circ_h s) \vdash C_i : (N^B\ M^A)}\ \backslash_h E \qquad \frac{[v \vdash C_j : M^B] \quad (v \circ_h s) \vdash C_i : N^A}{s \vdash B \backslash_h C_i : \lambda^l_h v^B.N^A}\ \backslash_h I$$

$$(3)\quad \frac{[v \vdash C_i : v], [w \vdash C_j : w] \quad t \vdash C_i \circ_h C_j : N^B \quad s[(v \ \circ_h \ w)] \vdash C_k : M^A}{s[t] \vdash C_k : [N^B/(v \ \circ \ w)].M^A}\ \circ_h E$$

$$\frac{s \vdash C_i : M^A \quad t \vdash C_j : N^B}{(s \ \circ_h \ t) \vdash C_i \circ_h C_j : \langle M^A, N^B \rangle}\ \circ_h I$$

$$(4)\quad \frac{s[((x \circ_i y) \circ_i z)] \vdash C_i : M^A}{s[(x \circ_i (y \circ_i z))] \vdash C_i : M^A}\ a$$

$$(5)\quad \frac{s[(x \circ_i y)] \vdash C_k : M^A}{s[(x \circ_j y)] \vdash C_k : M^A}\ i/j$$

$$(6)\quad \frac{s[(x \circ_i y)] \vdash C_k : M^A}{s[(y \circ_i x)] \vdash C_k : M^A}\ p$$

Labelled Proof Syntax for n-ary Modes[4]

$$(7)\quad \frac{\begin{array}{c} s_1 \vdash C_1 : M_1^{A_1} \cdots s_{m-1} \vdash C_{m-1} : M_{m-1}^{A_{m-1}} \\ t \vdash \div_i^n \{C_1, \ldots, C_{m-1}, \boxed{RC}, C_{m+1}, \ldots, C_n\} : (\lambda_i^l x^{A_1} \ldots x^{A_{m-1}} \lambda_i^r x^{A_{m+1}} \ldots x^{A_n} N)^{(\{A_1, \ldots, A_{m-1}\} \backslash^{\div_i^n} B /^{\div_i^n} \{A_{m+1}, \ldots, A_n\})} \\ s_{m+1} \vdash C_{m+1} : M_{m+1}^{A_{m+1}} \cdots s_n \vdash C_n : M_n^{A_n} \end{array}}{(s_1, \ldots, s_{m-1}, t, s_{m+1}, \ldots, s_n)^{\times_i^n} \vdash RC : (M_1^{A_1} \ldots M_{m-1}^{A_{M-1}} N^B M_{m+1}^{A_{m+1}} M_n^{A_n})}\ \div_i E$$

[4]My apologies for the tiny script - but otherwise, things would not have fit onto the page.

$$\text{(8)}\quad \frac{\begin{array}{c}[s_1 \vdash C_1 : M_1^{A_1} \cdots s_{m-1} \vdash C_{m-1} : M_{m-1}^{A_{m-1}}] \\ (s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_i^n} \vdash RC : (M_1^{A_1} ... M_{m-1}^{A_{M-1}} N^B M_{m+1}^{A_{m+1}} M_n^{A_n}) \\ [s_{m+1} \vdash C_{m+1} : M_{m+1}^{A_{m+1}} \cdots s_n \vdash C_n : M_n^{A_n}]\end{array}}{t \vdash \div_i^n \{C_1, ..., C_{m-1}, \boxed{RC}, C_{m+1}, ..., C_n\} : (\lambda_i^l x^{A_1} ... x^{A_{m-1}} \lambda_i^r x^{A_{m+1}} ... x^{A_n} N)^{(\{A_1, ..., A_{m-1}\} \backslash^{\div_i^n} B /^{\div_i^n} \{A_{m+1}, ..., A_n\})}} \div_i I$$

$$\text{(9)}\quad \frac{\begin{array}{l}[s_1 \vdash C_1 : M_1^{A_1}] \cdots [s_n \vdash C_n : M_n^{A_n}] \\ s[(s_1, ..., s_n)^{\times_i^n}] \vdash C_k : L^C \qquad t \vdash \times_i^n (C_1, .., C_n) : N^B\end{array}}{s[t] \vdash C_k : L^C [N^B / (M_1^{A_1} ... M_n^{A_n}}\ \times_i E$$

$$\text{(10)}\quad \frac{s_1 \vdash C_1 : M_1^{A_1} \cdots s_n \vdash C_n : M_n^{A_n}}{(s_1, , ..., s_n)^{\times_i^n} \vdash \times_i^n (C_1, ..., C_n) : \langle M_1^{A_1}, ..., M_n^{A_n} \rangle}\ \times_i I$$

Labelled Proof Syntax for $[OptDel]$***,*** $[FM \div E]$

$$\frac{\begin{array}{c}\mathcal{A}_{left} \quad \cdots \quad \mathcal{A}_{right} \\ \vdots \\ (Lft, t, Rght)^{\times_i^n} \vdash C_k\end{array}}{(Lft - \mathcal{A}_{left}^{\vee}, t, Rght - \mathcal{A}_{right}^{\vee})^{\times_i^{n-k}} \vdash C_k}\ OptDel$$

The idea for $[OptDel]$ *is to change the term corresponding to* $(Lft, t, Rght)^{\times_i^n} \vdash C_k$*, being* N^B*, by "deleting" the argument places for the optional unused dependencies. That is, every assumption in* $\mathcal{A}_{left}$ *and* $\mathcal{A}_{right}$ *which initially got filled in for a* λ*-bound variable in* N^B*, will be replaced by an* $\emptyset$*. Subsequently, we could define an analogue of* η*-reduction that gets rid of* $\emptyset$*'s.*

$$\frac{\begin{array}{l}s_1 \vdash C_1 \cdots s_{m-1} \vdash C_{m-1} \quad t \vdash \div_i^n \{C_1, ..., C_{m-1}, \boxed{B}, C_{m+1}, ..., C_n\} \\ s_{m+1} \vdash C_{m+1} \cdots s_n \vdash C_n\end{array}}{(s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_i^{n+k}} \vdash B}\ FM - \div E$$

The idea for $[FM - \div E]$ *is to simply copy an argument in the term of the head category, the copy being of the same type as the (initial) free modifier of course, and bind the copy by the term corresponding to the additional free modifier.* ■

4 Terms As Dependency Structures

As I already mentioned before (cf. the introduction here, and the example in the previous chapter), the term built up during the derivation will be understood as the dependency structure. In this section I will make this more concrete.

To start with, the basic types to be considered for the λ-calculus (and its term language) are *dependencies*. Simply put, if a wordform is interpreted as a dependent of a particular (categorial) kind, it will also get a type assigned to it reflecting that kind. For example, if we have a noun in nominative case, functionally interpreted as an Actor (which is a deep category), then we assign it a term that is of type Actor:[5]

$$[nom][fem][sg]kobliha \vdash n \otimes Actor : kobliha'^{Actor}$$

To make the reading more perspicuous, I will write $Actor : kobliha'$ instead of $kobliha'^{Actor}$ to indicate that $kobliha'$ as the semantics of the wordform "kobliha" is of type $Actor$.

The same can be done for variables: $Object : x_i$ means that variable x_i is of type $Object$. Consequently, a type corresponding to a valency frame will look like the following:

$$(\lambda_i^l x^{A_1} ... x^{A_{m-1}} \lambda_i^r x^{A_{m+1}} ... x^{A_n} .N)^{(\{A_1,...,A_{m-1}\} \backslash^{\div_i^n} B /^{\div_i^n} \{A_{m+1},...,A_n\})}$$
$$\leadsto \text{(in the revised formulation)}$$
$$\lambda_i^l x^{A_1} ... x^{A_{m-1}} \lambda_i^r x^{A_{m+1}} ... x^{A_n} .$$
$$(\{(A_1 : x_1)...(A_{m-1} : x_{m-1})\} \backslash_i^n \quad N^B \quad /_i^n \{(A_{m+1} : x_{m+1})...(A_n : x_n)\})$$

For example, if we take the verb *to read* to take an Actor dependent and an Object dependent, its type could be

$$\lambda^l x_a \lambda^r x_o .(\{(Actor : x_a)\} \backslash_i^n \quad read \quad /_i^n \{(Object : x_o)\})$$

Its arguments get filled in following the elimination of slashes in the derivation over the categories. Thus, if we find a wordform ("Albert") to the left of the verb which we can interpret as an Actor, and a wordform to the right interpretable as an Object ("a book"), then the resulting term would be

$$(\lambda^l x_a \lambda^r x_o .(\{(Actor : x_a)\} \backslash_i^n \quad read \quad /_i^n \{(Object : x_o)\}))(Actor : Albert')^l (Object : book')^r$$
$$\beta\text{-reduces to}$$
$$(\{(Actor : Albert)\} \backslash_i^n \quad read \quad /_i^n \{(Object : book)\})$$

Remark 4.1 For convenience I used superscripts l and r to $(Actor : Albert')$ and $(Object : book')$ in order to indicate that the former occurred to the left, and the latter to the right (cf. definition 3.2, point 2.) Note that if the slashes are omitted, the above representation of a dependency structure is easily transformable into a representation following the syntax used by Petkevič in [42].

[5] Although the first mentioned 'Actor' is formally a category, whereas the second mentioned 'Actor' is formally a type, I will use the same names throughout.

Remark 4.2 The above example does not show how to deal with diversions from canonical ordering, and how that would show up in the dependency structure. In the next chapter I will introduce commutativity for n-ary deep composition, and define term-assignment for that structural rule such that non-canonical positioning of dependents will also be reflected within the term itself.

Chapter 5

Extending DBLG

1 Introduction

Thus far, I have designed DBLG such that wordforms can only assigned categories that are either strictly concerned with surface composition, or purely with deep composition. There are, however, numerous examples that call for *mixed categories* - that is, categories in which both surface and deep composition reside side by side, possibly related by $\otimes$. In section 2 I consider various examples of mixed categories, and show how DBLG can be extended so as to incorporate the idea of mixed categories.

Other extensions of DBLG discussed in this chapter concern word order in the deep dimension. In section 3 I first present a discussion of adding commutativity to the deep mode proof syntax, and its role in determining structural indications of informativity. The kind of commutativity used there essentially preserves projectivity, for it only allows dependents together with their entire subtree (if any) to occur in a non-canonical position. The second part of section 3 is devoted to a discussion how nonprojectivity could be dealt with in DBLG.

2 Categories Combining Surface and Deep Modes

Let us consider three different kinds of examples.

(i) Prepositions or postpositions bring about a specific interpretation of the wordform group they combine with. For example, in Japanese the postposition "o" leads to the interpretation of the preceding wordform group as a Patient, whereas in Czech the preposition "v" when combined with a nominal group in accusative case brings about an interpretation as Effect, and in combination with a nominal group in locative case an interpretation as Location:

(5.1) v + locative: Bydlí me v Praze.
(*En.* We live in Prague.)

(5.2) v + accusative: Neptun proměnil dívku v mořskou vílu.
(*En* Neptune changed the girl into a mermaid.)

(ii) There are particular verbs that do not take an Actor, but do require to be realized at the surface with a grammatical subject - take for example the verb "to rain" in a number of Germanic languages. As such, the verb's category should not only specify with what dependency relations it would (or could) combine, but also that it needs an expletive pronoun to go with it as grammatical subject (at the surface).

(iii) In German, for example, various transitive verbs require their Patient to be in a particular case (either dative or accusative). The verb "hilfen" (*En.*to help) requires a Patient to be in dative case, whereas the verb "finden" (*En.*to find) requires a Patient to be in accusative case.

Points (i) and (iii) are relatively easy to deal with. Let me begin with (i).

Example 2.1 (Point (i) - "o") Formally, we should start by changing the definition 3.1 on page 27, for a category of the form $A \otimes B$ should be considered a proper surface category. Since B is to be a dependency, we first of all need a reference to the basic set of dependencies $\mathfrak{B}_{deep}$ considered for the deep categories: $B \in \mathfrak{B}_{deep}$ (cf. definition 4.1 on page 33). Secondly, I will restrict A to contain no subcategories involving $\otimes$.

Definition 2.1 (Extension to Surface Categories) *A category C is a surface category if either it is a category according to definition 3.1 on page 27, or it is of the form $A \otimes B$, with $B \in \mathfrak{B}_{deep}$ (cf. definition 4.1 on page 33) and A not containing a subcategory involving $\otimes$.* ■

Subsequently, we are able to construct a category for a postposition like "o" as follows:

$$n \backslash_s (n \otimes Patient)$$

that is, the postposition needs a noun (or nominal head) to its left, to result in a surface structure that will get as category a noun (or nominal head) to be interpreted as a Patient. The $n \otimes Patient$ will then be handled in exactly the same way as usual. The only difference in the derivation is that this time, no introduction step for $\otimes$ is needed to arrive at a functional interpretation of the noun, since it is enforced by the postposition.

Example 2.2 (Point (i)-"v") Because the surface category can include features as well, a category for the preposition "v" which needs, for example, a noun in accusative case to its right, leading to that noun -or nominal group- being interpreted as an $Effect$, could look as follows:

$$((n \otimes Location)/_s[acc]n)$$

Remark 2.1 The question is whether the category $((n \otimes Location)/_s[acc]n)$ for the preposition "v" does not only sound good *intuitively*, but also *formally*. The case we want to obtain is that the features of the nominal head are retained after checking agreement. Formally, following the way agreement and distribution have been defined for the binary case, that would mean however that the feature should appear in front, and not embedded:

$$[acc]((n \otimes Location)/_s n)$$

Since the remaining features of the noun or irrelevant to "v", we could either opt for adding underspecified features to the category, or add an asymmetric distribution rule for binary modes. For reasons of economy in lexical specification, I will opt for the latter. The formulation is left to the reader - see the definition for the n-ary case in section 4.4 in the Chapter 3.

Example 2.3 (Point (iii)) When the relation between form and function was formalized using a multi-dimensional modal framework (cf. the discussion in section 5, Chapter 3, particularly definition 5.1 on page 44), two operators were introduced: $\otimes$ to deal with functional interpretation, and $\oplus$ for functional realization. In the discussion so far, $\otimes$ figured prominently. Here, I would like to draw some attention to $\oplus$, proposing it as a solution for the case under point (iii).

On page 44 I already introduced the notation $DC \oplus SC$ as standing for the functional realization of a dependent DC as a surface category SC. Let me first of all extend the definition of deep category so as to cover all the constructions discussed so far.

Definition 2.2 (Deep Categories) *The set of* deep categories $\mathfrak{U}_{deep}$ *is defined over a finite, non-empty set of basic categories* $\mathfrak{B}_{deep}$ *as follows: (1) All the basic categories from* $\mathfrak{B}_d eep$ *are categories. (2) If* A *and* B *are categories, and* i *is a deep modes of composition, then* $A/_iB$ *and* $B\backslash_i A$ *are categories. (3) If* A *is a category, and* f *a feature, then* $[f]A$ *is a category. (4) If* A *is a deep category composed exclusively by categories as in (1) and (2), and* B *is a surface category of the form* $[f_1]...[f_k]w$ $(w \in \mathfrak{B}_{surf})$*, then* $A \oplus B$ *is a category. (5) If* A *is a category formed by (1)-(4), then* $A^{\boxminus\circledast}, A^{\boxminus\circledcirc}, A^{\boxplus\circledast}$ *and* $A^{\boxplus\circledcirc}$ *are also categories. (6) Nothing else is a deep category.* ■

Subsequently, if we want to say that for a noun to function as a Patient it should be in a specific case, then we can use $\oplus$ and specify the case like we did it above. Thus, for a verb like *finden*, we could get a category (for the third person singular, present tense "findet"):

$$findet \vdash [3rd][sing][pres] \div_d^n \{Actor, \boxed{\mathrm{s}}, (Patient \oplus [acc]n)\}$$

Example 2.4 (Point (ii)) In order to deal with the problem of specifying, for a particular verb, that it needs an expletive pronoun to go with it, we could first of all introduce a basic surface category ep (for expletive pronoun), and then introduce a surface mode of composition e to formulate the requirement in. Thus, if we take the verb "to rain" and assume -for the sake of simplicity- that it does not take any dependents, then its category would simply be $ep \backslash_e s$.

Thus,

$$\frac{it \vdash ep \quad rains \vdash ep \backslash_e s : rain'}{it \circ_e rains \vdash s : rain'} \backslash_e E$$

would be the straightforward analysis for the sentence "it rains". Because the expletive pronoun is a function word, it does not have any corresponding semantic term. Consequently, the dependency structure contains a single node for the verbal head, $rain'$.

However, what if we want the category for "rains" to include a $Time$ and $Location$, so that sentences like "In Prague it rains today" and "It rains in Prague today" could also be analysed? The issue is not so much how to specify the deep category including only the dependents (omitting features):

$$\div_d^3\{\boxed{\text{s}}, Location, Time\}$$

Rather, where should the "$ep \backslash_e ...$" be put? The proper placement is as follows:

$$ep \backslash_e (\div_d^3\{\boxed{\text{s}}, Location, Time\})$$

since this will enable us to analyse both of the sentences above. Assume for the moment that there d has access to (projective) commutativity $[pc]$ (see also the next sections). Then,

$$\frac{\dfrac{it \vdash ep \quad rains \vdash ep \backslash_e (\div_d^3\{\boxed{\text{s}}, Location, Time\})}{(it \circ_e rains \vdash \div_d^3\{\boxed{\text{s}}, Location, Time\}} \backslash E \quad \begin{matrix} \vdots \\ in\ Prague \vdash n \otimes Location \end{matrix} \quad \begin{matrix} \vdots \\ today \vdash n \otimes Time \end{matrix}}{((it \circ_e rains), (in\ Prague), today)^{\times_d^3} \vdash s} \div E$$

and

$$\frac{\dfrac{it \vdash ep \quad rains \vdash ep \backslash_e (\div_d^3\{\boxed{\text{s}}, Location, Time\})}{(it \circ_e rains \vdash \div_d^3\{\boxed{\text{s}}, Location, Time\}} \backslash E \quad \begin{matrix} \vdots \\ in\ Prague \vdash n \otimes Location \end{matrix} \quad \begin{matrix} \vdots \\ today \vdash n \otimes Time \end{matrix}}{\dfrac{((it \circ_e rains), (in\ Prague), today)^{\times_d^3} \vdash s}{((In\ Prague), (it \circ_e rains), today)^{\times_d^3} \vdash s} pc} \div E$$

Had we put the "$ep \backslash_e$ ..." directly to the $\boxed{s}$, like in

$$\div^3_d\{\boxed{ep \backslash_e s}, Location, Time\}$$

then we would have run into trouble, since from

$$\dfrac{it \vdash ep \qquad \dfrac{rains \vdash \div^3_d\{\boxed{ep \backslash_e s}, Location, Time\} \quad \overset{\vdots}{(in\ Prague) \vdash n \otimes Location} \quad \overset{\vdots}{today \vdash n \otimes Time}}{(rains, (in\ Prague), today)^{\times^3_d} \vdash ep \backslash_e s}\div E}{it \circ_e (rains, (in\ Prague), today)^{\times^3_d} \vdash s}\backslash E$$

we would not be able to obtain "In Prague it rains today" since projective commutativity only allows permutation within the structure composed by $\times$. Using the other categorial assignment to "rains", $(it \circ_e rains)$ ends up within the structure, so that projective commutativity can be properly applied.

3 Extending the Behavior of Deep Modes

In the previous chapters I restricted the behavior of deep modes to a base logic without any further structural rules defining associativity or (various kinds of) commutativity. Here I will add one kinds of commutativity to the deep dimension: *projective commutativity.*

In 4 I will discuss a basic setup for incorporating *structural indications of informativity* into DBLG. In FGD, the notions of 'contextual boundness' and 'contextual nonboundness' have been proposed as structural notions, on which a sentence's *topic/focus-articulation* can be based [45]. That is, nodes in a dependency structure can be labelled as either contextually bound or nonbound, and -for projective dependency trees- a recursive definition is given in [45] that groups the contextually bound nodes into the sentence's topic, and the nonbound nodes into the focus. The "null-hypothesis" advanced below is that we can infer from the application of specific deep composition rules (for example, projective commutativity, bringing about non-canonical word order) whether a dependent is contextually bound or nonbound.

4 Commutativity and Informativity

The basic idea explored here is rather simple: When dependents occur in a position different from the position specified in the category of a head (mirroring the head's valency frame), then such can be taken as an indication of the dependent's informativity. The hypothesis is particularly applicable to languages like Czech, and probably also Japanese, where it is primarily the *structure* that provides an indication of how informative particular dependents are - contrary to English, which is a relatively fixed word order language, in which particularly *intonation* is used to indicate informativity.

First, let me recapitulate the linguistic ideas behind FGD's notions of contextual boundness and nonboundness.

4.1 Linguistic Intuitions

Contextual boundness and nonboundness are primary linguistic notions used to classify semantemes in a tectogrammatical representation as reflecting a speaker's disposition towards the actual state of affairs talked about, and his efforts to accommodate the hearer's needs as to be able to interpret what the speaker intents to convey (cf. [45], p.177). Thereby, contextual boundness can be understood as the linguistically determined counterparts of cognitive notions such as salience, 'given', recoverable from the already established discourse context; whereas contextual nonboundness primarily corresponds to 'novelty', not indicating a reference to something established but signalling the *introduction* something new into the context, a choice among "competing" entities in the context, or the *modification* of something recoverable.

It should be noted that the property of being contextually bound or nonbound is a *local property*, namely localized to the governing head. A consequence of this localization is that we can try to 'bundle' a head with its contextually bound and nonbound elements, in order to get more abstract view of what is 'informatively' going on in the deep structure.

Here, we will conceive of a clause's verbal head and its contextually bound and nonbound dependents as a *CB/NB-configuration*. Furthermore, the more semantic counterpart of a CB/NB-configuration we take to be a topic/focus-articulation - which, due to the locality of the CB/NB-configuration, therefore also becomes a concept which starts operating already at clause-level (and not at sentence-level). The nice consequence of a clause-level perspective on topic/focus-articulation is that a complex sentence gets an embedded topic/focus structure. Notably, embedded clauses will have their own topic and focus, which, by the dependency on a higher clause, will relate to the higher level topic or focus depending on what the higher level clause belongs to.

4.2 Structure and Informativity

How do structure and informativity relate? As was already pointed out in Chapter 1, there is an ordering defined over all the dependencies distinguished for a particular language: the so-called *systemic ordering*. Important is that the systemic ordering is not just an abstract ordering. It is also reflected in a word's valency frame: for all dependencies D_i, D_j in that valency frame, it holds that $D_i \prec_{so} D_j$ implies that D_i precedes D_j in the valency frame. The relation between deep word order and surface word order can be sketched as follows. Assume we have a tectogrammatical representation in which all dependents are ordered according to the systemic ordering - or more precisely, the *dependents are ordered according to their respective dependency relations by which they modify the head.* Then, the sentence realizing that tectogrammatical representation would display the standard surface word order. We call such sentences with tectogrammatical representations that have all their dependents ordered according to the systemic ordering, *primary cases. Secondary cases* are those sentences which have strings that, once functionally interpreted, result in a tectogrammatical representation in which some dependents are *not* ordered according to the systemic ordering. Slightly rephrased, in primary cases dependents occur in what can be called *canonical order*, whereas in secondary cases there are dependents that occur in *non-canonical order.*

The main hypothesis underlying the discussion below is then that

> Structural indications of informativity arise by the interplay between (non-)canonical ordering and systemic ordering.

More precisely, a small number of hypotheses can be put forward which spell out this interaction in more explicit terms. Particularly, for DBLG, the following observations can be made. In DBLG, categories for heads formalize the notion of valency frame, and as such systemic ordering is maintained in the categories as well. As long as a deep mode of composition is used which is non-commutative, canonical sentences will be analysable. In other words, as soon as commutativity comes into play, non-canonical ordering can arise. Below I will argue that each of the hypotheses can conveniently be formalized using restricted forms of commutativity.

4.3 Basic Hypotheses

Thus far, contextual boundness and nonboundness have been described in the literature primarily from a generative point of view. That is, given a particular content a speaker would want to convey, how would a dependency structure have to be composed, such that a sentence could be generated that would convey that content?

Hajičová et al present in [10] a more analytic perspective by discussing an algorithm for identifying the possible articulation(s) of topic and focus. As the authors note, though, the algorithm holds only for a simple type of English sentences (namely those of primary case), and should be extended so as to take into account also deeper embedded modifications.

The hypotheses and their formalization in terms of DBLG as presented in the next subsections draw upon [10].

Consider the following quotation from [45]:

> *If no surface 'movement' rule intervenes (i.e esp. in cases of "free" word order), SO may be determined as follows: given a governing word that can be expanded by two different modifications, A and B, occurring in the surface in this order (with normal intonation), neither of them being CB, and given further that the surface order BA (or, in the case of A preceding B, placing the intonation center on A), is possible only if B is CB, then A precedes B under SO.* (pp.194-195)

Instead, if we take an analytic perspective, for which we can assume that the systemic ordering of kinds of dependency relations (SO) is given, then the following hypothesis can readily be formulated.

Hypothesis I *Given* a valency frame $\mathcal{V}$ for a word w, and two -different- slots A and B in $\mathcal{V}$, with A preceding B (in SO, and hence in $\mathcal{V}$). *If* strings s_1, s_2 are functionally interpreted as modifying w by dependency relations B and A, respectively, and string s_1 precedes s_2 in the surface form, *then* the modifier corresponding to s_1 is CB.

This hypothesis is highly similar to Hajičová et al's Rule 1'. However, what our hypothesis does not cover is the following ordering:

(5.3) a. Yesterday it rained.
b. It rained yesterday.

The ordering of "yesterday" in a (non-canonical) position to the right of its canonical position leads us to consider "yesterday" in (5.3b.) as contextually nonbound. Which can be easily verified by reading the sentence out loud with a standard intonation. The reason why hypothesis I does not cover (5.3) is because that hypothesis concerns *leftward* ordering, whereas here we are dealing with *rightward* ordering. Therefore, we could come to entertain the following hypothesis, besides hypothesis I:

Hypothesis II *Given* a valency frame $\mathcal{V}$ for a word w, and two -different- slots A and B in $\mathcal{V}$, with A preceding B (in SO, and hence in $\mathcal{V}$). *If* strings s_2, s_1 are functionally interpreted as modifying w by dependency relations A and B, respectively, and string s_2 succeeds s_1 in the surface form, *then* the modifier corresponding to s_2 is NB.

Remark 4.1 The problem of hypothesis II is that it is not always directly applicable. It clearly depends on the language, as for how far right movement is allowed. Thus, we would perhaps want to have a less general version - in Tamil, or Sinhala, only those elements (rightwards) moved into immediate preverbal or postverbal positions are to be considered contextually nonbound, whereas in Japanese it is only a dependent occurring in the directly preverbal position that -arguably- should be considered nonbound.

On the other hand, if we would formulate (different kinds of) commutativity such that these restrictions would be taken into account, hypothesis II could be properly relativized.

Hypotheses I and II are concerned with non-canonical ordering in the sense of a leftward or rightward "movement", respectively. However, neither hypothesis says something about elements that remain in situ, but whose CB/NB-ness may depend on the fact that *other* elements occur in non-canonical ordering.

Example 4.1 Consider for example the following two sentences:

(5.4) a. It rains in Prague.

b. In Prague it rains.

If we assign to "rains" a category $ep \backslash_e \div_d^2 \{\boxed{\mathrm{s}}, Location\}$, then (5.4.a) is a primary case. In this unmarked case, the verb is considered CB, and the dependent NB - although this does not follow from the hypotheses. For (5.4.b) it is the case that, by hypothesis I, the dependent "in Prague" can be considered CB, but that still leaves the verb's informativity in the middle.

Therefore, a third hypothesis is needed that covers the primary case, and cases like (5.4.b) where the node itself remains in situ, though whose contextually boundness or nonboundness depends on the context, i.e. the non-canonical ordering of other nodes.

Hypothesis III.a In a primary case, all dependents occurring before the head are considered contextually bound, and all dependents occurring after the head are considered contextually nonbound.

Hypothesis III.b Given an appropriate notion of sentence-finality (i.e. right before the verbal head like in Japanese, or end of sentence like in English or Czech), then the rightmost element in the dependency structure, including the verbal head, is considered contextually nonbound.

Remark 4.2 Before discussing how these hypotheses could be incorporated in DBLG, I should remark that the hypotheses are aimed to provide an outset from which more elaborate cases can be considered. By no means the hypotheses above should be taken as all-encompassing.

4.4 Formalization

For the formal incorporation of the hypotheses, I first introduce two kinds of commutativity: *Leftwards projective commutativity* (pcl) and *rightwards projective commutativity* (pcr). Both are structural rules in which nodes (possibly being entire subtrees) are moved within the domain of a $\times$-composition - therefore, they are projective since no crossing dependencies will arise from this kind of commutativity.

Definition 4.1 (Left- and Rightward Projective Commutativity)

$$\frac{\begin{array}{c} x \vdash C_i : M^{A_x} \quad y \vdash C_j : L^{A_y} \\ \vdots \\ s[(..x..y..)^{\times}] \vdash C_k : N^B[M^{A_x}, L^{A_y}] \end{array}}{s[(..y..x..)^{\times}] \vdash C_k : N^B[L^{A_y}, M^{A_x}]}\ pcl$$

$$\frac{\begin{array}{c} y \vdash C_j : L^{A_y} \quad x \vdash C_i : M^{A_x} \\ \vdots \\ s[(..y..x..)^{\times}] \vdash C_k : N^B[L^{A_y}, M^{A_x}] \end{array}}{s[(..x..y..)^{\times}] \vdash C_k : N^B[M^{A_x}, L^{A_y}]}\ pcr$$

■

Remark 4.3 The distinction between leftwards and rightwards should be clear: In [pcl] y is moved to the left of x, whereas in [pcr] y is moved to the right of x. The distinction is real since both inferences are one-way, contrary to standard commutativity. Projective commutativity, [pc], is simply defined as the combination of both [pcr] and [pcl].

Subsequently, if a grammar for a specific language is written as a DBLG, we can make use of [pcl] and [pcr] in combination with a set of suitable hypotheses to determine the contextual boundness or nonboundness of nodes in a dependency structure.

Example 4.2 Hypotheses I through III are applicable to (relatively simple) Czech sentences. Their formalization is as follows.

Notation 4.1 *Within a term (i.e. the dependency structure) I will mark a contextually bound node using a superscript cb, and a contextually nonbound node using a superscript nb.*

Definition 4.2 (Basic CB/NB for Czech) *Given Hypotheses I through III, the following structural rules model a basic theory about contextual boundness and nonboundness for Czech:*

Hypothesis I:

$$\frac{\begin{array}{c} x \vdash C_i : M^{A_x} \quad y \vdash C_j : L^{A_y} \\ \vdots \\ s[(..x..y..)^{\times}] \vdash C_k : N^B[M^{A_x}, L^{A_y}] \end{array}}{s[(..y..x..)^{\times}] \vdash C_k : N^B[(L^{A_y})^{cb}, M^{A_x}]}\ pcl$$

Hypothesis II:

$$\dfrac{\dfrac{y \vdash C_j : L^{A_y} \quad x \vdash C_i : M^{A_x}}{\vdots}\quad s[(..y..x..)^{\times}] \vdash C_k : N^B[L^{A_y}, M^{A_x}]}{s[(..x..y..)^{\times}] \vdash C_k : N^B[M^{A_x}, (L^{A_y})^{nb}]}\ pcr$$

Hypothesis III.a:

$$\dfrac{\dfrac{s_1 \vdash C_1 : M^{A_{s_1}} \ldots \quad t \vdash C_h : \div_i^n\{C_1, ..., C_{m-1}, \boxed{C_h}, C_{m+1}, ..., C_n\} \quad \ldots s_n \vdash C_n : M^{A_{s_n}}}{s[(s_1, ..., s_{m-1}, t, s_{m+1}, ..., s_n)^{\times_j^n}] \vdash C_h : M^{A_{s_1}} ... M^{A_{s_{m-1}}} N^B M^{A_{s_{m+1}}} ... M^{A_{s_n}}]}}{s[(s_1 ..., t, ...s_n)^{\times_j^n}] \vdash C_h : S[(M^{A_{s_1}})^{cb} ... (M^{A_{s_{m-1}}})^{cb} (N^B)^{cb} (M^{A_{s_{m+1}}})^{nb} ... (M^{A_{s_n}})^{nb}]}\ can$$

Hypothesis III.b:

$$\dfrac{(s_1, ..., t, ..., s_n)^{\times_i^n} \vdash s : (M^{A_{s_1}})^x ... (M^{A_{s_{m-1}}})^y (N^B)^z (M^{A_{s_{m+1}}})^w ... (M^{A_{s_{n-1}}})^v (M^{A_{s_n}})^u}{(s_1, ..., t, ..., s_n)^{\times_i^n} \vdash s : (M^{A_{s_1}})^x ... (M^{A_{s_{m-1}}})^y (N^B)^z (M^{A_{s_{m+1}}})^w ... (M^{A_{s_{n-1}}})^v (M^{A_{s_n}})^{nb}}\ is$$

■

Remark 4.4 In [is], the variables u,v,w,x,y,z are meant indicate that either the nodes already have received a cb/nb marking, or not yet.

Example 4.3 Let me illustrate the rules defined above with two examples.

(5.5) Koupil kluk koblihu.

(5.6) Kluk koupil koblihu.

The first sentence can be translated as "A boy bought a donut" whereas the second sentence can be translated as "The boy bought a donut". The definiteness of "boy" in the second sentence is due to the fact that it should be judged *cb*, whereas in the first sentence both dependents should be considered *nb*.

- Signature: Given a standard, non-commutative mode of deep composition d, the linkage axiom schema is instantiated for the couples [d/pc] and [d/sc] (that is, d can be changed into either pc or sc). Mode sc has access to [pcl], [pcr] and [is], whereas mode pc has access to [can]. Dependencies are as in the appendix; features are left out for the sake of brevity. $\otimes$ stands for functional interpretation, as usual.

- Lexicon: {
$(kluk \vdash n : boy)$,
$(koupil \vdash \div^3_d\{\boxed{s}, Actor, Patient\} : \lambda^r x_a, x_p buy\{(Actor : x_a), (Patient : x_p)\})$,
$(koblihu \vdash n : donut)$
}

The analysis for the first sentence is straightforward. Observe that, after the dependents and the head have been composed into a dependency structure, the mode of composition d is changed into pc, after which the structural rule corresponding to Hypothesis III.a (primary case/canonical ordering) can be applied.

$$\dfrac{\dfrac{\dfrac{\begin{array}{c} koupil \vdash \div^3_d\{\boxed{s}, Actor, Patient\} : \lambda^r x_a, x_p buy\{(Actor : x_a), (Patient : x_p)\} \\ kluk \vdash n \otimes Actor : boy \\ koblihu \vdash n \otimes donut : donut \end{array}}{(koupil, kluk, koblihu)^{\times^3_d} \vdash s : buy\{(Actor : boy), (Patient : koblihu)\}} \div E}{(koupil, kluk, koblihu)^{\times^3_{pc}} \vdash s : buy\{(Actor : boy), (Patient : koblihu)\}} d/pc}{(koupil, kluk, koblihu)^{\times^3_{pc}} \vdash s : buy^{cb}\{(Actor : boy)^{nb}, (Patient : koblihu)^{nb}\}} can$$

The analysis of the second case is slightly more involved.

$$\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c} koupil \vdash \div^3_d\{\boxed{s}, Actor, Patient\} : \lambda^r x_a, x_p buy\{(Actor : x_a), (Patient : x_p)\} \\ kluk \vdash n \otimes Actor : boy \\ koblihu \vdash n \otimes donut : donut \end{array}}{(koupil, kluk, koblihu)^{\times^3_d} \vdash s : buy\{(Actor : boy), (Patient : koblihu)\}} \div E}{(koupil, kluk, koblihu)^{\times^3_{sc}} \vdash s : buy\{(Actor : boy), (Patient : koblihu)\}} d/sc}{(kluk, koupil, koblihu)^{\times^3_{sc}} \vdash s : \{(Actor : boy)^{cb}\}buy\{(Patient : koblihu)\}} pcl}{(kluk, koupil, koblihu)^{\times^3_{sc}} \vdash s : \{(Actor : boy)^{cb}\}buy\{(Patient : koblihu)^{nb}\}} is$$

The reader should observe that this time, we are unable to decide (given the structure) whether the verb should be contextually bound or nonbound. Following [10] the verb can be considered ambiguous in this respect.

For the dependents it can be decided whether they should be contextually bound (the Actor) or nonbound (the Patient). What is important here is the move from d to sc: sc makes it possible to move resources around, so that we can arrive at the observed surface form. It is also due to this move (by linkage) that [*can*] can not be used in this derivation: there is no linkage between sc and pc.

4.5 Topic/Focus-Articulation

Finally, let me briefly elaborate on how the above discussion relates to the Praguian theory of Topic/Focus-Articulation.

The notions now called 'topic' and 'focus' can be traced back to the work of Weil mid-nineteenth century. Weil's work was resumed by several German linguists in the decades around the turn of the millennium, and subsequently in the Prague Circle by Mathesius, recognizing that the distinction between topic and focus was important to problems ranging from intonation to word order - issues central to the description of natural language. Within FGD, the theory of topic/focus-articulation was further developed by Sgall and his collaborators, in particular Hajičová.

A distinguishing characteristic of the Praguian notions of 'topic' and 'focus' is that they are not primary, but are derived from (based on) the structural notions of contextual boundness and non-boundness[1]. The following abstract definition of topic and focus as to their relation to contextual boundness and nonboundness is given by Sgall et al in [45] (p.216f):

- The main verb belongs to the focus if it is NB, and to the topic if it is CB;
- the NB nodes depending on the main verb belong to the focus, and so do all nodes (transitively) subordinated to them;
- if some of the elements of the tectogrammatical representation belong to its focus according to either of the above points, then every CB daughter of the main verb together with all nodes (transitively) subordinated to it belong to the topic;
- if no node of the tectogrammatical representation fulfills the first two points above, then the focus may be more deeply embedded.

It is this aspect of being *derived* that leads us to view topic/focus-articulation as symbolizing the interface between discourse and sentential structure, rather than a structural characteristic of the sentence. This interpretation is one of the subtly different interpretations one might distinguish in modern Prague School writings on this issue; cf. [8, 42, 39].

What is important to observe about the perspective of topic and focus being derived is that we arrive at different predictions as for what is in the focus, and what is in the topic. Consider for example sentence (5.7), which Vallduví and Engdahl provide as an illustration of an all-focus sentence:

(5.7) [He LOVES it.]$_f$

On the FGD-based account, however, every weak (unaccented) pronoun is considered as contextually bound, and as such we would *only* predict the narrow focus

[1] As compared to a range of other theories concerning semantic topic and focus; see Vallduví and Engdahl's [50].

(5.8) He $[\textsc{loves}]_f$ it.

We should note here that even though Vallduví and Engdahl do consider the weak pronouns 'He' and 'it' "inert" with regard to the focus, which might therefore as well be left out (p.476), they do seem to claim that the pronouns would not have to be grounded (p.475). Thus, there we arrive at a more subtle difference between these approaches (and as such relativizes the claim on (p.471) that their approach covers all informational constructions found in the literature).

Another interesting consequence of the derived notion of topic and focus is that, whereas some authors (like Halliday, Dahl) have opted to include both topic-comment and focus-ground in the informational structure to cover the following ambiguity

(5.9) a. $[\text{John}]_{topic}$ $[\text{drinks BEER.}]_{comment}$

b. $[\text{John drinks}]_{ground}$ $[\text{BEER}]_{focus}$

or adopt a tripartite structure instead of a bipartite one (Vallduví), the CB/NB-based account simply considers the verb ambiguous as for its CB/NB-ness. It is not structurally determinable whether the boundary should be strictly before or after the verb - see Hajičová et al's [10] for an account that (algorithmically) describes this phenomenon.

On the other hand, the characterization of contextual boundness and nonboundness we employ here is not related to intonation (yet). Proceeding strictly from word order, we are thus facing a problem when elements 'intended' for topic occur in canonical position:

(5.10) a. *What did Mary give to Harry?*

b. Mary gave a BOWTIE to Harry.

Because "to Harry" occurs in canonical position, it would be considered contextually nonbound - whereas, given the prosody, it should be characterized as bound given that it follows a H^* accounted unit. We leave this issue as a topic for further research, and just note within FGD the interaction between structure and prosody has received ample attention (cf. [45]), which could be formalized within the setting of DBLG mirroring approaches like the ones advocated by Oehrle [37] or Hendriks [12].

Similarly, there are cases in English where the *focus* appears fronted (instead of in situ). [50] cites the following example by Hannay:

(5.11) a. *Did you get wet?*

b. Bloody SOAKING I was.

This phenomenon (called Y(ddish)-movement, rhematization, et cetera) is not predicted by our account here. It could easily be incorporated, though, by adhering to Rochemont's proposal which requires that *all* foci get fronted in such a case. A structural rule that would allow (enforce) all the right canonical (default NB) elements to move in front of the left canonical elements (default CB), could achieve that[2].

Obviously, there are various other, similar phenomena that would need more attention - the relation between structure and kinds of foci like *verum focus* [50], *presentational focus* [17], et cetera.

[2] Again, when assuming neutral intonation; if we have a stressed verb, then the modifiers succeeding the verb are considered contextually bound (Sgall, p.c.).

Chapter 6

A Few Arguments for DBLG

1 Remarks on the Dependency/Constituency Debate

Perhaps ever since the entering of Chomsky on the stage of formal linguistics, and the accompanying turn to perceiving syntactic structure in terms of constituency, dependency grammarians have been campaigning for their cause against what they see as a denial of a well-established view of syntax, confirmed by the test of time. After all, as for example Mel'čuk notes in [29] (p.24), within various grammatical traditions dependency trees had been independently accepted as descriptions of natural language syntax - ever since Antiquity. Constituency, on the other hand, was devised only at the beginning of this century by the German psychologist Wundt, and brought into the realm of linguistics by Bloomfield in the thirties.

To set the two approaches apart as *theories* about natural language syntax, we may want to conceive of them as going on the following two hypotheses, respectively:

The Dependency Hypothesis: The structure of natural language can be explained in terms of how distinguishable units are *related* by semantically motivated relations.

The (Immediate) Constituency Hypothesis: [47], p.73:"[T]he main aspects of the syntactic patterning of the sentence are appropriately captured by dividing it into n parts, each of which again can be divided into parts of its own, etc., until individual word forms are reached."

Whether either hypothesis can indeed count as an explanation, leading to theory, should ultimately depend on whether it can be verified by the empirical data it purports to explain: natural language syntax. Dependency grammarians, as said above, content that (what is called here) the dependency hypothesis has been verified to a large degree. Time, and in particular, descriptive adequacy have told, as dependency they argue [29, 45, 47, 44].

Describing the syntactic structure of a sentence in terms of dependency relations between heads and dependents shows an interesting balance between what one could call *flexibility* and *fine-grainedness*:

Flexibility , in what may be taken to function as a dependent,

Fine-grainedness , *classificatory* in terms of being able to distinguish parts of a sentence not only by their internal structure, but also by their function (i.e. involvement in the dependency structure), and *structurally* in terms of there being no superfluous information in a dependency structure.

From the relational nature that is inherent to the perspective on syntax, flexibility is obtained by describing syntax in terms of *how* units should be related (vz. the valency frame), but allowing the form of these units to be of 'any' linguistically reasonable kind. This is distinctly opposite to the constituency approach, since there the forms are directly determined by the larger constituents/phrases in which they appear. This kind of flexibility is most easily illustrated by examples involving coordination:

(6.1) I believe X drinks his tea with and Y without sugar.

Remark 1.1 Needless to say that people working in the constituency-based approaches have tried to address this particular problem concerning coordination. Usually, (partial) solutions involved the introduction of "flexible constituents" (Steedman) or "dependency constituents" (Barry/Pickering [1]).

On the other hand, the dependency approach inherently has a fine-grainedness not easily matched by the constituency approach. One and the same form will, under the constituency approach, be classified as one and the same phrase, but may, under the dependency approach, be related to a head by different dependencies, depending on the context. An empirical example is that of complex fronting of groups of wordforms that are mostly PPs, but belong -generally- only to a small subset of dependency relations.

Fine-grainedness also shows itself in the dependency structures, where the head/dependent-asymmetry together with a tree containing no nonterminals makes it possible to easily characterize, for example, word order phenomena like verb-secondness or the Wackernagel position. Namely, the second position is simply directly after the leftmost dependent.

Remark 1.2 Naturally, the arguments above -in favor of a dependency-based approach- are not all the arguments advanced by dependency grammarians. For more arguments, see the references cited above. My reason for putting forward the arguments as I did is primarily that they appear quite clear and indisputable.

Dependency-based grammar is primarily put forward as a *descriptive* approach to natural language syntax, rather than a *formal grammar*[1]. Nevertheless, dependency-based frameworks have often been blamed for being "inconsistent", "incomplete", and "unverifiable" by various authors (for example, Bröker, Dressler), up to this day. As a result, perhaps, one can observe a trend to incorporate ideas from dependency-based grammar into formal grammars, rather than providing a formal, dependency-based grammar. Consider for example the incorporation of the head/dependent-asymmetry in Categorial Grammar [33, 1, 16], Head-Phrase Structure Grammar [43], or Dynamic Dependency Grammar [30].

2 Some Arguments for DBLG

What arguments can be given for the viability of DBLG? DBLG, being dependency-based in the sense of incorporating from the start both the head/dependent-asymmetry and dependency relations. The advantageous flexibility and fine-grainedness can be obtained in DBLG due to the use of dependencies as categories, and the formalization of functional interpretation ($\otimes$) and realization ($\oplus$) - provided, of course, that a proper model is given for $\otimes$ and $\oplus$.

What is more, it can be hoped that DBLG will be able to draw on the rich body of descriptive, linguistic theory that has been developed within the Prague School, in which much of contemporary issues have been addressed to quite some depth. A notable example is the role of informativity in Prague School's FGD, in the form of contextual boundness/nonboundness and the therefrom derived notion of topic/focus-articulation. In the previous chapter I showed how some basic intuitions about contextual boundness and nonboundness can be formally incorporated into a dependency-based logical grammar for Czech.

Finally, DBLG combines linguistic insights from the tradition of dependency-based grammar with a contemporary, powerful grammar formalism, namely multimodal logical grammar MMLG. That brings a dependency-based grammar into the realm of formal grammars.

3 Final Remarks

DBLG, as I developed it here, is of course not without any drawbacks. One criticism one may levy against DBLG is that a reasonable amount of decoration is introduced in order to deal with specific phenomena (like option-

[1] Whereby I understand a 'formal grammar' to be a grammar which is based on a well-defined logic that is shown to guide every description of a phenomenon, rather than a grammar which uses mathematical tools in its basic definitions. In this sense, HPSG is not a formal grammar either, since the (ontological) foundations of HPSG (in the 1994 version) are still to be formalized - see for example[23].

ality/obligatoriness of dependents, dependent-head agreement, etcetera), *in the context of n-ary composition.* Alternatively, if DBLG would have employed *binary* modes in both dimensions, decoration could be replaced by different modes of composition that would properly capture the desired behavior.

There is, however, a drawback to this solution. Although I find it technically enticing, it would also introduce a notion of "non-terminal" in the grammar. In DBLG as it stands, a valency frame is either entirely filled, or entirely empty - there are no inbetweens. If binary modes would be introduced, it would be possible for a valency frame to be only partially filled (or partially empty). Which goes against the intuitions of dependency-based grammar - as soon as a Curry-Howard isomorphism would be established between formulas and terms, the terms-as-dependency structures would include nonterminals. (Needless to say, there may be the possibility to define a contraction/normalization over terms to get rid of 'spurious information' like nonterminals.)

It remains a topic for further research whether n-ary products, modelling modes of composition, could be defined such as to enable one to dispense with decoration.

Other points that deserve further research are the definitions of free modification and of optionality. The reader familiar with work in linear logic may see the connection between these two definitions, and the "!" operator in linear logic that makes contraction and weakening accessible: For optionality, contraction would perhaps be a more natural formal definition, and weakening for free modification, instead of the definitions given in this report.

Finally, the relation between formulas and terms in a multi-modal logical grammar like DBLG deserves perhaps more attention - particularly, what it would mean to have an isomorphism rather than a correspondence, and what that would require in terms of the semantics ("model theory") underlying an MMLG (defining the intended meaning of its modal operators).

Appendix A

Dependencies and Features

In this appendix I give a list of kinds of dependencies, and types of features, as currently employed in DBLG.

1 Dependencies

Below the distinguished dependencies are given, in the order of the systemic ordering (Chapter 1; see also [45]). For a more exhaustive list of dependencies, see [42]. The numbers occurring before the dependencies correspond to their placement in the list of [42].

	Dependency	Description	Can appear on
3	Actor	Inner Participant	verbs
6	Time (when)	Free modifier	verbs
16	Manner	Free modifier	verbs
23	Location	Free modifier	nouns, verbs
30	Patient	Inner Participant	nouns, verbs
33	Effect	Inner Participant	verbs
36	Gen.Rel	Free modifier	nouns
38	Descr.Pr.	Free modifier	nouns

Remark 1.1 As [42] remarks, a Descriptive Property "denotes a property that does not restrict the semantic extent of the noun (*golden* Prague; *sweet* France)" (p.61) - unlike a General Relationship which "is restrictive and expressed by an adjunct or relative clause" (p.60).

2 Features

The table below describes the features -currently- distinguished in DBLG. For the technical descriptions, see Chapter 3.

General	Specific	Description	Can appear on
DEFIN	DEF,INDEF	Definiteness: DEF=definite, INDEF=indefinite	nouns
CASE	NOM, ACC DAT, INST	Case: NOM=nominative, ACC=accusative DAT=dative, INST=instrumental	nouns
GEN	FEM, MAS	Gender: FEM=feminine, MAS=masculine	nouns, verbs
NUM	SG, PL	Number: SG=singular, PL=plural	nouns, verbs
TENSE	PAST, PRES	Tense: PAST=Past tense, PRES=Present tense	verbs
PER	3RD,1ST	Person: 3RD = Third person, 1ST=First person	verbs

Bibliography

[1] Guy Barry and Martin Pickering. Dependency and constituency in categorial grammar. In Alain Lecomte, editor, *Word Order in Categorial Grammar/L'Ordre des Mots dans les Grammaires Catégorielles*, pages 39–57. Adosa, Clermont-Ferrand, France, 1992.

[2] Bob Carpenter. *Type-Logical Semantics.* The MIT Press, Cambridge Massachusetts, 1996.

[3] Thomas L. Cornell. A type-logical perspective on minimalist derivations. In Geert-Jan M. Kruijff, Glyn V. Morrill, and Richard T. Oehrle, editors, *Formal Grammar 1997, Proceedings of the Conference held in Aix en Provence, France, August 9-10*, 1997.

[4] Haskell B. Curry. Some logical aspects of grammatical structure. In Roman O. Jakobson, editor, *Structure of Language in its Mathematical Aspects. Proceedings of the 12th Symposium in Applied Mathematics*, pages 56–68, Providence, 1961. American Mathematical Society.

[5] Haskell B. Curry. *Foundations of Mathematical Logic*, volume 2 of *McGraw-Hill Series in Higher Mathematics.* McGraw-Hill, New York etc., 1963.

[6] Dov M. Gabbay. *Labelled Deduction Systems: Volume 1*, volume 33 of *Oxford Logic Guides.* Clarendon Press, Oxford, New York, 1996.

[7] L. Haegeman. *Introduction to Government and Binding Theory.* Basil Blackwell, Oxford, 1991.

[8] Eva Hajičová. *Issues of sentence structure and discourse patterns*, volume 2 of *Theoretical and Computational Linguistics.* Charles University Press, Prague, Czech Republic, 1993.

[9] Eva Hajičová, editor. *Text-and-Inference Based Approach to Question Answering*, volume 3 of *Theoretical and Computational Linguistics.* Charles University Press, Prague, Czech Republic, 1995.

[10] Eva Hajičová, Hana Skoumalová, and Petr Sgall. An automatic procedure for topic-focus identification. *Computational Linguistics*, 21(1):81–94, March 1995.

[11] Herman Hendriks. *Studied Flexibility: Categories and Types in Syntax and Semantics.* Illc dissertation series 1993-5, Universiteit van Amsterdam, Amsterdam, The Netherlands, December 1993.

[12] Herman Hendriks. The logic of tune. In Geert-Jan M. Kruijff, Glyn V. Morrill and Richard T. Oehrle, editor, *Formal Grammar 1997, Proceedings of the Conference held in Aix en Provence, France, August 9-10, 1997*, 1997.

[13] Mark Hepple. A general framework for hybrid substructural categorial logics. Technical Report 94-14, IRCS, University of Pennsylvania, Philadelphia PA, September 1994.

[14] Mark Hepple. Hybrid categorial logics. *Bulletin of the IGPL*, 3(2,3):343–356, June 1995. Special Issue on Deduction and Language.

[15] Mark Hepple. Mixing modes of linguistic description in categorial grammar. In *Proceedings EACL-7, Dublin Ireland*, 1995.

[16] Mark Hepple. A dependency-based approach to bounded and unbounded movement. In *Proceedings of the Fifth Meeting on Mathematics of Language (MOL-5)*, 1997. DFKI-D-97-02.

[17] Susan C. Herring and John C. Paolillo. Focus position in SOV languages. In Pamela Downing and Michael Noonan, editors, *Word Order in Discourse*, volume 30 of *Typological Studies in Language*, pages 163–198. John Benjamins, Amsterdam, The Netherlands, 1995.

[18] Dirk Heylen. Generalisation and coordination in categorial grammar. In Geert-Jan M. Kruijff, Glyn V. Morrill, and Richard T. Oehrle, editors, *Formal Grammar 1997, Proceedings of the Conference held in Aix en Provence, France, August 9-10, 1997*, 1997.

[19] Dirk Heylen. Underspecification in subsumption-based type logical grammars. In *Logical Aspects of Computational Linguistics*, Nancy, 1997.

[20] Roman Jakobson. *On Language*. Harvard University Press, Cambridge MA, 1990. Edited by Linda R. Waugh and Monique Monville-Burston.

[21] Mark Johnson. Resource-sensitivity in Lexical-Functional Grammar. In *Proceedings of the 1996 Roma Workshop*, 1996.

[22] Mark Johnson. Features as resources in R-LFG. In *Proceedings of the 1997 LFG Conference*, Stanford CA, 1997. CSLI Publications.

[23] Paul J. King and Carl J. Pollard. A formalism for HPSG'94. Unpublished manuscript. Available from ..., October 1997.

[24] A. Kosík and Petr Sgall. An intensional interpretation of linguistic meaning. *Prague Bulletin of Mathematical Linguistics*, 51:25–40, 1989.

[25] Esther Kraak. French object clitics: a multimodal analysis. In *Formal Grammar 1995*, 1995.

[26] Natasha Kurtonina. *Frames and Labels: A Modal Analysis of Categorial Inference.* PhD thesis, OTS, Utrecht University, 1994.

[27] Natasha Kurtonina and Michael Moortgat. Structural control. In Patrick Blackburn and Maarten De Rijke, editors, *Specifying Syntactic Structures.* CSLI Publications & FoLLi, Stanford CA, 1996.

[28] Maarten Marx and Yde Venema. *Multi-Dimensional Modal Logic*. Number 4 in Applied Logic Series. Kluwer Academic Publishers, Dordrecht, Boston, London, 1997.

[29] Igor A. Mel'čuk. *Dependency Syntax: Theory and Practice.* SUNY Press, Albany NY, 1988.

[30] David Milward. Dynamics, dependency grammar, and incremental interpretation. In *COLING'92*, 1992.

[31] Michael Moortgat. Multimodal linguistic inference. *Bulletin of the IGPL*, 3(2,3):371–402, June 1995. Special Issue on Deduction and Language.

[32] Michael Moortgat. Categorial type logics. In Johan van Benthem and Alice ter Meulen, editors, *Handbook of Logic and Language*. Elsevier Science B.V., Amsterdam New York etc., 1997.

[33] Michael Moortgat and Glyn Morrill. Heads and phrases: Type calculus for dependency and constituent structure. *Journal of Logic, Language and Information*, 1991.

[34] Michael Moortgat and Dick Oehrle. Adjacency, dependency and order. In *Proceedings of the Ninth Amsterdam Colloquium*, 1994.

[35] Glyn V. Morrill. *Type Logical Grammar: Categorial Logic of Signs*. Kluwer Academic Publishers, Dordrecht, Boston, London, 1994.

[36] Reinhard Muskens. *Meaning and Partiality*. Series in Logic, Language and Information. CSLI Publications & FoLLi, Stanford CA, 1995.

[37] Richard T. Oehrle. Substructural logic and linguistic inference. Draft manuscript, February 1997.

[38] Jarmila Panevová. Valency frames and the meaning of the sentence. In Philip Luelsdorff, editor, *The Prague School of Structural and Functional Linguistics*, volume 41 of *Linguistics and Literary Studies in Eastern Europe*, pages 223–243. John Benjamins, Amsterdam, 1994.

[39] Jaroslav Peregrin. Topic and focus in a formal framework. In Barbara H. Partee and Petr Sgall, editors, *Discourse and Meaning: Papers in Honor of Eva Hajičová*. John Benjamins, Amsterdam, The Netherlands, 1995.

[40] Vladimír Petkevič. A new dependency based specification of underlying representations of sentences. *Theoretical Linguistics*, 14:143–172, 1987.

[41] Vladimír Petkevič. *A new formal specification of underlying structures*. PhD thesis, Faculty of Mathematics & Physics, Charles University, Prague, Czech Republic, 1990.

[42] Vladimír Petkevič. A new formal specification of underlying structures. *Theoretical Linguistics*, 21(1):7–61, 1995.

[43] Carl Pollard and Ivan A. Sag. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, 1993.

[44] Petr Sgall. Underlying structures of sentences and its relations to semantics. In *Festschrift für Viktor Julevič Rozencvejg*, pages 273–282. Wiener Slawistischer Almanach, Vienna, 1992.

[45] Petr Sgall, Eva Hajičová, and Jarmila Panevová. *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. D. Reidel Publishing Company, Dordrecht, Boston, London, 1986.

[46] Petr Sgall, Ladislav Nebeský, Alla Goralčíková, and Eva Hajičová. *A Functional Approach to Syntax In Generative Description of Language.* Elsevier Science B.V., Amsterdam New York etc., 1969.

[47] Petr Sgall and Jarmila Panevová. Dependency syntax - a challenge. *Theoretical Linguistics*, 15(1):73–86, 1989.

[48] A.S. Troelstra and H. Schwichtenberg. *Basic Proof Theory.* Number 43 in Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge UK, 1996.

[49] Raymond Turner. Types. In Johan Van Benthem and Alice Ter Meulen, editors, *Handbook of Logic and Language*, pages 535–586. Elsevier Science B.V., Amsterdam New York etc., 1997.

[50] Enric Vallduví and Elisabet Engdahl. The linguistic realization of information packaging. *Linguistics*, 34:459–519, 1996.

[51] Johan van Benthem. *Language in Action: Categories, Lambdas, and Dynamic Logic.* North-Holland, Amsterdam NL, 1991.

[52] Heinrich Wansing. *The Logic of Information Structures.* Number 681 in Lecture Notes in Artificial Intelligence. Springer-Verlag, Berlin, 1993.