Unlike coordination in Polish: an LFG account

Agnieszka Patejuk

Rozprawa doktorska
przygotowana pod kierunkiem
dr. hab. Adama Przepiórkowskiego, prof. IPI PAN i UW

Acknowledgements

In the first place, I would like to thank Adam Przepiórkowski, my supervisor, without whom this work would never have started nor fishished. Without whom I would never have the chance to work with the fascinating area of syntax of coordination and LFG.

I am very grateful and lucky that I have had the opportunity to work with the great people at the Linguistic Engineering Group (LEG) and to interact with them. I would like to thank Marcin Woliński (who provided many helpful comments about the earlier version of this work and who patiently answered my questions when I started working with GFJP2 used by Świgra), Elżbieta Hajnicz (for discussions about Walenty and her tremendous work) and Katarzyna Krasnowska (for help with the LFG structure bank in INESS and POLFIE in general).

This work has greatly benefited from discussions with Mary Dalrymple (who came up with the off-path solution to unlike category coordination), Ron Kaplan and John Maxwell (who answered tons of my questions about the formalisation of coordination as well as about the possibilities and limitations of XLE).

I am also very grateful to Paul Meurer, the developer of INESS system, for creating such a wonderful piece of software for creating structure banks and for his INESS-related support. Another thank you goes to people working on the disambiguation of the LFG structure bank of Polish.

Last but certainly not least, I would like to thank my family for their love, unfailing support and patience.

Contents

1 Introduction 1
1.1 Aim and scope 1
1.2 Method(ology) 4
1.3 Overview of content 5

I Basics: adopting existing solutions 7

2 LFG basics 9
2.1 Basics: lexicon and rules 9
2.2 F-structure representation 12
2.2.1 Grammatical functions 12
2.2.2 F-structure wellformedness 14
2.2.3 More about notation 15
2.3 Prepositional phrases 19
2.4 LDDs, discourse functions and functional uncertainty 21
2.5 Coordination basics 23
2.6 Summary 25

3 Agreement and case assignment 27
3.1 Subject-verb agreement in Polish 27
3.1.1 Full agreement 27
3.1.2 Default agreement 28
3.1.2.1 Non-agreeing numerals 28
3.1.2.2 Clausal subjects 31
3.1.3 Single conjunct agreement 33
3.2 Case assignment in Polish 35
3.2.1 Assignment or checking? 35
3.2.2 Subject 36
3.2.3 Object 37
3.2.3.1 Basic generalisations 37
3.2.3.2 GoN in verb chains 39
3.2.3.3 Complex case 41
3.3 Summary 44

II Beyond basic coordination 45

4 Coordination of unlike categories 47

4.1 Introduction . . . 47
4.2 Grammatical function assignment . . . 49
4.2.1 Prototypical, without coordination . . . 49
4.2.2 Coordination issue . . . 50
4.2.3 Passivisation issue . . . 50
4.3 Imposing constraints: intuitions, issue . . . 51
4.4 Conservative solution using off-path constraints . . . 54
4.4.1 Introduction to off-path constraints . . . 54
4.4.2 Handling unlike category coordination using off-path constraints . . . 55
4.5 More data, complex interactions . . . 55
4.5.1 Unlike category subjects . . . 56
4.5.2 Unlike category objects . . . 60
4.5.2.1 Interaction with verb chains . . . 63
4.5.2.2 Lexical case, other clause types . . . 65
4.6 Alternative solution . . . 67
4.7 Control into selected conjuncts . . . 67
4.7.1 Which grammatical function? . . . 68
4.7.2 Establishing control . . . 69
4.7.2.1 First attempt (failed) . . . 70
4.7.2.2 Second attempt (failed in XLE) . . . 71
4.7.2.3 Third attempt (successful) . . . 72
4.7.3 Interaction with structural case assignment . . . 73
4.8 Examples of lexical entries . . . 75
4.9 Summary . . . 76

5 Coordination of unlike grammatical functions 79

5.1 Introduction . . . 79
5.2 Basic data and generalisations . . . 80
5.3 Is this coordination? . . . 82
5.4 Monoclausal or multiclausal: a critical review of tests for determining representation 83
5.4.1 Multiclausal analyses . . . 84
5.4.2 Critical review of selected arguments . . . 86
5.4.2.1 Sentence-level adverbs . . . 86
5.4.2.2 Clausal coordinators . . . 87
5.4.2.3 Degree questions . . . 89
5.4.2.4 Focus pronominal . . . 90
5.4.2.5 Rhetorical questions . . . 90
5.4.2.6 Voice mismatch . . . 91
5.4.2.7 Obligatory arguments . . . 92
5.4.2.8 Object agreement . . . 93

5.4.2.9 Distribution of question particles 94
5.4.2.10 Auxiliary between *wh*-phrases 96
5.4.2.11 Overt pronouns 97
5.4.2.12 Left-branch extraction (LBE) 98
5.4.2.13 Stranding 99
5.4.2.14 Governing numerals 101
5.4.2.15 Binding 102
5.4.2.16 Coordination with *yes/no* question particle 103
5.4.3 Summary and conclusion 104
5.5 Formalisation 105
5.5.1 Monoclausal coordination 108
5.5.2 Multiclausal coordination 110
5.5.3 Argument saturation under the multiclausal analysis 112
5.6 Problematic issues 115
5.6.1 Which elements are coordinated? 115
5.6.2 Representing the conjunction 116
5.6.3 Interference with verbal coordination 117
5.6.4 Embedded lexico-semantic coordination 119
5.6.5 Proposed solution 120
5.7 Improved analysis 121
5.7.1 Monoclausal structures 121
5.7.1.1 With verbal coordination 123
5.7.1.2 With embedding 124
5.7.2 Multiclausal structures 126
5.7.2.1 Multiclausal representation 126
5.7.2.2 Biclausal representation 129
5.7.3 Outstanding problematic issue 131
5.8 More types of conjuncts? 134
5.8.1 Demonstratives 135
5.8.2 Indefinites 136
5.8.3 Free relatives 137
5.8.4 Relatives 138
5.8.5 Modification 139
5.8.5.1 *Wszyscy/każdy*-based 139
5.8.5.2 Emphatic interrogatives 140
5.8.5.3 Based on existentials homophonous with *wh*-words 140
5.8.5.4 *Ktoś*-based 142
5.8.5.5 *Niektóry*-based 142
5.8.5.6 Comparatives 143
5.8.5.7 *N*-word-based 144
5.8.5.8 Demonstratives 144
5.8.6 Summary 145

5.9 Conclusion 145

III Implementation 147

6 About the implemented grammar 149

6.1 Introduction 149
6.2 Resources for grammar development 149
6.2.1 Previous grammars 150
6.2.2 Morfeusz 150
6.2.3 The National Corpus of Polish 150
6.2.4 Składnica 151
6.2.5 Walenty 151
6.3 Towards an LFG grammar for Polish 151
6.3.1 Annotating c-structure with f-descriptions 151
6.3.2 Lexicon creation 152
6.4 Quality control 153
6.4.1 Constructed testsuites 153
6.4.2 Treebank testing 153
6.4.3 NKJP1M testing 154
6.4.4 Building LFG structure bank for Polish 154
6.4.5 ParGram 155
6.5 Summary 156

7 Implementation basics 157

7.1 Introduction 157
7.2 Notation 157
7.2.1 Comments 157
7.2.2 Templates 157
7.2.3 Local variables 158
7.2.4 LFG notation vs XLE notation 159
7.3 Rewriting rules 159
7.4 Structural case assignment 163
7.4.1 Subject 163
7.4.2 Object 163
7.5 Verbal agreement 167
7.5.1 Full agreement 168
7.5.2 Default agreement 169
7.5.2.1 Non-agreeing numerals 170
7.5.2.2 Clausal subjects 171
7.5.2.3 Infinitival subjects 171
7.5.3 Single conjunct agreement 172
7.6 Summary 174

8 Valence information: converting Walenty to LFG 175
8.1 Introduction . . . 175
8.2 Formalism used in Walenty . . . 176
8.2.1 Grammatical function labels . . . 176
8.2.2 Syntactic positions as sets . . . 177
8.2.3 Unlike category coordination . . . 178
8.2.4 Arguments marked for structural case . . . 178
8.2.5 Control relations . . . 179
8.2.6 Adverbial-like complements classified according to semantic type . . . 181
8.2.7 Phraseological units . . . 182
8.2.7.1 Fixed expressions . . . 183
8.2.7.2 Lexicalised phrases . . . 183
8.3 Conversion to LFG constraints . . . 185
8.3.1 Choosing the grammatical function . . . 186
8.3.1.1 Without coordination . . . 186
8.3.1.2 Under coordination . . . 188
8.3.2 Imposing constraints . . . 190
8.3.2.1 Plain constraints . . . 190
8.3.2.2 Constraints under coordination . . . 193
8.3.2.3 Structural case assignment . . . 194
8.3.2.4 Realisations of container arguments . . . 196
8.3.2.5 Realisations of clausal arguments . . . 197
8.3.3 Passive voice formation . . . 199
8.3.4 Reducing frames . . . 202
8.4 Summary . . . 206

9 Implementation of unlike coordination 207
9.1 Introduction . . . 207
9.2 Coordination of unlike categories . . . 207
9.2.1 Basic rule . . . 207
9.2.2 Rule for control into selected conjuncts . . . 210
9.3 Coordination of unlike grammatical functions . . . 210
9.3.1 Top-level rule . . . 210
9.3.2 Monoclausal coordination . . . 211
9.3.3 Multiclausal coordination . . . 213
9.3.3.1 Multiclausal representation . . . 213
9.3.3.2 Biclausal representation . . . 215
9.3.4 Rules for conjuncts of particular type . . . 217
9.4 Summary . . . 218

10 Conclusion 219

A Summary in English 223

B Summary in Polish **227**

Bibliography **231**

Chapter 1

Introduction

1.1 Aim and scope

The aim of this work is to extend the knowledge about non-standard coordination phenomena such as coordination of unlike categories (as in (1.1)) and coordination of different grammatical functions (see (1.2)) by providing a formal analysis of rich attested Polish data, taken mainly from the National Corpus of Polish (*Narodowy Korpus Języka Polskiego*, NKJP; Przepiórkowski *et al.* 2012; `http://nkjp.pl/`).

(1.1) Owinął dziecko w koc i ręcznikiem.
wrapped baby in blanket.ACC and towel.INST
'He wrapped the baby in a blanket and in a towel.' (Kosek 1999, p. 43, ex. (8))

(1.2) Kogo i komu przedstawił?
who.ACC and who.DAT introduced
'Who did he introduce to whom?' (Kallas 1993, p. 121, ex. (241))

In (1.1) the first conjunct, *w koc*, is a prepositional phrase, while the other, *ręcznikiem*, is a nominal. In (1.2) *Kogo* is the direct object marked for the accusative case, while *komu* is the indirect object marked for the dative case.

This work supports the thesis – defended on different grounds in Przepiórkowski and Patejuk 2014 – that such phenomena are part of the system of Polish language (they are not typical of colloquial, careless utterances; they are not performance errors). The thesis of this dissertation is that unlike coordination in Polish – i.e. coordination of unlike categories and unlike grammatical functions – is part of the system of Polish in the sense that it may be formally described within a contemporary linguistic theory, namely in the framework of Lexical Functional Grammar (LFG; Bresnan 1982b, Dalrymple 2001). In doing so, this work takes a descriptive (rather than explanatory) approach to data discussed: it aims not to miss generalisations, but it does not seek any deep principles or parameters which make, say, coordination of unlike grammatical functions possible in Polish but not in English (at least, not to the same extent).

Under the coordination of unlike categories, particular conjuncts correspond to phrases which are different categorially, which is in stark contrast with the common assumption that only identical categories can be coordinated. This phenomenon has been discussed in the literature

– papers devoted to this issue include Sag *et al.* 1985 and Bayer 1996 and, in Polish, Świdziński 1992, 1993 and Kallas 1993, from which the examples presented below are taken:

(1.3) Jana dziwiło, [że Maria wybiera Piotra], i [jej brak gustu].
Jan.ACC puzzled.3.SG.N that Maria chooses Piotr and her lack.NOM.SG.M3 taste
'(The fact) that Maria prefers Piotr and her lack of taste puzzled Jan.'
(Świdziński 1992, 1993)

(1.4) Doradził mu [wyjazd] i [żeby nie wracał].
advised him departure.ACC and that NEG come back
'He advised him to leave and not to come back.' (Kallas 1993, p. 92, ex. (48a))

(1.5) Chcę pić i papierosa.
want drink.INF and cigarette.ACC
'I want to drink and (I want) a cigarette.' (Kallas 1993, p. 123, ex. (102))

In (1.3) a sentential clause with the complementiser ŻE is coordinated with a nominal, forming a coordinate subject. In (1.4) the direct object consists of a noun phrase coordinated with a sentential clause with the complementiser ŻEBY. In (1.5) the complement is an infinitival phrase (whose subject is controlled) coordinated with a nominal.

The phenomenon of unlike category coordination is fairly common; this becomes clear when inspecting Walenty, the new valence dictionary of Polish (Przepiórkowski *et al.* 2014b; `http://zil.ipipan.waw.pl/Walenty`; `http://walenty.ipipan.waw.pl`), which contains a number of schemata which allow such coordination, all attested in NKJP. The relative frequency of instances of such coordination suggests that this phenomenon is systematic. In spite of this, there seems to be no systematic analysis of this phenomenon for Polish and there is certainly no such analysis in the framework of LFG – the aim of this work is to fill this gap by providing a formal analysis couched in this framework.

Under the other kind of non-standard coordination, coordinated items of the same semantic type (interrogative words, as in the examples from the literature provided below, *n*-words, pronouns expressing universal quantification, -*kolwiek*-type pronouns, etc.) correspond to different grammatical functions (which need not necessarily be dependents of the same head).

(1.6) Kto i z kim gra?
who.NOM and with who.INST plays
'Who does play with whom?' (Bobrowski 1988, p. 148, ex. (40a))

(1.7) Kto, kiedy i dla kogo napisał te wiersze?
who.NOM when and for who.GEN wrote these poems
'Who wrote those poems, when and for whom?'
(Danielewiczowa 1996, p. 85, ex. (201))

(1.8) Kto komu kogo i kiedy przedstawiał?
who.NOM who.DAT who.ACC and when introduced
'Who did introduce who to whom and when?'
(Saloni and Świdziński 2001, p. 216, ex. (19e))

(1.9) Jakie i skąd zdobywał informacje?
what.ACC and from where obtained information.ACC
'What information and where from did he obtain?' (Kallas 1993, p. 141, ex. (108))

In (1.6) two arguments of the verb *gra* are coordinated: the first conjunct corresponds to the subject, while the second one is the prepositional complement. In (1.7) three dependents of *napisał* are coordinated: the subject, the temporal adjunct and the prepositional phrase expressing the beneficiary, which can be analysed either as an argument or an adjunct. In (1.8) – according to punctuation – only the last two interrogative items are coordinated: the direct object and the temporal adjunct of *przedstawiał*. Finally, the items coordinated in (1.9) depend on different heads: the first conjunct is a modifier of *informacje*, which is the direct object of *zdobywał*, while the second conjunct is a dependent (an ablative argument or an adjunct) of *zdobywał*. Together, these examples show that various grammatical functions may be coordinated, regardless of their argument/adjunct status, possibly even dependents of different heads, as long as they correspond to the same semantic type (interrogative items in the examples above).

Contrary to what was shown above, it is often assumed, tacitly or overtly, that the prerequisite for coordinating two or more items is that they correspond to the same grammatical function. An example of an overt expression of such a claim in the LFG literature may be found in Peterson 2004, p. 643: "Coordination is subject to the condition that items can be conjoined if and only if they satisfy the condition of functional equivalence." – to support this claim, the author argues that[1] "This condition does not have to be stipulated; it follows as an axiom from the general principles of functional application to sets." – as shown in ch. 5, not only is such a condition false for Polish (and many other languages), it also does not follow in any way from "the general principles of functional application to sets" as this is not the only way in which such coordination may be formalised.

A similar line of reasoning may be found in Saloni 2005, p. 45:[2]

> Example (16) is more difficult:
>
> (16) Kto, co i komu dał?
> who.NOM what.ACC and who.DAT gave
> 'Who gave what to whom?' (Saloni 2005, p. 45, ex. (16))
>
> Sentences constructed in this way occur in texts, though their structure is derailed from the perspective of the system of contemporary Polish. Two phrases taking different positions and having different semantic roles are conjoined here using a coordinating conjunction. A sentence built in accordance with syntactic rules is:
>
> (16') Kto co komu dał?
> who.NOM what.ACC who.DAT gave
> 'Who gave what to whom?' (Saloni 2005, p. 45, ex. (16'))

[1] Note that the rather unfortunate statement that "it follows as an axiom" is a fragment of the quotation.

[2] The relevant fragment was translated from Polish (together with the formatting of examples): *Trudniejszy jest przykład (16): [. . .]. Zdania tak zbudowane trafiają się w tekstach, choć struktura jest wykolejona z punktu widzenia systemu współczesnej polszczyzny. Spójnikiem współrzędnym są tu połączone frazy zajmujące różne pozycje i grające różne role semantyczne. Takich członów nie powinno się łączyć spójnikiem współrzędnym. Zdaniem zbudowanym zgodnie z regułami składniowymi jest (16').*

The claim that sentences involving such coordination are “derailed” was shown to be false in Przepiórkowski and Patejuk 2014, where various types of coordination of unlike grammatical functions were presented (i.e. involving conjuncts belonging to a range of semantic types, not only limited to interrogative items) and where it was demonstrated that the frequency of questions involving lexico-semantic coordination (considered to be “derailed” in Saloni 2005) can be higher than the frequency of the corresponding multiple *wh*-question (considered to be “in accordance with the syntactic rules” and “the system of contemporary Polish”).

With the notable exception of Kallas 1993, Polish works on lexico-semantic coordination seem to concentrate on interrogative items exclusively. Furthermore, no formalisation of this phenomenon was proposed in Polish literature – Kallas 1993 only provided a dependency-like representation for selected examples, though no explanation of how it was obtained was given. Again, this work aims to fill this gap by offering a formalised analysis of lexico-semantic coordination set in the framework of LFG.

1.2 Method(ology)

The aim set in § 1.1 is achieved through providing an LFG analysis based on attested data.

Lexical Functional Grammar was chosen as the theoretical framework of this work for a number of reasons. This formalism has been successfully put to the task of describing a wide range of phenomena in a number of typologically diverse languages. Moreover, LFG is a mature, stable formalism which is not only empirically adequate, but also theoretically sound – this property makes it possible to provide a very explicit linguistic analysis. Finally, theoretical LFG analyses may be tested with the help of XLE (Xerox Linguistic Environment; Crouch *et al.* 2011; `http://www2.parc.com/isl/groups/nltt/xle/`), a dedicated, very advanced platform for implementing LFG grammars.

The analysis of relevant phenomena in focus of this work is formalised in the framework of LFG as described in Dalrymple 2001, with modifications influenced by Alsina *et al.* 2005 (the discussion of the necessary repertoire of grammatical functions, crucial to the issue of coordination of unlike categories) and Maxwell and Manning 1996 (a very careful and explicit formalisation of non-constituent coordination, which inspired the early account of monoclausal lexico-semantic coordination presented in this work).

When it comes to the analysis of the general syntax of Polish, including phenomena such as agreement and case assignment, this work owes much to previous formal analyses of Polish, especially the HPSG (Head-driven Phrase Structure Grammar; Pollard and Sag 1987, 1994) analysis proposed in Przepiórkowski *et al.* 2002.

While this work is aware of previous analyses of the relevant coordination phenomena in other frameworks and sometimes draws from them, it is not a reimplementation nor modification of any existing analysis. It must be noted, however, that this work is based on a number of conference papers co-authored by Adam Przepiórkowski, the supervisor of this work.

As mentioned above, the proposed analysis is based on examples extracted from NKJP and, less often, found on the Internet – such choice of data was made in an attempt to discuss real problems stemming from real data and avoid practising armchair linguistics. While the discussion is based on authentic data whenever possible, sometimes examples from the literature or constructed examples based on either of previous example types are used, for instance in order to provide minimal pairs or so as to make it easier to compare various structures.

Finally, the analysis proposed in this work is not an isolated fragment (which is typical of article-length works) – it was implemented in XLE as a part of POLFIE, a large scale LFG grammar of Polish (Patejuk and Przepiórkowski 2012c). Such a design makes it possible to verify the proposed analysis, taking interactions with various phenomena into account – these include especially agreement, structural case assignment and control. Interactions with these phenomena are also addressed in this work and their formalisation is provided.

1.3 Overview of content

Let us briefly describe how this work is structured: it is organised into 3 parts.

The first part, assuming no previous knowledge of LFG, introduces necessary basics of this formalism (ch. 2) and provides some information about selected phenomena of Polish syntax which are important in the following discussion (ch. 3), including subject-verb agreement and structural case assignment.

The second, main, part discusses rich data related to the two selected non-standard coordination phenomena which are the focus of this work and provides formal LFG analyses of coordination of unlike grammatical categories (ch. 4) and coordination of different grammatical functions (ch. 5).

The third, last, part describes the implementation of the theoretical analyses presented earlier in this work. Some general information about how the implemented grammar is organised is provided in ch. 6; ch. 7 explains the correspondence between theoretical LFG analyses and the implementation on the basis of phenomena presented in ch. 3; ch. 8 is concerned with how Walenty, a valence dictionary of Polish providing an explicit account of unlike category coordination, is converted in order to be used by an implemented LFG grammar of Polish; finally, ch. 9 describes how theoretical analyses presented in the second part (ch. 4 and ch. 5) are implemented.

Part I

Basics: adopting existing solutions

Chapter 2

LFG basics

In order to familiarise the reader with the conventions and representations used throughout this work, this chapter provides a brief introduction to LFG by presenting some key concepts of this formalism and showing basic devices employed in this theory.

2.1 Basics: lexicon and rules

Roughly speaking, LFG grammars consist of two parts: a lexicon and syntactic rules.

Lexical entries stored in the lexicon introduce f(unctional)-descriptions – this is a general term used for all kinds of constraints: simple (consisting of a single statement) and complex (a set of simple statements), existential, defining and constraining, positive and negative.

Rules are used for combining words into larger units, namely phrases. Furthermore, the annotation of rules (using f-descriptions) serves as an instruction on how fragments of functional structure corresponding to particular constituent structure elements should be combined to yield functional structures corresponding to larger parts of the utterance.

As a result of the interaction of annotated rules and lexical entries, two levels of structure arise, providing a representation in terms of phrasal categories and grammatical functions, c(onstituent)- and f(unctional)-structure, respectively.

Let us present some most fundamental LFG devices on the basis of a very simple sentence:

(2.1) Facet idzie.
guy.NOM.SG.M1 walk.3.SG
'A/the guy walks.'

Example (2.1) consists of only two words: an intransitive verb, which is the main predicate, and a noun, which serves as the subject. Lexical entries of these words are provided below:

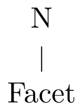
(2.2) facet N (↑ PRED)='GUY'
(↑ CASE)= NOM
(↑ GEND)= M1
(↑ NUM)= SG
(↑ PERS)= 3

(2.3) idzie V (↑ PRED)='WALK<(↑ SUBJ)>'
(↑ SUBJ CASE)$=_c$ NOM
(↑ SUBJ NUM)$=_c$ SG
(↑ SUBJ PERS)$=_c$ 3
(↑ TENSE)= PRES

Lexical entries consist of three elements: the orthographic form (*facet* 'guy' in (2.2)), a label providing part of speech information of the preterminal node dominating the leaf node (N in (2.2) stands for a noun) and f-descriptions, which constitute the rest of the entry. Let us investigate stepwise the f-descriptions provided in (2.2) and (2.3).

The '$\uparrow$' symbol is a metavariable – it corresponds to the f-structure of the mother c-structure category. In (2.2) it refers to N, the preterminal node dominating the leaf node *facet* – see the c-structure fragment in (2.4). The PRED attribute in the first line of provided sample lexical entries corresponds to the semantic form of the given predicate:[1] in (2.2) it is a zero-place predicate (it takes no arguments: 'GUY', see the f-structure in (2.5)), while in (2.3) the predicate WALK takes one argument (it requires a subject: 'WALK<($\uparrow$ SUBJ)>', as explained below).

(2.4)



(2.5)

$$\begin{bmatrix} \text{PRED} & \text{`GUY'} \\ \text{CASE} & \text{NOM} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{SG} \\ \text{PERS} & 3 \end{bmatrix}$$

Let us consider other aspects of f-descriptions: the value of the PRED attribute is set using defining equations (these use the equality symbol, '=') – such equations introduce the relevant attribute-value pair to the relevant f-structure.[2] (2.2) uses defining equations to specify a range of agreement features of the noun in the f-structure in (2.5): it is nominative (NOM), masculine (M1 stands for human masculine[3] gender in Polish), singular (SG) and third person (3).[4]

While the lexical entry in (2.2) uses only defining equations, (2.3) contains some constraining equations, which use the equality symbol with a subscript: '$=_c$'. Unlike defining equations, these do not introduce attribute-value pairs into the f-structure, but instead they check whether such a pair has already been introduced elsewhere in the grammar. For example, ($\uparrow$ SUBJ CASE)$=_c$ NOM, the first constraining equation in (2.3), requires that the verb's subject must bear the nominative case – it will not mark the subject for the nominative case, it only checks that such a specification already exists in the relevant f-structure.

F-descriptions in (2.3) use different paths to reach various fragments of the f-structure. The shortest paths consist of a metavariable and an attribute: ($\uparrow$ TENSE) points to the TENSE attribute of the verb, so the defining equation ($\uparrow$ TENSE)= PRES specifies the verb's tense as present. There are, however, more complex paths: ($\uparrow$ SUBJ NUM) points to the number attribute of the subject of the verb, so the constraining equation ($\uparrow$ SUBJ NUM)$=_c$ SG requires that the subject of this verb be specified for singular number (more formally: that the SUBJ attribute of the verb contains the attribute NUM whose value is SG).

This in turn shows that attributes may have various values. Some attributes have atomic values – these include morphosyntactic attributes such as case (CASE), number (NUM), gender

[1] Note that the value of PRED is provided in English in order to make it possible to understand the f-structures without returning to glosses.

[2] The PRED attribute is special in that it is an instantiated feature – its value can be set only once.

[3] See Przepiórkowski 2004a for discussion of the 5-gender system of Mańczak 1956 adopted in the IPI PAN Corpus and subsequently in NKJP (Przepiórkowski 2009).

[4] Nouns bear third person by default to simplify agreement mechanisms, though this is not necessary as the latter could be reformulated so as to handle agreement with nouns without this attribute.

(GEND), etc. There exist, however, attributes which take entire f-structures as their values. These include grammatical functions such as SUBJ – the value of this attribute is filled with the entire f-structure which corresponds to the subject.

F-structures constructed by the lexical entries given in (2.2) and (2.3) are provided in (2.5) above and (2.6) below, respectively. Note that these f-structures contain only f-descriptions introduced by defining equations – constraining equations used in (2.3), although effective, are normally not represented, see (2.6) below:

(2.6)
$$\begin{bmatrix} \textsc{pred} & \text{`}\textsc{walk}\langle\boxed{1}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\begin{bmatrix} \quad \end{bmatrix} \\ \textsc{tense} & \textsc{pres} \end{bmatrix}$$

A note on the representation: in (2.6) there is an index,[5] a boxed number ($\boxed{1}$), which corresponds to the f-structure of the subject (SUBJ), which is empty here, but see (2.9). This index is also placed in the PRED value of the verb – it is the same object, but its value is shown only in one place (next to SUBJ), elsewhere only an index is used for reference.

Let us now take a look at an annotated c-structure rule which makes it possible to build a sentence and its corresponding f-structure:

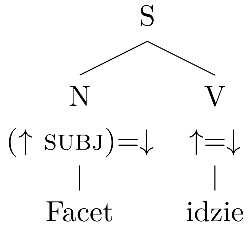
(2.7)
$$\begin{array}{ccc} \text{S} \quad \rightarrow & \text{N} & \text{V} \\ & (\uparrow \textsc{subj})=\downarrow & \uparrow=\downarrow \end{array}$$

The left-hand side (to the left of '→', the rewrite symbol) of the rule in (2.7), S, is the mother category, while its children are given in the right-hand side. Only right-hand side elements may bear functional annotation – it is attached below the category to which it applies. The '↑' metavariable in (2.7) points to the f-structure of the mother category, S. (2.7) contains a new metavariable, '↓' – it points to the f-structure of the category to which it is attached. As a result, the annotation attached to N, (↑ SUBJ)=↓, means that N ('↓') is the subject of the mother category ((↑ SUBJ)) – the f-structure of N is the value of the SUBJ attribute of the mother.

Let us now consider the annotation of the other right-hand side category, the verb: ↑=↓. It is the annotation which is characteristic of heads in LFG – it has the effect of unifying the f-structure of the mother and the f-structure of the node to which this annotation is attached.

Now it is possible to proceed to the interaction of lexical entries in (2.2) and (2.3) (for their partial f-structures, see (2.5) and (2.6), respectively) and the annotated rule in (2.7):

(2.8)



(2.9)
$$\begin{bmatrix} \textsc{pred} & \text{`}\textsc{walk}\langle\boxed{1}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\begin{bmatrix} \textsc{pred} & \text{`}\textsc{guy}\text{'} \\ \textsc{case} & \textsc{nom} \\ \textsc{gend} & \textsc{m}1 \\ \textsc{num} & \textsc{sg} \\ \textsc{pers} & 3 \end{bmatrix} \\ \textsc{tense} & \textsc{pres} \end{bmatrix}$$

[5] While HPSG uses indices such as in (2.6), the convention traditionally used in theoretical LFG was to use arrows linking relevant parts of the f-structure. Since this is only a representational issue (note that XLE uses indices in f-structure display), this work adheres to the HPSG convention for the sake of clarity of representation – the f-structures in § 5.7 would be considerably harder to read if arrows were used instead of indices.

The verb, *idzie* ‘walks’, is annotated as the head, so (2.6), the f-structure constructed by its lexical entry (see (2.3)), contributes to the f-structure of the mother, S. According to the information provided in its PRED attribute, the verb requires an argument – a subject. The annotation in (2.7) assigns the subject grammatical function to the noun, so its f-structure, (2.5), is placed as the value of the SUBJ attribute of the verb.

Moreover, using constraining equations defined in its lexical entry (see (2.3)), the verb checks that its subject bears an appropriate agreement specification (nominative case, singular number, third person). The verb and the noun agree and the f-structure provided in (2.9) results.

2.2 F-structure representation

LFG uses multiple levels of representation, c-(onstituent) structure and f-(unctional) structure being the two syntactic levels. Unlike in configurational theories of syntax, in LFG grammatical functions are not determined by c-structure positions – some element is a subject because it was annotated as such using f-descriptions (as in (2.7)); it does not have to be a specifier of some category.

C-structure provides a representation in terms of phrasal categories; it is produced by rules with functional annotation, such as (2.7). It is appropriate for encoding surface phenomena such as word order and therefore it is claimed to be subject to much variation across languages.

By contrast, the other level of representation, namely f-structure, is a product of the interaction of functional descriptions placed in lexical entries and attached to c-structure rules. Fragments of such annotation contribute to the creation of attribute-value matrices which provide a representation in terms of grammatical functions. Some element may consist of two different categories at the level of c-structure, but it may still correspond to the same grammatical function at the level of f-structure – this is the case of the coordination of unlikes, discussed in ch. 4. While c-structure provides information about surface-level phenomena such as word order, f-structure provides a deeper, more universal represenation – even though various languages (or even the same language, as in the case of free word order languages) may use different word orders for a given utterance, the corresponding f-structures may still be identical.

2.2.1 Grammatical functions

Although there is a rich body of literature discussing grammatical functions in LFG, only the most basic aspects of this topic are discussed here.

Grammatical functions are primitives of the LFG theory, as Dalrymple 2001, pp. 3–4 puts it: “abstract grammatical functions like subject and object are not defined in terms of phrase structure configurations or of semantic or argument structure relations”, “they are not derived or defined in terms of other linguistic notions such as agenthood or phrasal configuration”.

LFG defines a universally available set of grammatical functions (though this does not mean that every language must use all grammatical functions). The most basic classification divides grammatical functions into governed (or governable) ones and modifiers. The first class includes argument grammatical functions such as SUBJ (subject), OBJ (primary object), OBJ_θ (secondary object), OBL (oblique object), COMP (closed complement, usually sentential) and XCOMP[6] (open complement, infinitival or predicative). Modifiers include ADJ (closed adjunct, often represented as ADJUNCT in implemented grammars) and sometimes XADJ (open adjunct).

Argument grammatical functions may be further divided into core and noncore arguments on the basis of the variation in their behaviour with respect to various phenomena (such as agreement, binding and control, for instance). The former, also known as terms, include the subject (SUBJ) and object grammatical functions (OBJ, OBJ_θ). The remaining arguments belong to the class of obliques (or nonterms): OBL, COMP, XCOMP.

Furthermore, there is the distinction between closed and open grammatical functions. The grammatical functions XCOMP and XADJ are classified as open grammatical functions because they share a common characteristic: although they subcategorise for a subject, it is not realised clause-internally and therefore it needs to be specified externally – their subject is controlled by an argument from another phrase. By contrast, the COMP grammatical function has its own, independent subject, while ADJ typically takes no SUBJ at all.

The following four sentences are constructed so as to show the entire repertoire of grammatical functions discussed above. Let us start with two simple examples:

(2.10) Eryk czeka na Antka.
Eryk waits for Antek
'Eryk is waiting for Antek.'

(2.11) Antek dał Erykowi ciekawą płytę.
Antek gave Eryk interesting record
'Antek gave Eryk an interesting record.'

(2.12)
$$\begin{bmatrix} \textsc{pred} & \text{`}\textsc{wait}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\begin{bmatrix}\textsc{pred} & \text{`}\textsc{Eryk}\text{'}\end{bmatrix} \\ \textsc{obl} & \boxed{2}\begin{bmatrix}\textsc{pred} & \text{`}\textsc{Antek}\text{'}\end{bmatrix}\end{bmatrix}$$

(2.13)
$$\begin{bmatrix} \textsc{pred} & \text{`}\textsc{give}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\begin{bmatrix}\textsc{pred} & \text{`}\textsc{Antek}\text{'}\end{bmatrix} \\ \textsc{obj} & \boxed{2}\begin{bmatrix}\textsc{pred} & \text{`}\textsc{record}\text{'} \\ \textsc{adj} & \left\{\begin{bmatrix}\textsc{pred} & \text{`}\textsc{interesting}\text{'}\end{bmatrix}\right\}\end{bmatrix} \\ \textsc{obj}_\theta & \boxed{3}\begin{bmatrix}\textsc{pred} & \text{`}\textsc{Eryk}\text{'}\end{bmatrix}\end{bmatrix}$$

In (2.10) the verb CZEKAĆ 'wait' takes two arguments: a subject (SUBJ) and an oblique object (OBL) – this is represented in the f-structure provided in (2.12).[7] The verb DAĆ 'give' in (2.11) requires, apart from a subject, two objects: a direct (OBJ) and an indirect one (OBJ_θ), as in the f-structure provided in (2.13). Moreover, the f-structure of the direct object contains an adjunct.

The following example focuses on open grammatical functions:

(2.14) Wychodząc, Marysia kazała Erykowi czekać na Antka.
leaving Marysia ordered Eryk wait for Antek
'Leaving, Marysia ordered Eryk to wait for Antek.'

[6] Implemented LFG grammars often use XCOMP grammatical function exclusively for infinitival complements, while predicative complements are assigned the XCOMP-PRED grammatical function.

[7] See § 2.3 for discussion of how prepositional phrases are analysed in LFG.

(2.15)
$$\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{order}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\
\textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Marysia}\text{'}\end{array}\right] \\
\textsc{obj}_\theta & \boxed{2}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Eryk}\text{'}\end{array}\right] \\
\textsc{xcomp} & \boxed{3}\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{wait}\langle\boxed{2},\boxed{4}\rangle\text{'} \\
\textsc{subj} & \boxed{2} \\
\textsc{obl} & \boxed{4}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Antek}\text{'}\end{array}\right]
\end{array}\right] \\
\textsc{xadj} & \left\{\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{leave}\langle\boxed{1}\rangle\text{'} \\
\textsc{subj} & \boxed{1}
\end{array}\right]\right\}
\end{array}\right]$$

The main verb in (2.14) is a control verb taking a subject, an indirect object ($\textsc{obj}_\theta$) and an open complement (XCOMP). As indicated using relevant indices in the f-structure in (2.15), the indirect object ($\textsc{obj}_\theta$, $\boxed{2}$) of the control verb serves as the subject of its open complement (XCOMP) – the predicate of the infinitival complement is the same as the main verb in (2.10), but its subject is structure-shared with an argument from another, higher clause. Furthermore, there is another open grammatical function: the participial modifier of the main verb is represented as its open adjunct (XADJ) – its subject is controlled by the subject of the main verb ($\boxed{1}$).

The last example features COMP, the closed complement:

(2.16) Zosia powiedziała Marysi, że Antek dał Erykowi ciekawą płytę.
Zosia told Marysia that Antek gave Eryk interesting record
'Zosia told Marysia that Antek gave Eryk an interesting record.'

(2.17)
$$\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{tell}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\
\textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Zosia}\text{'}\end{array}\right] \\
\textsc{obj}_\theta & \boxed{2}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Marysia}\text{'}\end{array}\right] \\
\textsc{comp} & \boxed{3}\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{give}\langle\boxed{4},\boxed{5},\boxed{6}\rangle\text{'} \\
\textsc{subj} & \boxed{4}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Antek}\text{'}\end{array}\right] \\
\textsc{obj} & \boxed{5}\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{record}\text{'} \\
\textsc{adj} & \left\{\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{interesting}\text{'}\end{array}\right]\right\}
\end{array}\right] \\
\textsc{obj}_\theta & \boxed{6}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{Eryk}\text{'}\end{array}\right] \\
\textsc{comp-form} & \textsc{że}
\end{array}\right]
\end{array}\right]$$

The verb which is the main predicate in (2.11) serves as the clausal complement (COMP) of the main verb in (2.16), see the corresponding f-structure in (2.17). Unlike XCOMP, the open complement whose subject needs to be controlled, COMP is a closed complement which has its own, independent subject – compare the f-structures provided in (2.15) and (2.17), respectively.

2.2.2 F-structure wellformedness

As discussed in Dalrymple 2001, f-structures must adhere to three rules which ensure that subcategorisation requirements (if any) are satisfied and that there are no conflicting values of attributes.

The completeness condition requires that all arguments (governable grammatical functions, i.e. those mentioned in PRED) of each predicate are present at f-structure (they have a value). For example: (2.6), the partial f-structure built by (2.3), the lexical entry of the verb *idzie* 'walks', is incomplete because SUBJ is mentioned in PRED but the f-structure in (2.6) lacks the value of SUBJ. The completeness condition is satisfied when this argument slot is filled: either using a lexical subject, as in (2.9), or using an implicit argument which is represented at f-structure (with 'PRO' as the value of the PRED attribute of the relevant grammatical function) but has no c-structure counterpart (for an example, see (2.51) and the corresponding structures provided in (2.53) and (2.55)).

The coherence condition prohibits f-structures containing arguments which are not required by a given predicate – these are arguments which are not listed in its semantic form (the value of the PRED attribute). For instance: the lexical entry provided in (2.3) specifies *idzie* as an intransitive verb – it takes only one argument, the subject. While (2.9) satisfies this condition, an f-structure with an extra argument would violate it. The f-structure representation of (2.18) given in (2.19)[8] is incoherent because it contains a complement clause[9] which is not subcategorised according to the lexical entry of the verb (see (2.3)):

(2.18) *Facet idzie, iż kot śpi.
guy walks that cat sleeps
'A/the guy walks that a/the cat sleeps.' (intended)

(2.19)
$$*\left[\begin{array}{ll}
\textsc{pred} & \text{`WALK}\langle\boxed{1}\rangle\text{'} \\
\textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{`GUY'}\end{array}\right] \\
\textsc{comp} & \boxed{2}\left[\begin{array}{ll}
\textsc{pred} & \text{`SLEEP}\langle\boxed{3}\rangle\text{'} \\
\textsc{subj} & \boxed{3}\left[\begin{array}{ll}\textsc{pred} & \text{`CAT'}\end{array}\right] \\
\textsc{tense} & \textsc{pres} \\
\textsc{comp-form} & \text{ŻE}
\end{array}\right] \\
\textsc{tense} & \textsc{pres}
\end{array}\right]$$

The last condition, consistency (or uniqueness) condition, makes sure that attributes have no conflicting values – an attribute may have only one value. For example: the subject in (2.9) bears the nominative case (according to the specification in its lexical entry, see (2.2)), so assigning it the accusative case ((↑ SUBJ CASE)= ACC) would cause a violation of the consistency condition.[10]

2.2.3 More about notation

Let us add some more elements to the repertoire of LFG devices that have been introduced.

Constraints presented so far were exclusively positive – they introduced some value (defining equations) or required it to be already present in some f-structure (constraining equations). LFG makes it possible to use negation to create negative constraints.

[8] The f-structure in (2.19) is simplified – the agreement features of subjects are skipped as irrelevant.

[9] The complementiser form used in (2.18) is *iż*, but it is represented in (2.19) as the ŻE value of the COMP-FORM attribute. This is because, following Świdziński 1992, complementisers are classified into types, whereby *iż* belongs to ŻE-type complementisers.

[10] There are, however, situations when a given form has an ambiguous specification for some feature – see § 3.2.3.3 for discussion of how case syncretism can be modelled in LFG.

To account for the agreement pattern shown in (2.20), where the subject of the past tense verb *szły* 'walked' may be specified for any gender except human masculine, it is possible to use the negative constraint defined in (2.21) instead of 4 separate lexical entries (each with a different positive gender requirement such as (↑ SUBJ GEND)$=_c$ F).

(2.20) Kobiety/dzieci/koty/roboty/*faceci szły.
woman/child/cat/robot/guy.NOM.PL.F/N/M2/M3/M1 walks
'Women/children/cats/robots/guys walk.'

(2.21) (↑ SUBJ GEND)$\neq$ M1

(2.22) szły V (↑ PRED)='WALK<(↑ SUBJ)>'
(↑ SUBJ CASE)$=_c$ NOM
AGR-GEND
(↑ SUBJ NUM)$=_c$ PL
(↑ SUBJ PERS)$=_c$ 3
(↑ TENSE)= PAST

(2.23) AGR-GEND ≡ (↑ SUBJ GEND)$\neq$ M1

It is worth noting that (2.22), the lexical entry of *szły* used in (2.20), makes use of another LFG notational device, namely a call to a template – AGR-GEND. Templates can be seen as a notational shortcut[11] – once a template is defined, as in (2.23), where the template AGR-GEND introduces the constraint in (2.21), there is no need to put the same constraint in every relevant lexical entry – it is enough to make a call to the relevant template. The use of templates makes it possible to state constraints in an economic, clear and consistent way, which is particularly important with more complex constraints (see (3.29) in § 3.1.2.2 for an example).

An alternative to the solutions proposed above would be to use one lexical entry with a four-way disjunction – see (2.24), where the template AGR-GEND is redefined so as to use a disjunction, which is represented using the '∨' symbol (in LFG literature sometimes '|', the pipe symbol, is used instead, as in (2.61)). This is how the disjunctive gender constraint could be formalised:

(2.24) AGR-GEND ≡ (↑ SUBJ GEND)$=_c$ F ∨ (↑ SUBJ GEND)$=_c$ N ∨
(↑ SUBJ GEND)$=_c$ M2 ∨ (↑ SUBJ GEND)$=_c$ M3

If the definition of the template AGR-GEND called in the lexical entry in (2.22) uses the disjunctive gender constraint in (2.24) instead of the negative gender constraint in (2.23), it also successfully accounts for the agreement facts in (2.20).

An alternative formalisation of the constraint in (2.24) is provided in (2.25):

(2.25) AGR-GEND ≡ (↑ SUBJ GEND)$\in_c$ {F, N, M2, M3}

It requires that the value of gender of the subject must belong to the 4-element set which contains the relevant values of gender – while the set membership operator '∈' adds an element to a set, '$\in_c$' is its constraining counterpart: it checks that an element belongs to a given set.

[11] Templates can also be used to express deeper linguistic generalisations, see Asudeh *et al.* 2013.

Another logical operator used in LFG for creating complex statements, conjunction, has not been discussed so far even though it is used extensively in the lexical entries. Unlike in the case of disjunction, conjunction is usually left implicit in LFG – if there is no operator linking statements, these are assumed to be conjoined. Sometimes, however, conjunction will be signalled explicitly in this work (using the '$\wedge$' symbol) in order to avoid confusion when discussing more complex statements.

Apart from constraints which use attribute-value pairs (be they defining or constraining), there are existential constraints – these are satisfied if a certain attribute is present in the f-structure, regardless of its value. Such constraints may also be negated; in such a situation the relevant attribute must not be present in the f-structure for the constraint to be satisfied. Two examples of existential constraints, positive and negative, are provided below:

(2.26) (↑ SUBJ CASE)

(2.27) ¬(↑ XCOMP TENSE)

The constraint in (2.26) requires case marking to be present on the subject, i.e. the f-structure of the subject must contain the CASE attribute – this is a positive existential constraint. By contrast, (2.27) is a negative existential constraint – it ensures that the open infinitival complement (represented as the XCOMP grammatical function) is not specified for tense, i.e. that there is no TENSE attribute in this path.

All constraints use paths but so far examples used the outside-in variety exclusively: such constraints are interpreted from left to right and they build f-structure in a top-down manner, using embedding (the leftmost attribute contains the attribute which follows it, and so on):

(2.28) (↑ OBJ CASE)= DAT

(2.29) $\begin{bmatrix} \text{OBJ} & \begin{bmatrix} \text{CASE} & \text{DAT} \end{bmatrix} \end{bmatrix}$

With the other variety of paths, namely inside-out paths, the order of functional application is reversed: these paths are interpreted from right to left and they build f-structure using a bottom-up strategy. Inside-out paths start from a certain f-structure and, instead of embedding, they build f-structures which contain this f-structure. Let us assume that the constraint provided in (2.30) is added to the lexical entry defined in (2.2), as in (2.31):

(2.30) (SUBJ ↑)=↓

(2.31) facet N (↑ PRED)='GUY'
(↑ CASE)= NOM
(↑ GEND)= M1
(↑ NUM)= SG
(↑ PERS)= 3
(SUBJ ↑)=↓

(2.32)
$$\begin{bmatrix} \text{SUBJ} & \begin{bmatrix} \text{PRED} & \text{'GUY'} \\ \text{CASE} & \text{NOM} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{SG} \\ \text{PERS} & 3 \end{bmatrix} \end{bmatrix}$$

As shown in (2.32), when placed in the lexical entry of the noun (see (2.31)), (2.30) builds the SUBJ f-structure on the top of the f-structure specified by the original lexical entry (see (2.5)) – ↑ points to the mother category, while ↓ points to the current category.

It is worth mentioning that inside-out and outside-in paths may be combined: instead of using a metavariable as the beginning of an outside-in path, an inside-out one could be used to specify the starting point for the outside-in path. Agreement between the adjectival modifier and its head may be handled in this way:

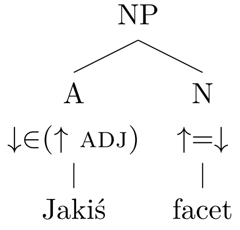
(2.33) Jakiś facet idzie.
some.NOM.SG.M1 guy.NOM.SG.M1 walks
‘Some guy walks.’

(2.34) jakiś A $(\uparrow \textsc{pred})=\text{‘}\textsc{some}\text{’}$
$((\textsc{adj} \in \uparrow)\ \textsc{case}) =_c \textsc{nom}$
$((\textsc{adj} \in \uparrow)\ \textsc{gend}) =_c \textsc{m}1$
$((\textsc{adj} \in \uparrow)\ \textsc{num}) =_c \textsc{sg}$

(2.35) $$\begin{array}{lccc} \text{NP} & \rightarrow & \text{A} & \text{N} \\ & & \downarrow\in(\uparrow \textsc{adj}) & \uparrow=\downarrow \end{array}$$

According to the rule provided in (2.35), while the noun (N) is the head of the noun phrase (NP), the adjective (A) is added to its adjunct set (ADJ). The following structures result:

(2.36)



(2.37) $$\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{guy}\text{’} \\ \textsc{adj} & \left\{ \begin{bmatrix} \textsc{pred} & \text{‘}\textsc{some}\text{’} \end{bmatrix} \right\} \\ \textsc{case} & \textsc{nom} \\ \textsc{gend} & \textsc{m}1 \\ \textsc{num} & \textsc{sg} \\ \textsc{pers} & 3 \end{bmatrix}$$

The lexical entry of *jakiś* ‘some’, see (2.34), contains three constraining equations which use a combination of inside-out and outside-in paths to ensure appropriate agreement between the modifier and the head. First, an inside-out path is used to point to the f-structure of the head noun ((ADJ ∈ ↑)): it points from the embedded f-structure of the modifier to the f-structure which contains the ADJ attribute. Next, this inside-out path serves as a part of an outside-in path, pointing to relevant attributes of the noun (CASE, GEND, NUM) and checking for appropriate values (NOM, M1 and SG, respectively). This is how noun-adjective agreement can be handled directly in the lexicon (rather than by matching relevant values in syntax).[12]

[12] It must be mentioned, however, that the latter method is given preference in the LFG grammar of Polish – adjectives have their own agreement features and they are only matched in syntax with the relevant values of the head. Let us present two arguments supporting the use of this analysis.

First, there are pronouns (such as CO ‘what’, NIC ‘nothing’ and COKOLWIEK ‘whatever’) which in certains forms (nominative, accusative and vocative) require the modifier to appear in the non-agreeing case, namely genitive, as in (i), where *coś* is marked for the accusative case, while its modifier, *ciekawego*, is genitive:

(i) pokażę ci coś ciekawego
show.1.SG you.DAT something.ACC.SG.N interesting.GEN.SG.N
‘I will show you something interesting.’ (NKJP)

To handle such facts, agreement statements placed in lexical entries of adjectives such as (2.34) would have to check the lemma of the head and then require agreement or non-agreement.

Secondly, it is possible in Polish to use adjectives on their own – without a word that could be modified by the adjective, as in (ii), where *Najbrzydsza* is a superlative adjectival form without a nominal head:

2.3 Prepositional phrases

Specific uses of prepositions can be classified as semantic and non-semantic. While the former, semantic prepositions, have well-defined meaning and they contribute to the semantics of the utterance (as in (2.38), where the prepositional phrase has locative semantics), the latter, non-semantic prepositions, have no semantic contribution – the predicate selecting for such a preposition arbitrarily requires a certain form (see (2.39) for an example). Instead of carrying semantics, non-semantic prepositions only serve as a means of connecting a nominal as a dependent to some head – they can be thought of as an alternative to overt case marking on a noun (though many languages, including Polish, use both prepositions and case-marked nominals).

(2.38) Książka leżała na stoliku.
book lay on table.LOC
'A/the book was lying on the table.' (NKJP)

(2.39) Pół godziny musiała czekać na stolik.
half hour had wait for table.ACC
'She had to wait half an hour for a table (to be free).' (NKJP)

In LFG the difference between semantic and non-semantic prepositional phrases is visible in their f-structure representation. Semantic prepositions are treated as heads – they have their own PRED attribute whose value corresponds to the lemma of the preposition. The f-structure provided in (2.43), built using the rule in (2.40), corresponds to the prepositional phrase *na stoliku* used in (2.38), where *na* is a semantic preposition which takes a nominal object marked for the locative case: the f-structure of the semantic preposition is provided in (2.42), while (2.41) is the f-structure of the nominal.

(2.40)
$$\begin{array}{lllll} \text{PP} & \rightarrow & \text{P} & & \text{NP} \\ & & \uparrow=\downarrow & & (\uparrow \text{ OBJ})=\downarrow \end{array}$$

(2.41)
$$\begin{bmatrix} \text{PRED} & \text{`TABLE'} \\ \text{CASE} & \text{LOC} \end{bmatrix}$$

(2.42)
$$\begin{bmatrix} \text{PRED} & \text{`ON}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1} \\ \text{PTYPE} & \text{SEM} \end{bmatrix}$$

(2.43)
$$\begin{bmatrix} \text{PRED} & \text{`ON}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1}\begin{bmatrix} \text{PRED} & \text{`TABLE'} \\ \text{CASE} & \text{LOC} \end{bmatrix} \\ \text{PTYPE} & \text{SEM} \end{bmatrix}$$

By contrast, non-semantic prepositions such as in (2.39) are treated as co-heads (see the rule in (2.44), where both right-hand side elements bear the head annotation: ↑=↓) which do not contribute a PRED attribute of their own (see the lexical entry of *na* in (2.46)) – the value of this attribute is provided by the semantic head of the prepositional phrase, namely by the nominal (see (2.45)). As a result, it is the nominal which contributes the semantic form, not the

(ii) Najbrzydsza z nich podarowała mu małą Alfę Romeo.
ugly.SUP of they gave he.DAT small Alfa Romeo
'The ugliest of them gave him a small Alfa Romeo.' (NKJP)

In such a situation, if the analysis where the adjective does not have its own agreement features but only requires certain agreement with the head, one would be forced to introduce a pro head, thereby assuming an analysis with an implicit head, which seems rather unmotivated.

preposition. However, the preposition makes a contribution to the f-structure of the prepositional phrase – there is a dedicated attribute which hosts the form (lemma) of the preposition, namely PFORM, which is necessary to ensure that valence requirements of the predicate taking a non-semantic prepositional phrase are met (that the required preposition is used). Let us take a look at relevant f-structures:

(2.44)
$$\begin{array}{cccc} \text{PP} & \rightarrow & \text{P} & \text{NP}\\ & & \uparrow=\downarrow & \uparrow=\downarrow \end{array}$$

(2.45)
$$\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{table}\text{'} \\ \textsc{case} & \textsc{acc} \end{array}\right]$$

(2.46)
$$\left[\begin{array}{ll} \textsc{pform} & \textsc{for} \\ \textsc{ptype} & \textsc{nosem} \end{array}\right]$$

(2.47)
$$\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{table}\text{'} \\ \textsc{case} & \textsc{acc} \\ \textsc{pform} & \textsc{for} \\ \textsc{ptype} & \textsc{nosem} \end{array}\right]$$

(2.47), built using the rule in (2.44), provides a representation of the prepositional phrase *na stolik* used in (2.39), where the nominal is marked for the accusative case and the preposition is non-semantic. While the f-structure of the nominal *stolik* in (2.45) differs only in case from the f-structure of *stoliku* provided in (2.41), the f-structure of the preposition is different (compare non-semantic (2.46) with semantic (2.42)): apart from the difference in PRED values explained above ((2.46) has no PRED, while (2.42) does), note that the former has NOSEM as the value of PTYPE attribute (preposition type), while the latter is marked as SEM.[13]

The final c-structure rule for handling prepositional phrases is provided in (2.48). It uses a disjunctive annotation of NP: when the first disjunct is used, a semantic prepositional phrase results; when the second disjunct is used, the resulting prepositional phrase is non-semantic:

(2.48)
$$\begin{array}{cccc} \text{PP} & \rightarrow & \text{P} & \text{NP}\\ & & \uparrow=\downarrow & [(\uparrow \textsc{obj})=\downarrow \vee \uparrow=\downarrow] \end{array}$$

The f-structures below provide a representation of sentence (2.38) and a simplified version of (2.39) (*Czekała na stolik* 'She waited for a table (to be free).'):

(2.49)
$$\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{lay}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{book}\text{'} \end{array}\right] \\ \textsc{obl-locat} & \boxed{2}\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{on}\langle\boxed{3}\rangle\text{'} \\ \textsc{obj} & \boxed{3}\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{table}\text{'} \\ \textsc{case} & \textsc{loc} \end{array}\right] \\ \textsc{ptype} & \textsc{sem} \end{array}\right] \\ \textsc{tense} & \textsc{past} \end{array}\right]$$

(2.50)
$$\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{wait}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{pro}\text{'} \end{array}\right] \\ \textsc{obl} & \boxed{2}\left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{table}\text{'} \\ \textsc{case} & \textsc{acc} \\ \textsc{pform} & \textsc{for} \\ \textsc{ptype} & \textsc{nosem} \end{array}\right] \\ \textsc{tense} & \textsc{past} \end{array}\right]$$

Note that one of the arguments of the predicate LAY in (2.49) is OBL-LOCAT, which was not introduced in § 2.2.1 when discussing grammatical functions. For obliques with specific semantics, an index corresponding to its semantics may be added, which results in grammatical functions such as OBL-LOCAT (locative), OBL-ADL (adlative), OBL-ABL (ablative) and so on.

[13] Values of PTYPE attribute, SEM and NOSEM, come from the common feature declaration used in the ParGram project (Butt *et al.* 2002; `http://pargram.b.uib.no/`; see § 6.4.5).

2.4 LDDs, discourse functions and functional uncertainty

In Polish and many other languages question words (or *wh*-words) may appear outside the clause to which they belong semantically – even though they belong to a given phrase at the level of c-structure, they may depend on a different phrase at the level of f-structure. Consider the following example:

(2.51)	Komu	chcecie	pomóc?	
	who.DAT	want	help	
	‘Who do you want to help?’			(NKJP)

In (2.51) *komu* ‘who’ is not an argument of *chcecie* ‘want’, it depends instead on the verb *pomóc* ‘help’. This is because in Polish question words may be extracted[14] from the phrase on which they depend (which contains their head) and placed in another phrase where they are usually fronted. The initial position in such *wh*-questions is of importance – it marks the prominence of a given item in discourse: it is related to the FOCUS discourse function (DF). The focused element (the member of the FOCUS attribute of the phrase in which it is placed) is at the same time assigned an appropriate grammatical function inside the f-structure of the predicate on which it depends semantically – this is how a long-distance dependency (LDD) is created.

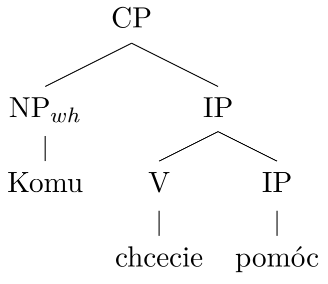
The LDD can be established using an appropriate path which structure-shares relevant fragments of the f-structure. For (2.51), it would be the following path:

(2.52) $(\uparrow \textsc{focus})=(\uparrow \textsc{xcomp obj}_\theta)$

As a result of the annotation provided in (2.52), the f-structure which is the value of $(\uparrow \textsc{focus})$ is at the same time the value of $(\uparrow \textsc{xcomp obj}_\theta)$, i.e., the element which appears in the initial, focused position is additionally assigned the indirect object ($\textsc{obj}_\theta$) grammatical function by its f-structure head (embedded as the infinitival complement: XCOMP).

Going back to (2.51), once *Komu* is assigned the FOCUS discourse function, the annotation in (2.52) structure-shares it with the value of the $\textsc{obj}_\theta$ attribute of the verb *pomóc* – the relevant annotations are attached to the NP_{wh} in (2.54); the structures in (2.53) and (2.55) result.

(2.53)



(2.54)
$$\text{CP} \rightarrow \underset{\substack{(\uparrow \textsc{focus})=\downarrow \\ (\uparrow \textsc{focus})=(\uparrow \textsc{xcomp obj}_\theta)}}{\text{NP}_{wh}} \qquad \underset{\uparrow=\downarrow}{\text{IP}}$$

(2.55)
$$\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{want}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \textsc{focus} & \boxed{3}\left\{\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{who}\text{’} \\ \textsc{case} & \textsc{dat} \end{bmatrix}\right\} \\ \textsc{subj} & \boxed{1}\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{pro}\text{’} \\ \textsc{case} & \textsc{nom} \\ \textsc{num} & \textsc{pl} \\ \textsc{pers} & 2 \end{bmatrix} \\ \textsc{xcomp} & \boxed{2}\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{help}\langle\boxed{1},\boxed{3}\rangle\text{’} \\ \textsc{subj} & \boxed{1} \\ \textsc{obj}_\theta & \boxed{3} \end{bmatrix} \\ \textsc{tense} & \textsc{pres} \end{bmatrix}$$

[14] Note, however, that it does not mean that extraction, as it is commonly understood in transformational theories, is really involved in the examples discussed here – for simplicity, these examples involve scrambling rather than extraction across a finite clause boundary, though the same mechanism may be used in LFG to handle both phenomena.

This is how LDDs were traditionally handled in LFG (see, for instance, King 1995). There are, however, some controversies as to whether *wh*-words should always correspond to the FOCUS discourse function, especially when there is no corresponding c-structure motivation (such as a strict fronting requirement).

The recent proposal advanced in Asudeh 2011 is to use one discourse function, UDF (unbounded dependency function), instead of FOCUS and TOPIC – the motivation is to avoid introducing information about information structure into functional structure. Since LFG provides a range of parallel structures representing different levels of linguistic information, the postulate is to keep these levels separate by using UDF in f-structure, and fine-grained discourse functions such as FOCUS and TOPIC at the level of discourse/information structure.

Let us come back to LDDs: the name comes from the fact that the extracted item can be structurally very distant from the head on which it depends – while in (2.51) the distance spans one infinitival clause, in (2.56) there are two such clauses, as shown in its f-structure in (2.57). The LDD path used to build this f-structure is provided in (2.58).

(2.56) Komu chcecie spróbować pomóc?
who.DAT want try help
'Who do you want to try to help?'

(2.57)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`WANT}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{FOCUS} & \boxed{4}\left\{\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{DAT}\end{array}\right]\right\} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{NOM}\\ \text{NUM} & \text{PL}\\ \text{PERS} & 2\end{array}\right] \\
\text{XCOMP} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`TRY}\langle\boxed{1},\boxed{3}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{XCOMP} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`HELP}\langle\boxed{1},\boxed{4}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ}_\theta & \boxed{4}\end{array}\right]\end{array}\right] \\
\text{TENSE} & \text{PRES}
\end{array}\right]$$

(2.58) $(\uparrow \text{FOCUS})=(\uparrow \text{XCOMP XCOMP OBJ}_\theta)$

There is an LFG device making it possible to express that the LDD path may span an undetermined number of infinitival phrases – it is functional uncertainty, illustrated in (2.59):

(2.59) $(\uparrow \text{FOCUS})=(\uparrow \text{XCOMP}^* \text{ OBJ}_\theta)$

The Kleene star attached to XCOMP, the grammatical function corresponding to the infinitival phrase, expresses the fact that this part of the path can be repeated arbitrarily many times (including zero). As a result, (2.59) can generate the LDD path in (2.52) and (2.58), as well as any longer and shorter paths. In order to ensure that the LDD path contains one or more infinitival phrases, Kleene plus should be used instead of Kleene star, as in (2.60):

(2.60) $(\uparrow \textsc{focus}){=}(\uparrow \textsc{xcomp}^{+}\ \textsc{obj}_{\theta})$

Finally, functional uncertainty can be used to express the fact that the extracted item can correspond to any grammatical function of the head, not only $\textsc{obj}_{\theta}$. This can be done by using a disjunctive specification of the grammatical function at the end of the constraint:

(2.61) $(\uparrow \textsc{focus}){=}(\uparrow \textsc{xcomp}^{*}\ \{\textsc{subj}|\textsc{obj}|\textsc{obj}_{\theta}|\textsc{obl}|\textsc{adjunct}\ \in\})$

2.5 Coordination basics

In LFG coordination is handled in a similar way as adjuncts (mentioned briefly when discussing (2.33)): the f-structure of every conjunct belongs to a set representing the coordinate structure. There is one important difference, however, in comparison to adjuncts: coordinate structures may additionally contain their own attributes with values, alongside the set gathering particular conjuncts.[15] These attributes correspond to the features of the entire coordinate structure, as opposed to the features of particular conjuncts which are stored inside relevant conjuncts. As a result, f-structures representing coordination are hybrid in nature: on the one hand they gather particular conjuncts as elements of a set, on the other they may bear their own features such as the conjunction form or the resolved features of the entire coordinate structure (such as number and gender).

Before proceeding further, let us introduce some basics of how coordinate structures are modelled in LFG. As mentioned above, every conjunct is treated as an element of a set. In the case of adjuncts, they are added to the adjunct set using the following annotation (see the rule adding modifiers to nouns provided in (2.35)):

(2.62) $\downarrow\in(\uparrow \textsc{adj})$

Under coordination, however, conjuncts are not added to the set representing any particular grammatical function – it is only later that such a coordinate structure may be assigned a grammatical function. Conjuncts are simply added to the set representing their mother, the right-hand side category:

(2.63) $\downarrow\in\uparrow$

This is how particular conjuncts are annotated:[16]

(2.64)
$$\begin{array}{llll} \text{XP} \rightarrow & \text{XP} & \text{Conj} & \text{XP} \\ & \downarrow\in\uparrow & & \downarrow\in\uparrow \end{array}$$

XP is a variable, it may be replaced with any phrase structure category. The category corresponding to the conjunction lacks any annotation – according to LFG conventions, it bears the default (co-)head annotation ↓=↑, which unifies the f-structure of the conjunction with the f-structure of the mother. As a result, any annotation of the conjunction will contribute to the annotation of the entire coordinate structure.

Let us take a look at an abstract example of coordination; here are the lexical entries of particular conjuncts and the conjunction:

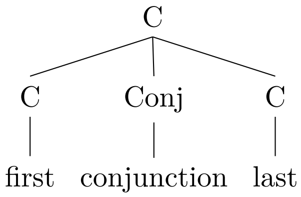
[15] This is the standard treatment, but see Peterson (2004).

[16] The rule provided in (2.64) assumes that coordinated elements represent the same phrasal category. Coordination of unlike categories is discussed later, in ch. 4.

(2.65)	first	C	$(\uparrow \textsc{attr1}) = \textsc{a}$	$(\uparrow \textsc{attr2}) = \textsc{b}$
	last	C	$(\uparrow \textsc{attr1}) = \textsc{a}$	$(\uparrow \textsc{attr2}) = \textsc{c}$
	conjunction	Conj	$(\uparrow \textsc{attr3}) = \textsc{d}$	

The tree and the corresponding f-structure built using the rule in (2.64) are provided below:

(2.66)



(2.67)
$$\left[\begin{array}{l} \left\{ \begin{bmatrix} \textsc{attr1} & \textsc{a} \\ \textsc{attr2} & \textsc{b} \end{bmatrix}, \begin{bmatrix} \textsc{attr1} & \textsc{a} \\ \textsc{attr2} & \textsc{c} \end{bmatrix} \right\} \\ \textsc{attr3} \quad \textsc{d} \end{array}\right]$$

The f-structure representing the coordinate structure consists of an attribute contributed by the conjunction (ATTR3, whose value is D) and a set containing the f-structures of the two conjuncts.

Let us now proceed to a less abstract example which raises some interesting issues:

(2.68) John and Mary walk/*walks.

Both conjuncts are singular, yet the verb must appear in the form appropriate to plural subjects. This is because the number feature of the entire coordinate structure resolves to plural (see Dalrymple and Kaplan 2000 for a detailed discussion of feature resolution in LFG) – this is shown in the f-structure below, which corresponds to the fragment *John and Mary*:

(2.69)
$$\left[\begin{array}{l} \left\{ \begin{bmatrix} \textsc{pred} & \text{`}\textsc{John}\text{'} \\ \textsc{num} & \textsc{sg} \end{bmatrix}, \begin{bmatrix} \textsc{pred} & \text{`}\textsc{Mary}\text{'} \\ \textsc{num} & \textsc{sg} \end{bmatrix} \right\} \\ \textsc{num} \quad \textsc{pl} \\ \textsc{coord-form} \quad \textsc{and} \end{array}\right]$$

Such an f-structure is the result of the following feature resolution annotation:

(2.70) $(\uparrow \textsc{num}) = \textsc{pl}$

(2.70) could be added to the annotation of the conjunction in the rule responsible for the coordination of nominal phrases:[17]

(2.71)
$$\begin{array}{llccc} \text{NP} & \rightarrow & \text{NP} & \text{Conj} & \text{NP} \\ & & \downarrow\in\uparrow & (\uparrow \textsc{num}) = \textsc{pl} & \downarrow\in\uparrow \end{array}$$

It is worth noting that it is on purpose that (2.70) is placed in (2.71) rather than in the lexical entry of the conjunction shown in (2.72) – in this way feature resolution applies where it is appropriate, while the conjunction only contributes an attribute specifying the form of the conjunction (COORD-FORM).

[17] The rule in (2.71) handles coordination with AND-type conjunctions. However, in order to account for agreement patterns with OR-type conjunctions, see (i) below, (2.71) must be modified as in (ii):

(i) John or Mary walks/*walk.

(ii)
$$\begin{array}{llccc} \text{NP} & \rightarrow & \text{NP} & \text{Conj} & \text{NP} \\ & & \downarrow\in\uparrow & [(\uparrow \textsc{coord-form}) =_c \textsc{and}\ (\uparrow \textsc{num}) = \textsc{pl}] & \downarrow\in\uparrow \\ & & & \vee\ (\uparrow \textsc{coord-form}) =_c \textsc{or} & \end{array}$$

The rule in (ii) resolves the number of the coordinate phrase to plural if the conjunction is AND-type (conjoining) and it does not introduce any feature resolution constraints when the conjunction is OR-type (alternative) – single conjunct agreement is used in such cases, whereby the verb agrees with the closest conjunct (see § 3.1.3).

(2.72) and Conj $(\uparrow \textsc{coord-form}) = \textsc{and}$

If (2.70) was placed in the lexical entry of the conjunction, it would apply in every instance of coordination using a given conjunction, for example where clauses are coordinated.

There is an important issue which makes it possible for feature resolution to operate – it is the distinction between distributive and non-distributive features. By default, all features are distributive except for ones that are declared to be non-distributive – such a declaration accompanies the grammar. While outside coordination these feature types are indiscernible, their behaviour differs considerably in this environment: f-descriptions involving distributive features apply to every single element of the coordinate structure. If number (the NUM attribute) was a distributive feature, the f-structure corresponding to the grammatical version of (2.68) would be inconsistent (see § 2.2.2), as shown in (2.73): although particular conjuncts are specified as singular (due to f-descriptions placed in their lexical entries), the plural number assigned to the entire coordinate phrase as a result of feature resolution (the statement (2.70) used in (2.71)) would distribute to particular elements of the coordinate phrase, leading to a clash of respective number values – this is represented in (2.73) as inequality ('≠').

(2.73)
$$ {}^{*}\left[\begin{array}{ll} \multicolumn{2}{l}{\left\{ \left[\begin{array}{ll} \textsc{pred} & \text{`John'} \\ \textsc{num} & \textsc{sg}{\neq}\textsc{pl} \end{array}\right], \left[\begin{array}{ll} \textsc{pred} & \text{`Mary'} \\ \textsc{num} & \textsc{sg}{\neq}\textsc{pl} \end{array}\right] \right\}} \\ \textsc{coord-form} & \textsc{and} \end{array}\right] $$

By contrast, f-descriptions employing non-distributive features do not distribute to particular elements of a coordinate structure; instead, they apply to the entire f-structure, as in (2.69) above, where particular conjuncts are singular but the entire coordinate phrase is plural, as demonstrated by the agreement pattern in (2.68).

After a coordinate structure is formed, it may be assigned a grammatical function. (2.74) is the full f-structure representation of (2.68):

(2.74)
$$ \left[\begin{array}{lll} \textsc{pred} & \multicolumn{2}{l}{\text{`walk}\langle\boxed{1}\rangle\text{'}} \\ \textsc{subj} & \boxed{1} & \left[\begin{array}{ll} \multicolumn{2}{l}{\left\{ \left[\begin{array}{ll} \textsc{pred} & \text{`John'} \\ \textsc{num} & \textsc{sg} \end{array}\right], \left[\begin{array}{ll} \textsc{pred} & \text{`Mary'} \\ \textsc{num} & \textsc{sg} \end{array}\right] \right\}} \\ \textsc{num} & \textsc{pl} \\ \textsc{coord-form} & \textsc{and} \end{array}\right] \\ \textsc{tense} & \multicolumn{2}{l}{\textsc{pres}} \end{array}\right] $$

2.6 Summary

This chapter provided a brief introduction to the very basics of LFG theory, covering issues such as basic parts of an LFG grammar, rules of building f-structures, notation, selected formal devices and basics of the LFG account of coordination. The aim was to keep this introduction as simple as possible, so certain more complicated aspects are discussed only later in this work.

The following chapters build on the notions introduced here when presenting analyses of more complex phenomena such as agreement and case assignment.

Chapter 3

Agreement and case assignment

The aim of this chapter is to provide some background information about selected phenomena in Polish syntax relevant to the discussion in the following chapters. These include the treatment of subject-verb agreement, discussed in § 3.1 (including default agreement and single conjunct agreement, see § 3.1.2 and § 3.1.3, respectively), and structural case assignment to arguments such as subject and object, discussed in § 3.2.

3.1 Subject-verb agreement in Polish

This section discusses the interaction between case assignment to subjects in Polish and subject-verb agreement. It presents agreement patterns such as full agreement (§ 3.1.1), default agreement (§ 3.1.2) and single conjunct agreement (§ 3.1.3).

3.1.1 Full agreement

§ 2.1 offered the first, very simple approach to agreement on the basis of example (2.1), repeated below as (3.1):

(3.1) Facet idzie.
guy.NOM.SG.M1 walk.3.SG
‘A/the guy walks.’

The lexical entry of the verb ((2.3) repeated below as (3.2)) ensures appropriate agreement between the subject and the verb:

(3.2) idzie V (↑ PRED)=‘WALK<(↑ SUBJ)>’
(↑ SUBJ CASE)$=_c$ NOM
(↑ SUBJ NUM)$=_c$ SG
(↑ SUBJ PERS)$=_c$ 3
(↑ TENSE)= PRES

In Polish nominative subjects agree with the verb in all relevant features, which include number, person and gender:[1]

[1] Gender agreement is visible only on past tense verb forms such as in (3.3).

(3.3) Dziecko szło.
child.NOM.SG.N walked.3.SG.N
'A/the child walked.'

With respect to the lexical entry of *idzie* provided in (3.2), the entry of *szło* requires slight modifications to account for (3.3):

(3.4) szło V $(\uparrow \text{PRED})=\text{`WALK}<(\uparrow \text{SUBJ})>\text{'}$
$(\uparrow \text{SUBJ CASE}) =_c \text{NOM}$
$(\uparrow \text{SUBJ GEND}) =_c \text{N}$
$(\uparrow \text{SUBJ NUM}) =_c \text{SG}$
$(\uparrow \text{SUBJ PERS}) =_c 3$
$(\uparrow \text{TENSE}) = \text{PAST}$

The verbs in (3.2) and (3.4), *idzie* and *szło*, respectively, are defined as third person, singular forms. Using constraining equations, they ensure that their subject bears appropriate values of person and number (third and singular, respectively) and that it is marked for the nominative case. While the present tense form, *idzie*, does not impose any gender constraint on its subject, *szło*, the past tense form, requires a subject marked for a specific gender – neuter.

The ungrammaticality of examples provided below seems to lend support to the agreement constraints defined in (3.2) and (3.4):

(3.5) *Faceta idzie.
guy.ACC/GEN.SG.M1 walk.3.SG

(3.6) *Faceci idzie.
guy.NOM.PL.M1 walk.3.SG

(3.7) *Ty idzie.
you.NOM.SG.2 walk.3.SG

(3.8) *Facet szło.
guy.NOM.SG.M1 walked.3.SG.N

(3.5) uses a subject specified for other case (accusative or genitive) than the required nominative. In (3.6) the subject is plural, while singular number is required by the verb. (3.7) features a second person subject but the verb accepts only third person subjects. In (3.8) the human masculine subject violates the verb's neuter gender agreement requirement.

There are, however, examples which show that constraints defined in (3.2) and (3.4) are not appropriate – these feature subjects which trigger default agreement (see Dziwirek 1990), a phenomenon which is discussed in the following section.

3.1.2 Default agreement

3.1.2.1 Non-agreeing numerals

Once data provided by agreement with numeral phrases is taken into consideration, some constraints defined in § 3.1.1 in the lexical entries of verbs (see (3.2) and (3.4)) prove too strong as they predict sentences provided in (3.9) and (3.10), to be ungrammatical, contrary to fact:

(3.9) Dwóch facetów idzie.
two.ACC.SG.M1 guy.GEN.PL.M1 walk.3.SG
'Two guys walk.' (Google)

(3.10) Dwóch facetów szło.
two.ACC.PL.M1 guy.GEN.PL.M1 walked.3.SG.N
'Two guys walked.' (NKJP)

On the basis of the agreement pattern displayed with the adjective TEN 'this' in (3.11), it is possible to show that the numeral subject in (3.9)–(3.10) is marked for the accusative rather than nominative case – the nominative form is not allowed, while the accusative is grammatical:[2]

(3.11) Tych / *Ci dwóch facetów idzie.
this.ACC/GEN.PL.M1 this.NOM.PL.M1 two.ACC.PL.M1 guy.GEN.PL.M1 walked.3.SG.N
'These two guys walked.'

For a thorough discussion of the idea that the numeral subject in sentences such as (3.9)–(3.10) is in the accusative rather than nominative, as claimed by many linguists (see Saloni 2005 for an example) who choose to adhere to the traditional view that the inherent characteristic of the subject is that it must be marked for the nominative case, see Przepiórkowski 1999, 2004b.

So, in (3.9) the subject is a numeral phrase marked for the accusative case and plural number, while the verb, according to its lexical entry in (3.2), requires a singular subject marked for the nominative case. Moreover, following the information in the lexical entry in (3.4), the verb in (3.10) is expected to take a neuter subject, while it is human masculine and besides, as explained above, accusative and plural, which also conflicts with the agreement requirements of the verb.

It is not the case, however, that all numeral subjects display the agreement pattern presented in (3.9)–(3.10) – it depends on the particular form of the numeral used: the numeral in (3.12) triggers full agreement with the verb, unlike the one in (3.10).

(3.12) Dwaj faceci szli.
two.NOM.PL.M1 guy.NOM.PL.M1 walked.3.PL.M1
'Two guys walked.' (Google)

Examples provided above contain forms of the numeral DWA 'two': in (3.12) an agreeing form is used – both the numeral and the accompanying nominal are specified for the nominative case. By contrast, (3.9)–(3.10) feature a non-agreeing form of the numeral – the numeral is marked for the accusative case while the nominal is in the genitive case. This difference in case agreement between the numeral and the nominal is modelled by the feature known under the name of accommodability (Bień and Saloni 1982), which takes one of two values: agreeing or non-agreeing.

As shown above, accommodability influences verbal agreement: agreeing numeral forms trigger full agreement between the subject and the verb, as in (3.12), while non-agreeing forms

[2] In (3.11) the form *Tych* is glossed as either accusative or genitive, while the form *dwóch* is glossed as only accusative despite the fact that it is also syncretic with the genitive. This is because it can be shown, on the basis of agreement with feminine nominals, that the numeral can only be marked for the accusative case (*pięć*; genitive *pięciu* is ungrammatical), while the modifier can be marked for the accusative (*Te*) or genitive (*Tych*) case because it may agree either with the numeral head, or with the genitive nominal:

(i) Te/Tych pięć/*pięciu kobiet szło.
this.ACC/GEN.PL.F five.ACC/GEN.PL.F woman.GEN.PL.F walked.3.SG.N

Though the numeral form *pięć* is syncretic with the nominative case, this possibility is excluded on the basis of examples such as (3.11), where a nominative modifier is impossible.

trigger default agreement (third person, singular, neuter), compare (3.10). For the purposes of modelling agreement, the information about the type of the numeral form is represented in its f-structure as an appropriate value, CONGR for agreeing forms and REC for non-agreeing ones, of the ACM attribute, which corresponds to accommodability, the feature introduced above.

While omitting an agreeing numeral form does not affect the grammaticality of the sentence (see (3.13) corresponding to (3.12)), removing a non-agreeing numeral form results in ungrammaticality (as shown in (3.14) for (3.10)).

(3.13) (Dwaj) faceci szli. (3.14) *(Dwóch) facetów szło.

This suggests that it is the numeral that is the head of the phrase, while the accompanying nominal should be analysed as its dependent. Since there seems to be no motivation to do otherwise, a unified treatment of numerals is adopted in this work, whereby the numeral is always the head and the nominal is its object – either agreeing or marked for the genitive case (see Saloni and Świdziński 2001).

The f-structures of the numerals used in examples above are provided below: (3.15) corresponds to (3.12), while (3.16) is the f-structure for (3.10).

(3.15)

$$\left[\begin{array}{ll} \text{PRED} & \text{`TWO}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{`GUY'} \\ \text{CASE} & \text{NOM} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{PL} \end{array}\right] \\ \text{ACM} & \text{CONGR} \\ \text{CASE} & \text{NOM} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{PL} \end{array}\right]$$

(3.16)

$$\left[\begin{array}{ll} \text{PRED} & \text{`TWO}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{`GUY'} \\ \text{CASE} & \text{GEN} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{PL} \end{array}\right] \\ \text{ACM} & \text{REC} \\ \text{CASE} & \text{ACC} \\ \text{GEND} & \text{M1} \\ \text{NUM} & \text{PL} \end{array}\right]$$

Using the information about accommodability, it is possible to introduce changes necessary to account for agreement patterns with numeral subjects. It should suffice to change the lexical entries of selected third person singular verb forms: *idzie*, the present tense form, and one of the past tense forms, namely the one specified for neuter gender, *szło*.

(3.17) is a template[3] aiming at capturing agreement patterns sketched above. It is defined as a disjunction of two other templates: (3.18) allows subjects marked for the nominative case, while (3.19) takes care of accusative non-agreeing numeral subjects.[4]

(3.17) $\text{AGR-CASE} \equiv \text{AGR-CASE-NOM} \vee \text{AGR-CASE-NUMACC}$

(3.18) $\text{AGR-CASE-NOM} \equiv (\uparrow \text{SUBJ CASE}) =_c \text{NOM}$

(3.19) $\text{AGR-CASE-NUMACC} \equiv (\uparrow \text{SUBJ ACM}) =_c \text{REC} \wedge (\uparrow \text{SUBJ CASE}) =_c \text{ACC}$

It is not enough, however, to simply replace (3.18), the constraint which requires the subject to be marked for the nominative case, with the one provided in (3.17) – appropriate (non-)agreement must also be taken into consideration.

[3] Templates were introduced in § 2.2.3.

[4] This formalisation assumes that there are no nominative non-agreeing numerals.

After replacing (3.18) with (3.17) in the lexical entries of *idzie* and *szło* ((3.2) and (3.4), respectively), these verbs would still require a singular subject. Moreover, while *idzie* is underspecified for gender, *szło* would additionally impose a constraint requiring a subject with neuter gender specification. As shown in examples featuring non-agreeing numeral subjects ((3.9) and (3.10)), these constraints are not applicable in such cases and the lexical entries of relevant verb forms must be revised accordingly:[5]

(3.20) idzie V (↑ PRED)=‘WALK<(↑ SUBJ)>’
[[AGR-CASE-NOM ∧ (↑ SUBJ NUM)$=_c$ SG ∧ (↑ SUBJ PERS)$=_c$ 3]
∨
AGR-CASE-NUMACC]
∧
(↑ TENSE)= PRES

(3.21) szło V (↑ PRED)=‘WALK<(↑ SUBJ)>’
[[AGR-CASE-NOM ∧ (↑ SUBJ GEND)$=_c$ N ∧ (↑ SUBJ NUM)$=_c$ SG ∧
(↑ SUBJ PERS)$=_c$ 3]
∨
AGR-CASE-NUMACC]
∧
(↑ TENSE)= PAST

The non-agreement pattern discussed above is known as default agreement (see Dziwirek 1990 for discussion): unless the subject is marked for the nominative case, the verb appears in the third person singular neuter[6] form. It is not the case, however, that any non-nominative element may serve as the subject; see (3.5), where the candidate is a syncretic accusative or genitive noun.

3.1.2.2 Clausal subjects

Apart from accusative non-agreeing numerals presented above, possible default agreement triggers include clausal subjects, as in the examples below:[7]

(3.22) Naszych gości dziwiło, że mamy tak dużo obowiązków.
our guests.ACC puzzled.3.SG.N that have so many duties
‘(The fact) that we have so many duties puzzled our guests.’ (NKJP)

(3.23) Cieszyło ją, że mam tak oryginalne zainteresowania.
made.happy.3.SG.N she.ACC that have.1.SG so original interests
‘(The fact) that I have so original interests made her happy.’ (NKJP)

(3.24) Że Janek kochał Ewę było dla wszystkich oczywiste.
that Janek loved Ewa was for all obvious
‘That Janek loved Ewa was obious to everyone.’ (Dziwirek 1990, p. 154, ex. (17b))

[5] To make embedded statements more readable, grouping of statements is indicated using square brackets.
[6] If the verb displays gender marking, as in the past tense.
[7] Polish diacritics were added (restored) in (3.24).

In order to account for such examples, further changes need to be introduced in the lexical entries of verb forms used with default agreement.

In case of (3.22), the lexical entry of *dziwiło* would require minimal changes with respect to the entry of *szło* provided in (3.21), most importantly adding the disjunct provided in (3.25) to allow, alongside nominative and accusative non-agreeing subjects, a subject which contains the complementiser ŻE:

(3.25) (↑ SUBJ COMP-FORM)$=_c$ ŻE

The constraint provided in (3.25) ensures that the subject contains the complementiser ŻE required by the verb DZIWIĆ. However, other verbs may have different requirements, reflected as appropriate values of the COMP-FORM attribute.[8] A template may be used to obtain, among others, the constraint in (3.25):

(3.26) GF-COMP-FORM-SUBC(GF CF) ≡ (↑ GF COMP-FORM)$=_c$ CF

(3.27) GF-COMP-FORM-SUBC(SUBJ ŻE)

Unlike templates presented so far, the template GF-COMP-FORM-SUBC defined in (3.26) takes parameters – depending on the values provided when making a call to such a template, different constraints result. (3.26) takes two parameters: GF, which hosts the grammatical function, and CF, which hosts the form of the complementiser – these parameters can in principle take any value. In (3.27) the template (3.26) is called with SUBJ as the value of the GF parameter and ŻE as the value of CF parameter. As a result, (3.27) introduces the constraint in (3.25).

It is not the case, however, that sentential subjects are allowed with every predicate:

(3.28) *Śpi, że pięć kobiet szło.
sleep.3.SG that five.ACC.PL.F woman.GEN.PL.F walked.3.SG.N
'(The fact) that five women walked sleeps.' (intended)

Due to this fact, unlike in the case of numeral subjects, which seem to be allowed almost universally, there is no need to modify the lexical entries of default agreement forms of all verbs. It is enough to add the template call in (3.27) to the lexical entries of forms which allow clausal subjects with a ŻE-type complementiser. The full[9] lexical entry of *dziwiło* used in example (3.22) is as follows:

(3.29) dziwiło V (↑ PRED)='PUZZLE<(↑ SUBJ)(↑ OBJ)>'
[[AGR-CASE-NOM ∧ (↑ SUBJ GEND)$=_c$ N ∧ (↑ SUBJ NUM)$=_c$ SG
∧ (↑ SUBJ PERS)$=_c$ 3]
∨
AGR-CASE-NUMACC
∨
GF-COMP-FORM-SUBC(SUBJ ŻE)]
∧
(↑ TENSE)= PAST

[8] Interrogative clauses are handled using the CLAUSE-TYPE attribute – see the discussion of (4.79) in § 4.5.2.2.
[9] Case assignment constraints related to the object are omitted. This issue is discussed in § 3.2.3.

(3.29) accepts three types of subjects with different agreement patterns. Full subject-verb agreement is possible with nominative subjects (the subject must be marked for neuter gender and singular number). Default agreement is possible with non-agreeing accusative numerals. Clauses featuring ŻE-type complementisers are the last type of subjects accepted by *dziwiło*.

3.1.3 Single conjunct agreement

Even though in the examples provided below the subject is plural as a result of coordination, the verb (the agreement target) is specified for singular number, which starkly shows the lack of expected agreement with resolved features of the entire coordinate phrase (see § 2.5 for some discussion of the resolution of number under coordination):

(3.30) trwa transformacja i proces dostosowywania się
continue.3.SG transformation.NOM.SG.F and process.NOM.SG.M3 adjusting REFL
do wymogów
to requirements
'The transformation and the process of adjusting to requirements continue.' (NKJP)

(3.31) panowała harmonia i spokój
prevail.3.SG.F harmony.NOM.SG.F and peace.NOM.SG.M3
'Harmony and peace prevailed.' (NKJP)

(3.32) Pan Mirosław i czternastu ludzi pracowało dzień
Mr Mirosław.NOM.SG.M1 and fourteen.ACC.PL.M1 man.GEN.PL.M1 worked.3.SG.N day
i noc
and night
'Mr Mirosław and fourteen men worked night and day.' (NKJP)

According to standard feature resolution rules (such as (2.71) in § 2.5), verbs in (3.30)–(3.32) should be marked for plural number as a result of agreement with their respective subjects, as in (3.33), which is a modified version of (3.30):

(3.33) trwają transformacja i proces
continue.3.PL transformation.NOM.SG.F and process.NOM.SG.M3

Instead, the verb in (3.30) agrees with only one of the conjuncts – typically it is the closest conjunct (*transformacja*). But since the verb form in (3.30) is present tense and therefore there is no gender marking, it makes it possible, in theory, to analyse this example as an instance of furthest conjunct agreement (with *proces*).

However, there is no such ambiguity with past tense verbs because they display gender marking, which, assuming that conjuncts are marked for different gender values, makes it possible to unambiguously identify the agreement controller. In (3.31) the verb agrees with the leftmost conjunct (*harmonia*), since it is the one which is the closest to the preceding verb. By contrast, in (3.32) the verb follows the subject, so it is the the rightmost conjunct (*czternastu ludzi*) which serves as the agreement controller, being closest to the verb. Since it is a non-agreeing numeral, it triggers default agreement on the verb: third person, singular number, neuter gender, as described in § 3.1.2.1. Under feature resolution, the verb would be expected to display the following agreement features: third person, plural number and human masculine (M1) gender:

(3.34)	Pan Mirosław	i	czternastu	ludzi	pracowali
	Mr Mirosław.NOM.SG.M1	and	fourteen.ACC.PL.M1	man.GEN.PL.M1	worked.3.PL.M1

In constructions featuring single conjunct agreement (SCA), while the entire coordinate phrase is a subject, only one of the conjuncts serves as the agreement controller – usually[10] it is the conjunct which is closest to the verb.

LFG analyses of representation and formalisation of SCA include Kuhn and Sadler 2007 and Dalrymple and Hristov 2010. The analysis discussed below is based on one of the approaches presented in Kuhn and Sadler 2007, § 3.2.2.[11] The following template is used to determine which conjunct is closest to the verb:

(3.36) AGR-CCA-PATH(CONTROLLER CONTROLLEE) ≡
(CONTROLLER ∈)=%L
∧
[CONTROLLEE <H CONTROLLER ∧ ¬[(CONTROLLER ∈) <H %L]
∨
CONTROLLER <H CONTROLLEE ∧ ¬[%L <H (CONTROLLER ∈)]]

The template defined in (3.36) is called inside the lexical entry of the verb, as an alternative to full agreement (with a non-coordinate subject or with resolved features under coordination). This template is called with two parameters: the first one is CONTROLLER, which hosts the path to the agreement controller, the subject: (↑ SUBJ), while the other, CONTROLLEE, contains the path to the agreement target, the verb: ↑. The statement in (3.36) consists of two conjoined statements – let us discuss them one by one.

The second line of (3.36) constitutes the first conjunct of the statement. It contains local variable assignment: %L is assigned to (CONTROLLER ∈), which expands to (↑ SUBJ ∈) after substituting the path to the subject for CONTROLLER (as explained above). The aim of this line is to select a conjunct (any conjunct) from the set corresponding to the subject.

Lines 4–6 of (3.36) form its second conjunct. This fragment is a disjunctive statement handling the selection of the conjunct which is closest to the verb – it uses the head-precedence operator (<H) to achieve this.

[10] Though not very frequent, there are instances of furthest conjunct agreement (FCA) in Polish:

(3.35)	Czołowa	trójka	oraz	sześć	kolejnych	zespołów	będzie
	top.NOM.SG.F	three.NOM.SG.F	and	six.ACC.PL.M3	next.GEN.PL.M3	team.GEN.PL.M3	will.3.SG

mogła	wystawić	w	Atenach	czteroosobowe	reprezentacje
be able to.3.SG.F	field.INF	in	Athens	four-people	representations

'The top three and the next six teams will be able to field four-people representations in Athens.' (NKJP)

While in (3.35) the closest conjunct is *sześć kolejnych zespołów*, the agreement features of the verb, *będzie mogła*, are controlled by *Czołowa trójka* – there is full agreement in all relevant features (such as number and gender – singular and feminine, respectively), while the closest conjunct would trigger default agreement since it is a non-agreeing numeral, see § 3.1.2.1.

[11] Since closest conjunct agreement (CCA) is not the main focus of this work, this analysis was chosen for presentation because of its relative simplicity. It is not, however, the best analysis: for instance, it does not handle nested coordination – see Dalrymple and Hristov 2010 for an analysis which takes this phenomenon into account.

The first disjunct (fourth line of (3.36)) handles the case where the subject precedes the verb: in this situation the leftmost conjunct is chosen as the closest one – this is done by ensuring that there is no conjunct that would be closer to the verb than the one assigned to %L variable.

The second disjunct (the last line of (3.36)) handles the opposite case, namely the situation where the verb precedes the subject: the rightmost conjunct is then chosen as the closest – similarly as above, this is achieved by making sure that there is no conjunct to the right of the selected one (bound to %L variable).

Summing up, the effect of calling the template defined in (3.36) is choosing the conjunct which is closest to the verb. Such information can be used by relevant templates handling subject-verb agreement to ensure appropriate agreement (full or default, see § 3.1.1 and § 3.1.2, respectively) with the closest conjunct.

3.2 Case assignment in Polish

This section briefly discusses structural case assignment in Polish. It assumes the distinction between structural and lexical case – similar to that introduced in the early Government and Binding theory (Rouveret and Vergnaud 1980, Vergnaud 1982, Chomsky 1980, 1981, as well as – apparently independently – Babby 1980a,b) – but here based on the observation that sometimes the case assigned to certain grammatical functions depends on the syntactic environment: while structural case depends on factors related to the syntactic environment (such as part of speech of the head assigning case, presence or absence of negation), lexical case (also known as inherent) is assigned directly in the lexicon, without taking any such factors into consideration. This is the understanding of the structural vs lexical case distinction in the HPSG literature (e.g. Pollard 1994, Heinz and Matiasek 1994, Przepiórkowski 1999, Przepiórkowski *et al.* 2002). In LFG, on the other hand, non-lexical case is assigned mostly on the basis of grammatical functions rather than structural factors (cf., e.g., Neidle 1982, 1988), with purely structural case playing a marginal role in earlier LFG analyses (cf., e.g., King 1995, p. 178).

§ 3.2.1 discusses terminology related to case assignment, namely the difference between case assignment and case checking in LFG, § 3.2.2 discusses structural case assignment to the subject, while § 3.2.3 is concerned with structural objects.

3.2.1 Assignment or checking?

As explained in Patejuk and Przepiórkowski 2014d, while it is customary to talk about structural case assignment, this does not necessarily mean that a particular analysis actually uses case assignment rather than case checking. Consider the following minimal pair of examples:

(3.37) $(\uparrow \textsc{subj case}) = \textsc{dat}$

(3.38) $(\uparrow \textsc{subj case}) =_c \textsc{dat}$

(3.37) is a defining equation, which, as discussed in § 2.1, introduces an attribute-value pair to the f-structure – it assigns the dative case to the subject. This is an instance of genuine case assignment. By contrast, (3.38) is a constraining equation, which only checks whether the relevant attribute-value pair already exists in the f-structure: it checks whether the subject is marked for the dative case, but it does not introduce any new attribute-value pairs.

While the discussion presented in this work uses traditional terms, saying for instance that in (3.38) the subject is assigned the dative case, the actual functional descriptions given below in fact use constraining equations.

3.2.2 Subject

The issue of case assignment to Polish structural subjects was raised in § 3.1.1 and § 3.1.2.1 when discussing subject-verb agreement. This section briefly summarises relevant fragments of this discussion.

The following generalisation describes case assignment to the structural subject of finite verb forms in Polish:

(3.39) subject bearing structural case must typically be in the nominative,

(3.40) unless the subject is a non-agreeing numeral phrase; then it must be in the accusative.

The difference in case marking of the subject has influence on subject-verb agreement: while nominative subjects trigger full agreement (in person, number and gender) with the verb, as in (3.41), accusative numerals trigger default agreement (third person, singular number, neuter gender) in this environment, as demonstrated in (3.42).

(3.41) Ludzie pracowali ciężko po 16-18 godzin na dobę.
man.NOM.PL.M1 worked.3.PL.M1 hard DISTR 16–18 hours for day
'People worked hard 16–18 hours a day.' (NKJP)

(3.42) Ośmiu ludzi pracowało po dwanaście godzin przez cztery miesiące.
eight.ACC.PL.M1 man.GEN.PL.M1 worked.3.SG.N DISTR twelve hours for four months
'Eight people worked twelve hours a day each for four months.' (NKJP)

(3.41) satisfies the constraint in (3.39), while (3.40) allows for sentences such as (3.42).

The above rules determining which case should be assigned to the subject as the structural case can be formalised using the following constraints repeated from § 3.1.2.1:

(3.43) AGR-CASE-NOM $\equiv$ ($\uparrow$ SUBJ CASE)$=_c$ NOM

(3.44) AGR-CASE-NUMACC $\equiv$ ($\uparrow$ SUBJ ACM)$=_c$ REC $\wedge$ ($\uparrow$ SUBJ CASE)$=_c$ ACC

(3.43) corresponds to (3.39), while (3.44) is a formalisation of (3.40).

It is worth noting that constraints provided in (3.43)–(3.44) do not apply to subjects[12] of non-finite verbal forms such as gerunds (as in (3.45)) and adjectival participles (see (3.46)–(3.48)) – the rules of case assignment to the subject are different in these environments.

[12] On the basis of binding and control phenomena, it is assumed that not only finite verb forms have a subject, but also gerunds, participles and derived nominals.

(3.45) Jak woda wstrzymuje działanie ognia, tak brak miłości wstrzymuje działanie
as water stops operate.GER fire.GEN so lack love stops operate.GER
Ducha Świętego.
Holy Spirit.GEN
'Like water stops fire, so the lack of love stops the Holy Spirit.' (NKJP)

(3.46) Odeszli też w wieczność ludzie mieszkający wówczas we Lwowie
left also in eternity people.NOM living.NOM then in Lwów
'People living at that time in Lwów also passed away.' (NKJP)

(3.47) podobno widziano ich wchodzących razem do pobliskiej bramy.
reportedly see.IMPS they.ACC entering.ACC together to nearby gate
'Reportedly, they were seen entering a nearby gate together.' (NKJP)

(3.48) teraz widzimy go trzepocącym ledwie wyczuwalnie
now see he.ACC fluttering.INST barely noticeably
'Now we see him fluttering barely noticeably.' (NKJP)

Gerunds uniformly assign the genitive case to their structural subject (see (3.45)) – the relevant constraint is provided in (3.49)

(3.49) (↑ SUBJ CASE)$=_c$ GEN

By contrast, adjectival participles seem to impose no specific case constraints: while they tend to agree with their controller (their subject), as in (3.46), where both the participle (*mieszkający*) and the noun (*ludzie*) are marked for the nominative case, there are some environments where agreement between the participle and its controller is not obligatory. For example, this is the case with secondary predicates, where the non-agreeing instrumental case is possible as an alternative to case agreement – see (3.48) and (3.47), respectively. For this reason, no case assignment constraints are imposed on the subject of such a form – it is assigned by other rules (for example by the verb WIDZIEĆ 'see' in (3.47)–(3.48)).

3.2.3 Object[13]

3.2.3.1 Basic generalisations

In Polish, the value of case assigned to objects requiring structural case depends on the syntactic context, which includes factors such as the part of speech of the head which assigns case and the availability of sentential negation:

(3.50) Proponuję też poczytanie książki
suggest.1.SG also read.GER book.GEN.SG.F
'I also suggest reading a book.' (NKJP)

(3.51) Poczytam książkę.
read.1.SG book.ACC.SG.F
'I'll read a book.' (NKJP)

[13] This section is based on Patejuk and Przepiórkowski 2012a, 2014d.

(3.52) nie poczytają książki czy gazety.
NEG read.3.PL book.GEN.SG.F or newspaper.GEN.SG.F
'They can't read a book or a newspaper.' (NKJP)

With the exception of gerunds, which uniformly assign the genitive case to their structural objects (see (3.50)), the following basic rules govern structural case assignment to objects in Polish:

(3.53) objects bearing structural case are in the accusative,

(3.54) unless they are in the syntactic scope of sentential negation, in which case they are in the genitive (so-called Genitive of Negation, GoN; cf., e.g., Willim 1990, Tajsner 1990, Przepiórkowski 2000, Błaszczak 2001).

(3.53) accounts for the accusative case in (3.51) – since there is no negation in the f-structure of the verb assigning structural case (see (3.55)), the object is marked for the accusative case (*książkę*). By contrast, the verb in (3.52) is negated (the top-level f-structure in (3.56) contains the attribute NEG, whose value is +) and therefore both conjuncts of the coordinate object bear the genitive case (*książki czy gazety*), in accordance with (3.54).[14]

(3.55)
$$\left[\begin{array}{ll} \text{PRED} & \text{`READ}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\ \text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`BOOK'}\\ \text{CASE} & \text{ACC}\end{array}\right]\end{array}\right]$$

(3.56)
$$\left[\begin{array}{ll} \text{PRED} & \text{`READ}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\ \text{OBJ} & \boxed{2}\left[\begin{array}{l}\left\{\begin{array}{l}\left[\begin{array}{ll}\text{PRED} & \text{`BOOK'}\\ \text{CASE} & \text{GEN}\end{array}\right], \\ \left[\begin{array}{ll}\text{PRED} & \text{`NEWSPAPER'}\\ \text{CASE} & \text{GEN}\end{array}\right]\end{array}\right\} \\ \begin{array}{ll}\text{COORD-FORM} & \text{CZY}\end{array}\end{array}\right] \\ \text{NEG} & + \end{array}\right]$$

The generalisations provided in (3.53)–(3.54) could be formalised as follows:

(3.57) $\text{STRCASE} \equiv [\text{AFFIRMATIVE} \ \vee \ \text{NEGATIVE}]$

(3.58) $\text{AFFIRMATIVE} \equiv [\neg(\uparrow \text{NEG}) \ \wedge \ (\uparrow \text{OBJ CASE}) =_c \text{ACC}]$

(3.59) $\text{NEGATIVE} \equiv [(\uparrow \text{NEG}) =_c + \ \wedge \ (\uparrow \text{OBJ CASE}) =_c \text{GEN}]$

(3.57) is the top-level template handling structural case assignment to the object[15] – it corresponds to a disjunction of templates defined in (3.58) and (3.59). (3.58) uses a negated existential constraint ($\neg(\uparrow$ NEG$)$) to ensure that there is no negation in the f-structure of the head and requires the object to be marked for the accusative case (ACC) – this accounts for examples such as (3.51); see (3.55) for its f-structure. By contrast, (3.59) checks that the verb is negated

[14] F-structures in (3.55) and (3.56) are simplified: only features relevant to the discussion are represented.

[15] Templates called by (3.57) assume that the grammatical function of the object is OBJ. However, these templates can be redefined so as to take a parameter which would take the relevant grammatical function – OBJ would then be replaced by a variable such as GF, whose value would be specified when calling the template.

((↑ NEG)$=_c$ +) and checks that its object bears the genitive case (GEN) – these requirements are satisfied by sentences such as (3.52); compare its f-structure provided in (3.56).

To ensure proper case assignment to structurally case-marked objects of verbs, the call to the template defined in (3.57) should be placed in the lexical entries of relevant verbs – see (3.60), which is the lexical entry for *poczytam*, as used in (3.51).

(3.60) poczytam V (↑ PRED)='READ<(↑ SUBJ)(↑ OBJ)>'
(↑ SUBJ CASE)$=_c$ NOM
(↑ SUBJ NUM)$=_c$ SG
(↑ SUBJ PERS)$=_c$ 1
STRCASE
(↑ TENSE)= FUT

Since *poczytam* is not a form which is possible with default agreement, only a nominative subject is allowed in (3.60). Structural case assignment to the object is handled by the call to the template STRCASE – appropriate templates are called: in the case of (3.51) the object satisfies the generalisation in (3.53), which is handled by the template AFFIRMATIVE defined in (3.58).

3.2.3.2 GoN in verb chains

There is, however, an outstanding issue which may be illustrated using the following examples:

(3.61) Nie chcesz poczytać Kodeksu Prawa kanonicznego.
NEG want.2.SG read.INF Code of Canon Law.GEN.SG.M3
'You don't want to read the Code of Canon Law.' (NKJP)

(3.62) Mama nie chce iść poczytać książkę.
mum NEG want.3.SG go.INF read.INF book.ACC.SG.F
'Mum doesn't want to go and read a book.' (NKJP)

These examples demonstrate how structural case assignment operates in Polish verb chains. They immediately show that (3.57), the formalisation of generalisations provided in (3.53)–(3.54), needs some refinement – it would only allow an accusative object (*książkę*) in (3.62), but it would reject the genitive object (*Kodeksu. . .*) in (3.61) as ungrammatical, counter to fact.

While in the absence of sentential negation objects of verbs which require structural case must be marked for the accusative case, when negation is transferred (sentential negation is available but is not local to the predicate assigning case), as discussed in Przepiórkowski 2000, GoN is optional – the structural object can be marked for either the accusative or genitive case, as in (3.61)–(3.62); the corresponding f-structures are provided in (3.63) and (3.64), respectively.

(3.63)

$$\left[\begin{array}{ll} \text{PRED} & \text{`WANT}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\ \text{XCOMP} & \boxed{2}\left[\begin{array}{ll} \text{PRED} & \text{`READ}\langle\boxed{1},\boxed{3}\rangle\text{'} \\ \text{SUBJ} & \boxed{1} \\ \text{OBJ} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`CODE...'} \\ \text{CASE} & \text{GEN}\end{array}\right]\end{array}\right] \\ \text{NEG} & + \end{array}\right]$$

(3.64)
$$\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{want}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{mum}\text{’}\end{array}\right] \\ \textsc{xcomp} & \boxed{2}\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{go}\langle\boxed{1},\boxed{3}\rangle\text{’} \\ \textsc{subj} & \boxed{1} \\ \textsc{xcomp} & \boxed{3}\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{read}\langle\boxed{1},\boxed{4}\rangle\text{’} \\ \textsc{subj} & \boxed{1} \\ \textsc{obj} & \boxed{4}\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{book}\text{’} \\ \textsc{case} & \textsc{acc} \end{array}\right] \end{array}\right] \end{array}\right] \\ \textsc{neg} & + \end{array}\right]$$

There are, however, certain constraints which restrict the environments in which negation may be transferred to other clauses; consider the example below:

(3.65) (Wcale) nie powiedziałeś, że poczytasz Kodeks/*Kodeksu.
not at all NEG said.2.SG that read.2.SG code.ACC/GEN.SG.M3
‘You did not say that you will read the Code.’

(3.65) shows that sentential negation cannot be transferred to sentential clauses, as opposed to infinitival clauses, which was demonstrated in (3.61). LFG provides a convenient means to distinguish between the two embedded clause types, infinitival and sentential: as discussed in § 2.2.1, typically the former correspond to the XCOMP grammatical function, while the latter are represented in the f-structure as COMP.[16] Compare the f-structure representation of (3.65) provided in (3.66)[17] below with (3.63):

(3.66)
$$\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{say}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{pro}\text{’}\end{array}\right] \\ \textsc{comp} & \boxed{2}\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{read}\langle\boxed{3},\boxed{4}\rangle\text{’} \\ \textsc{subj} & \boxed{3} \\ \textsc{obj} & \boxed{4}\left[\begin{array}{ll} \textsc{pred} & \text{‘}\textsc{code}\ldots\text{’} \\ \textsc{case} & \textsc{acc} \end{array}\right] \\ \textsc{comp-form} & \text{Ż}\textsc{e} \end{array}\right] \\ \textsc{neg} & + \end{array}\right]$$

Even though the verb POWIEDZIEĆ ‘say’ is negated in (3.66), the sentential negation available in the main clause cannot be used for the purposes of case assignment by the lower predicate, the clausal complement (COMP) POCZYTAĆ ‘read’. As a result, unlike in (3.63), the object of POCZYTAĆ may only be marked for the accusative case in (3.66) – the genitive case marking leads to ungrammaticality, as indicated in (3.65).

This is how (3.57), the template handling case assignment to objects, can be modified so as to take long-distance genitive of negation into account, together with its optionality:

[16] However, when such a clause can be coordinated with a nominal conjunct, it would be classified as one of object grammatical functions – see ch. 4 for discussion.

[17] The f-structure in (3.66) corresponds to the grammatical version of (3.65) which features the object marked for the accusative case.

(3.67) $\textsc{strcase} \equiv [\textsc{affirmative} \;\vee\; \textsc{negative}]$

(3.68) $\textsc{affirmative} \equiv [\neg((\textsc{xcomp}^{*}\uparrow)\;\textsc{neg}) \;\wedge\; (\uparrow\textsc{obj case}) =_{c} \textsc{acc}]$

(3.69) $\textsc{negative} \equiv [\textsc{anyneg} \;\wedge\; \textsc{negtype}]$

(3.70) $\textsc{anyneg} \equiv ((\textsc{xcomp}^{*}\uparrow)\;\textsc{neg}) =_{c} +$

(3.71) $\textsc{negtype} \equiv [\textsc{locneg} \;\vee\; \textsc{nonlocneg}]$

(3.72) $\textsc{locneg} \equiv [(\uparrow\textsc{neg}) =_{c} + \;\wedge\; (\uparrow\textsc{obj case}) =_{c} \textsc{gen}]$

(3.73) $\textsc{nonlocneg} \equiv [\neg(\uparrow\textsc{neg}) \;\wedge\; (\uparrow\textsc{obj case}) \in_{c} \{\textsc{acc}, \textsc{gen}\}]$

The top-level template remains unchanged: (3.67) is a disjunction of two templates, AFFIRMATIVE and NEGATIVE, but see (3.68) and (3.69), respectively, for their new definitions.

(3.68), the first disjunct of (3.67), is a slightly modified version of (3.58). It handles the situation where sentential negation is not available at all – neither locally (on the verb assigning case), nor in the verb chain. This is achieved by using a negative existential constraint with an inside-out path, $\neg((\textsc{xcomp}^{*}\uparrow)\;\textsc{neg})$, which makes it possible to reach into any number of successive higher predicates subcategorising for XCOMP, an infinitival complement, and ensure that none of these predicates is negated – the verb which assigns case could be an infinitival complement of some predicate which is negated itself or is an infinitival complement of some higher negated verb. This template handles sentences such as (3.51), where sentential negation is not available anywhere – see (3.55) for its f-structure representation.

(3.69), the second disjunct of (3.67), is defined as a conjunction of two templates: (3.70), which ensures that sentential negation is present at some level of the structure, and (3.71), which assigns an appropriate value of structural case to the object depending on whether negation is local or not to the predicate assigning case.

(3.72), the first disjunct of (3.71), is identical to (3.59) – it assigns the genitive case to the object when the verb assigning case is negated; this is the obligatory genitive of negation clause. It is applied in examples such as (3.52); see (3.56) for the corresponding f-structure.

(3.73), the second disjunct of (3.71), handles the case in which there is no local negation. However, (3.70) called by (3.69) makes sure that there is negation at some level, so transferred negation must be available. In such environments, the genitive of negation is optional – the object of the verb is assigned the accusative (see (3.62) and its f-structure in (3.64)) or genitive case (as in (3.61), the corresponding f-structure is provided in (3.63)).

3.2.3.3 Complex case

So far it was assumed in this work that the CASE attribute takes an atomic value. There are, however, phenomena which require some formal account of forms which seem to bear an ambiguous specification for some features – Dalrymple and Kaplan 2000 discuss the following example of case syncretism from Dyła 1984:

(3.74) Kogo Janek lubi a Jerzy nienawidzi?
who.ACC/GEN Janek likes and Jerzy hates
'Who does Janek like and Jerzy hate?' (Dyła 1984, ex. (2))

In (3.74), the verb *lubi* 'likes' requires its object to bear the accusative case while *nienawidzi* 'hates' requires an object marked for the genitivecase. As a result, the shared object *kogo* 'who' must satisfy two different case requirements at the same time.

In such situations, rather than assigning two different atomic values at the same time, which would cause the f-structure to be inconsistent, the relevant attribute may be set-valued – this is the solution suggested by Dalrymple and Kaplan 2000:

(3.75) kogo Pron (↑ PRED)='WHO'
(↑ CASE)= {ACC,GEN}

(3.76)
$$\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \left\{\text{ACC,GEN}\right\}\end{array}\right]$$

However, the solution of Dalrymple and Kaplan 2000 was proven to have problems with handling modification where the head displays case syncretism, but the modifier is more restricted than the head with respect to the possible values of case it may take (Dalrymple *et al.* 2009). This problem is illustrated using constructed examples in (3.77)–(3.79), where the form *emu* is used as the object – this form is ambiguous between all 7 case specifications possible in Polish (nominative, accusative, genitive, dative, instrumental, locative and vocative). However, the modifier of the object is not so ambiguous: in (3.77) it is accusative (structural case assigned to the object by the verb *widzisz*), while in (3.78) it is unambiguously genitive (the verb *boisz się* assigns the genitive case to its object). Under the analysis with the set-valued representation of case proposed by Dalrymple and Kaplan 2000, (3.79) would be expected to be grammatical, counter to fact – it is ungrammatical because modifiers impose different case requirements on the head: *jakie* requires it to be accusative and not genitive, while *jakiego* requires a genitive head, not accusative.

(3.77) Jakie emu widzisz?
what.ACC emu.ACC see
'What emu do you see?'

(3.78) Jakiego emu boisz się?
what.GEN emu.GEN fear REFL
'What emu are you afraid of?'

(3.79) *Jakie/jakiego emu widzisz i boisz się?
what.ACC/GEN emu.ACC/GEN see and fear REFL
'What emu do you see and are you afraid of?' (intended)

The solution that Dalrymple *et al.* 2009 offered involves using a complex CASE attribute whose value is an f-structure containing attributes which correspond to particular cases with their own values (positive or negative). Let us first discuss the representation of (3.74) under this analysis:

(3.80) kogo Pron (↑ PRED)='WHO'
(↑ CASE {ACC|GEN})= +
(↑ CASE DAT)= −
(↑ CASE INST)= −
(↑ CASE LOC)= −
(↑ CASE NOM)= −
(↑ CASE VOC)= −

(3.81)
$$\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \left[\begin{array}{ll}\text{ACC} & +\\ \text{DAT} & -\\ \text{GEN} & +\\ \text{INST} & -\\ \text{LOC} & -\\ \text{NOM} & -\\ \text{VOC} & -\end{array}\right]\end{array}\right]$$

The lexical entry provided in (3.80) contains the statement $(\uparrow \textsc{case}\ \{\textsc{acc}|\textsc{gen}\})=+$: using functional uncertainty (see § 2.4), it introduces a positive specification (+) for one of the following values of case possible with this form: accusative or genitive. As a result, 2 f-structures are produced: one where ACC has a positive specification and GEN is not present at all and one where GEN has a positive value and ACC is absent. The remaining 5 values of case have negative values (−) since these values of case are not possible with this form.

When case assignment statements using defining[18] equations are used, the verb *lubi* assigns *kogo* the accusative case, while *nienawidzi* assigns the genitive case. As a result, whichever f-structure created by the lexical entry in (3.80) is used, the resulting f-structure for *kogo* is (3.81), where both ACC and GEN have positive values (one of which is introduced by the lexical entry in (3.80), while the other is introduced by the relevant verb).

To account for the data in (3.77)–(3.79), the following lexical entries could be postulated:[19]

(3.82) emu N $(\uparrow \textsc{pred})=\text{`}\textsc{emu}\text{'}$
$(\uparrow \textsc{case}\ \{\textsc{nom}|\textsc{acc}|\textsc{gen}|\textsc{dat}|\textsc{inst}|\textsc{loc}|\textsc{voc}\})=+$
$(\uparrow \textsc{gend})=\textsc{n}$
$(\uparrow \textsc{num})=\textsc{sg}$

(3.83) jakie A $(\uparrow \textsc{pred})=\text{`}\textsc{what}\text{'}$
$((\textsc{adj} \in \uparrow)\ \textsc{case}\ \{\textsc{nom}|\textsc{acc}|\textsc{voc}\})=_c +$
$((\textsc{adj} \in \uparrow)\ \textsc{case gen})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case dat})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case inst})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case loc})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{gend})=_c \textsc{n}$
$((\textsc{adj} \in \uparrow)\ \textsc{num})=_c \textsc{sg}$

(3.84) jakiego A $(\uparrow \textsc{pred})=\text{`}\textsc{what}\text{'}$
$((\textsc{adj} \in \uparrow)\ \textsc{case gen})=_c +$
$((\textsc{adj} \in \uparrow)\ \textsc{case nom})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case acc})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case dat})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case inst})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case loc})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{case voc})=-$
$((\textsc{adj} \in \uparrow)\ \textsc{gend})=_c \textsc{n}$
$((\textsc{adj} \in \uparrow)\ \textsc{num})=_c \textsc{sg}$

The sentence in (3.85) is a modified version of (3.79), where the modifier was removed (and word order was changed accordingly). The f-structure in (3.86) would be built for *emu* in (3.85):

[18] The solution proposed in Dalrymple *et al.* 2009 involves defining equations related to CASE in the lexical entries of nominals as well as in case assignment statements used by the verb. However, it could be adapted so as to use constraining case assignment statements placed on the verb, though this is not discussed here.

[19] See § 2.2.3 (especially (2.34) and below) for discussion of inside-out paths such as those used in (3.83)–(3.84).

(3.85)

Widzisz	i	boisz	się	emu?
see	and	fear	REFL	emu.ACC/GEN

‘Do you see emu and are you afraid of emu?’

(3.86)

$$\begin{bmatrix} \textsc{pred} & \text{‘}\textsc{emu}\text{’} \\ \textsc{case} & \begin{bmatrix} \textsc{acc} & + \\ \textsc{gen} & + \end{bmatrix} \end{bmatrix}$$

The f-structure in (3.86) is well-formed: it describes *emu* as specified positively both for the accusative and genitive case required by the respective predicates in (3.85).

However, when modifiers are used, as in (3.79), the sentence becomes ungrammatical due to the fact that modifiers impose more restrictive case requirements than the lexical entry of *emu* which allows any case marking. This is ensured by the lexical entries of relevant forms of modifiers: in (3.83) the modifier allows its head to be specified positively for three case values (nominative, accusative and vocative), while it imposes negative specification for the remaining values, which include genitive. By contrast, the lexical entry of the modifier in (3.84) only allows genitive as the case of its head, while all other cases are specified negatively. This is why the sentence in (3.79) is ungrammatical with either modifier because the requirements of either modifier are inconsistent with joint requirements of the two verbs.

For the time being, the simpler analysis of case (where it takes atomic values) will be assumed, though.

3.3 Summary

This chapter provided some basic information about two fundamentals of the syntax of Polish, namely subject-verb agreement (see § 3.1), covering not only full agreement, but also default agreement and single conjunct agreement, and structural case assignment (§ 3.2) to the subject and object, focusing on the latter, including genitive of negation (also in verb chains). Finally, alternative representations of case were briefly discussed in § 3.2.3.3.

Part II

Beyond basic coordination

Chapter 4

Coordination of unlike categories

4.1 Introduction[1]

So far, only coordination of roughly identical categories has been taken into account – the only exception is (3.32) from § 3.1.3, where a noun is coordinated with a numeral phrase, but both are ultimately treated as nominal phrases. Same category coordination may be handled in LFG using a rule such as (2.64) from § 2.5, repeated in (4.1):

(4.1) $\begin{array}{lllll} \mathrm{XP} & \rightarrow & \mathrm{XP} & \mathrm{Conj} & \mathrm{XP} \\ & & \downarrow\in\uparrow & & \downarrow\in\uparrow \end{array}$ (4.2) $\begin{array}{lllll} \mathrm{NP} & \rightarrow & \mathrm{NP} & \mathrm{Conj} & \mathrm{NP} \\ & & \downarrow\in\uparrow & & \downarrow\in\uparrow \end{array}$

XP is a variable which stands for any category: once XP is substituted with NP, the rule in (4.2) results, whereby an NP may be coordinated with another NP to yield an NP. An example which uses this rule (together with relevant feature resolution templates, not presented here)[2] for NP coordination is provided below:

(4.3) Idą Jan i Marysia.
walk.PL Jan.SG.NOM and Marysia.SG.NOM
'Jan and Marysia walk.'

As indicated by agreement triggered on the verb, the coordinate phrase is treated as a plural NP while its conjuncts are singular. Same category coordination such as in (4.3) is very common and the rule in (4.1) provides a convenient generalisation.

However, there are well-known examples where elements belonging to different categories are coordinated. For English, such examples were discussed, *inter alia*, in Sag *et al.* 1985:

(4.4) That was a rude remark and in very bad taste. (Sag *et al.* 1985, p. 117, ex. (2e))

(4.5) Pat became a republican and quite conservative. (Sag *et al.* 1985, p. 142, ex. (67a))

(4.6) *Tracy has become a republican and of the opinion that we must place nuclear weapons in Europe. (Sag *et al.* 1985, p. 142, ex. (67b))

[1] This chapter is based on the following papers: Patejuk and Przepiórkowski 2012a, Przepiórkowski and Patejuk 2012, Patejuk and Przepiórkowski 2014a.

[2] See Dalrymple and Kaplan 2000 for discussion of feature resolution in LFG; see Przepiórkowski *et al.* 2002, especially § 9.1.4, for discussion of feature resolution in Polish (together with HPSG formalisation).

These examples show that it is possible to coordinate a noun phrase (*a rude remark*) and a prepositional phrase (*in very bad taste*) as dependents of BE, as in (4.4), or a noun phrase (*a republican*) and an adjectival phrase (*quite conservative*) as dependents of BECOME, see (4.5). However, (4.6), where a noun phrase (*a republican*) is coordinated with a prepositional phrase (*of the opinion that we must place nuclear weapons in Europe*) as dependents of BECOME, is ungrammatical. This contrast shows that coordination of unlike categories is not universally acceptable – it depends instead on the head (BE as opposed to BECOME), more precisely on its lexicalised subcategorisational requirements. Analyses of examples such as the ones cited above were provided in the following selected papers set in various non-transformational formalisms: Sag *et al.* 1985 (GPSG),[3] Bayer 1996 (Categorial Grammar) and Sag 2002 and Chaves 2006 (HPSG).

LFG seems to be particularly well suited for handling the issue of coordination of unlikes due to the fact that it employs various levels of representation, including c-(onstituent) structure and f-(unctional) structure. While particular conjuncts may belong to different categories at c-structure, they may still correspond to the same grammatical function at f-structure.

For Peterson 2004, p. 650, the identity of grammatical function is the main condition ruling coordination: “The genuine generalization is expressed in functional rather than categorial terms; it is equivalence of grammatical function rather than of syntactic category which determines whether two items can be conjoined.”[4] Following this idea, Peterson 2004, p. 652, offers a simple rule for handling unlike category coordination:[5]

$$
(4.7) \qquad \begin{array}{lllll} \mathrm{X} & \rightarrow & \mathrm{X} & \mathrm{C} & \mathrm{Y} \\ & & \downarrow\in\uparrow & & \downarrow\in\uparrow \end{array}
$$

Perhaps it was the intention of Peterson 2004 that coordination in (4.7) is endocentric in that the mother category is the same as one of the conjuncts. For example, if an NP is coordinated with some other phrase, the resulting coordinate phrase would be treated categorially as an NP. An alternative version of (4.7), which does not make such assumptions, is provided in (4.8), where, unlike in (4.1), XP is a category name,[6] not a variable:

$$
(4.8) \qquad \begin{array}{lllll} \mathrm{XP} & \rightarrow & \mathrm{YP} & \mathrm{Conj} & \mathrm{ZP} \\ & & \downarrow\in\uparrow & & \downarrow\in\uparrow \end{array}
$$

Whichever version is preferred, the functional annotation of conjuncts remains the same and it yields a feature structure which consists of a set containing the f-structures of particular conjuncts and possibly some attributes of the coordinate structure itself (e.g. annotation of the conjunction or, if applicable, resulting from feature resolution templates). Subsequently, such an item may be assigned a grammatical function and constraints may be imposed on this structure.

Though, as shown above, there has been some discussion of unlike category coordination in the framework of LFG, also in the context of the repertoire of available grammatical functions (Dalrymple and Lødrup 2000, Alsina *et al.* 2005, see § 4.2.2), no formalisation showing how

[3] *Generalized Phrase Structure Grammar*; Gazdar *et al.* 1985

[4] It is shown in ch. 5, which discusses the coordination of different grammatical functions, that this claim is too strong for some languages.

[5] Though typically the category C is used for complementisers, in (4.7) it corresponds to the conjunction.

[6] It could also be UP as in Unlike Phrase.

constraints are imposed in this environment has been provided. It is, therefore, the aim of this chapter to fill this gap by presenting how unlike category coordination phenomena can be handled formally in the framework of LFG on the basis of data from Polish, taking interactions with various complex phenomena (such as structural case assignment and control) into account.

Polish seems to be a good basis for such a task since unlike category coordination is a fairly common phenomenon in Polish. Thanks to the existence of the valence dictionary Walenty (see ch. 8 for a detailed discussion), which explicitly accounts for the possibility of coordination within one argument position, it is possible to estimate the relative frequency of such coordination. The version of 25/10/2014 contains 52167 schemata, 6927 out of which contain at least one argument which allows coordination of different categories, which amounts to over 13% of all schemata.[7] Furthermore, Walenty provides authentic examples supporting particular valence requirements, so it can serve as a rich source of examples involving unlike category coordination, some of which are used in this work (especially in ch. 8).

4.2 Grammatical function assignment

4.2.1 Prototypical, without coordination

The following general rules describe the prototypical assignment of grammatical functions in Polish LFG (note that it does not take coordination into consideration):

- subject (SUBJ) is the argument which can trigger agreement with the verb (it does not have to do so, however, as with gerunds, participles, infinitives, impersonal forms and non-agreeing subjects of finite forms; see § 3.1 for discussion);
- direct object (OBJ) is the argument which can passivise (which can become the subject under passive voice), regardless of its category or case marking;
- indirect object is the nominal object which cannot passivise; since there can be more than one such an object, they are assigned different grammatical functions according to case:
  - OBJ_θ: dative case,[8]
  - OBL_{str}: structural case (see § 3.2.3),
  - OBL_{gen}: genitive case (lexical),
  - OBL_{inst}: instrumental case;
- oblique object (OBL) is the prepositional argument; if there is more than one such argument, an appropriate numerical index is appended to the name of the grammatical function (OBL2, etc.); there is also a range of semantic obliques (such as adlative OBL-ADL, etc. – see § 8.2.6);

[7] It must be noted, however, that this is the frequency of schemata allowing unlike category coordination in Walenty, rather than relative textual frequency of this phenomenon.

[8] While OBJ_θ was originally (Bresnan 1982a, Dalrymple 2001) defined in LFG as a "semantically restricted" grammatical function (it is characterised by specific semantics, unlike OBJ which is unrestricted semantically), in Polish LFG this function is used exclusively for dative objects (though such arguments usually do have special semantics: typically a beneficiary).

- closed complement (COMP) is the clausal argument, usually finite, which is self-contained with respect to argument realisation – all its arguments are realised independently of other predicates (see § 2.2.1 for discussion);

- open complement (XCOMP) is the clausal argument, typically infinitival, whose subject must be structure-shared with some argument of the higher predicate (this is determined by the controller–controllee relation introduced lexically by the predicate which takes an open complement, see § 2.2.1 and § 8.2.5 for discussion);

- predicative open complement (XCOMP-PRED) is a specialised instance of an open complement: it is a predicative item, regardless of its category: it can be a nominal (noun, gerund), an adjectival form (adjective, adjectival participle) or a preposition.

4.2.2 Coordination issue

While the assignment of grammatical functions such as presented above is unproblematic in simple, "prototypical" cases, it turns out to be imperfect when coordination of unlike categories is taken into account. When, as in (4.9), a prepositional phrase (*o nazwach roślin*; OBL in § 4.2.1) is coordinated with a closed clausal complement (*komu i czemu zostały poświęcone*; COMP in § 4.2.1), which grammatical function is appropriate for the entire coordinate phrase?

(4.9)

zostaną	uroczyście	odsłonięte	tabliczki	informujące	[o	nazwach	roślin]	oraz
will be	solemnly	unveiled	plate	informing	about	names.ACC	plants.GEN	and

[komu	i	czemu	zostały	poświęcone]
who.DAT	and	what.DAT	were	devoted

'During the ceremony plates informing about plants' names and to whom and to what they were devoted will be unveiled.' (NKJP)

The assumption is that elements of a coordinate phrase usually[9] correspond to the same grammatical function, so the entire coordinate phrase should be assigned one common grammatical function. Drawing on the suggestions and conclusions of the so-called "OBJ vs COMP" debate (Dalrymple and Lødrup 2000, Alsina *et al.* 2005), it was decided to assign the OBL grammatical function in such situations, treating COMP as an elsewhere grammatical function, assigned when the only possible realisation of a given argument is clausal. Similar problems arise when other categories corresponding prototypically to different grammatical functions are coordinated – again, a common grammatical function must be chosen. Additional problems arise when a category corresponding to a closed grammatical function is coordinated with a category which corresponds to an open grammatical function – the problem is how to establish control relations properly (see § 4.7 for a discussion of this problem and a potential solution).

4.2.3 Passivisation issue

Another problem arises for the grammatical function assignment procedure presented above when coordination of various objects is taken into consideration, as in the examples below:

[9] With the exception of lexico-semantic coordination discussed in ch. 5.

(4.10) Marek manipuluje i wysługuje się Marysią.
Marek.NOM manipulates and lackey REFL Marysia.INST
'Marek manipulates and lackeys Marysia.'

(4.11) Marysia lubi ale też boi się Marka.
Marysia.NOM likes but also be afraid REFL Marek.ACC/GEN
'Marysia likes but at the same time is afraid of Marek.'

(4.10) is an instance of coordination of verbs which mark their object for instrumental, a lexical case. It shows that dependents of verbs with different passivisation capabilities can be shared under coordination: since the object of the verb MANIPULOWAĆ 'manipulate' can become the subject under passive voice, it would correspond to the OBJ grammatical function following rules presented in § 4.2.1, but this is not possible with the object of the verb WYSŁUGIWAĆ SIĘ 'lackey', so it would be assigned a different grammatical function: OBL_{inst}. This demonstrates that passivisation should not be the decisive factor in the process of assigning a grammatical function to objects, because elements of a coordinate phrase are assigned the same[10] grammatical function. When discussing other problems with passivisation as a criterion for distinguishing the OBJ grammatical function, Börjars and Vincent 2008, p. 155 come to a similar conclusion: "We would argue that there is evidence that the passive is conditioned not just by grammatical relations, but also by a complex interaction between structural position and semantics and hence is not a reliable test for a grammatical relation."

In (4.11), as in (4.10), the object of one of the conjuncts, LUBIĆ 'like', is passivisable, so it would correspond to the OBJ grammatical function, unlike the object of the other conjunct, BAĆ SIĘ 'fear, be afraid' (passivisation is not possible when the reflexive marker SIĘ is used), which requires the lexical genitive case, so it would bear a different grammatical function, OBL_{gen}.[11]

Nevertheless, this work ignores such problems and assumes that OBJ is defined on the basis of passivisation.

4.3 Imposing constraints: intuitions, issue

After a grammatical function is selected for the syntactic position (taking all its possible realisations into consideration), constraints are imposed for each realisation depending on its syntactic category and its properties: nominals bear case, prepositions have a certain form and select a nominal marked for a particular case, clauses are of a certain type.

Let us start with a fairly simple example from Kosek 1999, where a coordinate phrase consisting of a prepositional phrase and a nominal phrase serves as an argument:

(4.12) Owinął dziecko w koc i ręcznikiem.
wrapped baby in blanket.ACC and towel.INST
'He wrapped the baby in a blanket and with a towel.' (Kosek 1999, p. 43, ex. (8))

[10] Again, with the exception of lexico-semantic coordination (see ch. 5), which is not relevant here.

[11] Furthermore, the two coordinated verbs in (4.11) impose different case constraints on their object: LUBIĆ assigns structural case, realised in (4.11) as the accusative due to the absence of sentential negation, while BAĆ SIĘ assigns its object the lexical genitive case. To see how this could be handled in LFG, see the discussion of (3.74) in § 3.2.3.3.

This example shows that the verb OWINĄĆ ‘wrap’ requires an argument which may be realised as a prepositional phrase containing the preposition W and a noun phrase marked for the accusative case (*w koc*), or as a nominal phrase marked for the instrumental case (*ręcznikiem*). According to the rules of grammatical function assignment presented in § 4.2, the prepositional conjunct would be an oblique (OBL), while the nominal conjunct would be an object requiring the instrumental case (OBL_{inst}). Let us assume that the common grammatical function is the former – OBL.

The f-structure representing (4.12) is provided in (4.13) below:[12]

(4.13)

$$\left[\begin{array}{lll}
\text{PRED} & \multicolumn{2}{l}{\text{‘WRAP}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{’}} \\
\text{SUBJ} & \boxed{1} & \left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right] \\
\text{OBJ} & \boxed{2} & \left[\begin{array}{ll}\text{PRED} & \text{‘BABY’}\end{array}\right] \\
\text{OBL} & \boxed{3} & \left[\begin{array}{l}
\left\{\left[\begin{array}{ll}\text{PRED} & \text{‘BLANKET’} \\ \text{CASE} & \text{ACC} \\ \text{PFORM} & \text{W}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{‘TOWEL’} \\ \text{CASE} & \text{INST}\end{array}\right]\right\} \\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]
\end{array}\right]$$

Constraints satisfied by relevant conjuncts are provided below:

(4.14) $(\uparrow \text{OBL PFORM}) =_c \text{W} \;\wedge\; (\uparrow \text{OBL CASE}) =_c \text{ACC}$

(4.15) $(\uparrow \text{OBL CASE}) =_c \text{INST}$

Intuitively, a disjunction of constraints provided in (4.14)–(4.15), shown in (4.16), should allow for sentences where the oblique is realised as only one of the conjuncts used in (4.12) or as the entire coordinate phrase.

(4.16) $(\uparrow \text{OBL CASE}) =_c \text{INST} \;\vee$
$[(\uparrow \text{OBL PFORM}) =_c \text{W} \;\wedge\; (\uparrow \text{OBL CASE}) =_c \text{ACC}]$

This is not the case, however, given standard LFG assumptions about the interpretation of disjunctions applied to coordinate structures, under which one disjunct is chosen and it is applied to the entire coordinate structure (instead of evaluating the disjunction for each element of the coordinate structure separately). As a result, if the constraint in (4.16) is used, only sentences with non-coordinate obliques are accepted or sentences where identical obliques are used. The following examples illustrate the latter case: in (4.17) there are two nominal phrases marked for the instrumental case, while in (4.18) there are two prepositional phrases both of which feature the preposition W and a nominal marked for the accusative case:

(4.17) Owinął dziecko kocem i ręcznikiem.
wrapped baby blanket.INST and towel.INST

(4.18) Owinął dziecko w koc i w ręcznik.
wrapped baby in blanket.ACC and in towel.ACC

[12] The preposition W in (4.12) is non-semantic (see § 2.3), so it is analysed in (4.13) as a co-head: the preposition contributes the attribute PFORM, which stores the form of the preposition, while the accompanying noun phrase contributes the semantic form (PRED) and relevant attributes.

If CASE is treated as a non-distributive feature (see the discussion below (2.71) in § 2.5), $(\uparrow \text{OBL CASE}) =_c \text{INST}$, the first disjunct of (4.16) defined in (4.15), would not reach individual conjuncts – it would only apply to the coordinate structure itself. However, it is dubious whether a coordination of unlike categories such as in (4.12) should bear a resolved CASE attribute and if it does not (as in (4.13)), the constraint in (4.15) cannot be satisfied.

By contrast, if CASE is a distributive feature, the constraint in (4.15), the first disjunct of (4.16), distributes to each conjunct, so that each element of a coordinate structure is required to bear the instrumental case. While the second conjunct in (4.12), *ręcznikiem*, satisfies this constraint, the first one, *w koc*, does not – as shown in (4.13), the CASE attribute in the conjunct corresponding to the prepositional phrase is the case of the nominal, namely the accusative, which conflicts with the requirement imposed by (4.15).

When (4.14), the second disjunct of (4.16), is used, the same problems as described above result depending on whether CASE is treated as a distributive or non-distributive feature; the only difference is that (4.14) requires the accusative case. Similar issues arise for the PFORM attribute – if it is distributive, it fails for the non-prepositional conjunct (*ręcznikiem*); if it is non-distributive, only the f-structure containing conjuncts is checked for this feature, but not particular conjuncts themselves. Again, it seems wrong to assign a resolved PFORM value for unlike category coordination (such as in (4.12)).

Summing up, constraints such as (4.16) can handle non-coordinate phrases and coordinate phrases whose conjuncts display strong parallelism, but they fail immediately when unlike category coordination is taken into consideration.

This problem is caused by the fact that disjunction is interpreted too early: while the intended meaning of (4.16) is that under coordination each conjunct should independently satisfy either the first disjunct, (4.15), or the second one, (4.14), its current LFG interpretation is that all conjuncts must satisfy its first disjunct or all must consistently satisfy the other one. This is how this contrast may be formalised[13] (A stands for (4.15), B for (4.14) and GF for OBL):

(4.19) a. $\forall x \in (\uparrow \text{GF})[A(x) \vee B(x)]$ (intended)

b. $\forall x \in (\uparrow \text{GF})A(x) \vee \forall x \in (\uparrow \text{GF})B(x)$ (actual)

The interpretation formalised in (4.19b) corresponds to the current effect of (4.16), whereby one of the disjuncts is chosen (A or B) and it is applied to all elements of the relevant f-structure. As a result, under coordination all conjuncts must satisfy the same constraint, which is equivalent to having two separate lexical entries for the given verb, one containing the constraint A, the other containing B. Such a solution is unable to account for unlike category coordination.

By contrast, (4.19a) is the formalisation of the desired effect of (4.16), whereby the disjunction is evaluated independently for each element of the relevant f-structure. As a result, under coordination each conjunct can satisfy a different disjunct of (4.16) (A or B), which makes it possible to account for unlike category coordination.

While, as explained above, simple LFG constraints fail to account for the coordination of unlikes, there is a solution which successfully employs a relatively rarely used device to this end – off-path constraints.

[13] Thanks are due to Ron Kaplan for discussing this issue and proposing the formalisation in (4.19).

4.4 Conservative solution using off-path constraints[14]

This section offers a conservative solution to the problem of unlike category coordination in LFG – it uses the formal device known as off-path constraints, which was used for handling long-distance dependencies (Dalrymple 2001) and binding anaphora (Dalrymple 1993).

4.4.1 Introduction to off-path constraints

Dalrymple 2001, p. 148, shows how off-path constraints may be applied for the purpose of restricting paths used for extraction in long-distance dependencies. Let us briefly explain how these work using simple abstract examples.

(4.20) is a constraining equation – a minimal f-structure which satisfies this constraint is provided in (4.21). Off-path constraints provide a means of restricting some path or its part through the introduction of additional constraints which must be satisfied – the constraint defined in (4.22) makes use of this mechanism. While (4.20) is satisfied by both structures provided below, (4.21) and (4.23), (4.22) is minimally satisfied by (4.23).

(4.20) $(\uparrow \textsc{a b c}) =_c +$

(4.21) $\begin{bmatrix} \textsc{a} & \begin{bmatrix} \textsc{b} & \begin{bmatrix} \textsc{c} & + \end{bmatrix} \end{bmatrix} \end{bmatrix}$

(4.22) $(\uparrow \underset{(\leftarrow \textsc{d}) =_c \textsc{e}}{\textsc{a}} \quad \textsc{b c}) =_c +$

(4.23) $\begin{bmatrix} \textsc{a} & \begin{bmatrix} \textsc{b} & \begin{bmatrix} \textsc{c} & + \end{bmatrix} \end{bmatrix} \\ \textsc{d} & \textsc{e} \end{bmatrix}$

In (4.22) a part of the constraint provided in (4.20) is further constrained using an off-path constraint: $(\leftarrow \textsc{d}) =_c \textsc{e}$. Such constraints are easy to identify: they are placed directly below the attribute to which they apply – in (4.22) the off-path constraint is attached to the attribute A. Moreover, off-path constraints use different metavariables: horizontal arrows (‘←’, ‘→’) instead of vertical ones known from constraints presented so far.

The left arrow (‘←’) used in (4.22) stands for the structure which contains the attribute to which the constraint is attached. In order to satisfy the off-path constraint $(\leftarrow \textsc{d}) =_c \textsc{e}$ attached to the attribute A, the structure containing the attribute A must also contain the attribute D whose value is E.

The other off-path metavariable, the right arrow (‘→’), stands in turn for the value of the attribute to which it is attached. It is used in (4.24) below:

(4.24) $(\uparrow \underset{(\rightarrow \textsc{d}) =_c \textsc{e}}{\textsc{a}} \quad \textsc{b c}) =_c +$

(4.25) $\begin{bmatrix} \textsc{a} & \begin{bmatrix} \textsc{b} & \begin{bmatrix} \textsc{c} & + \end{bmatrix} \\ \textsc{d} & \textsc{e} \end{bmatrix} \end{bmatrix}$

In order to satisfy the off-path constraint $(\rightarrow \textsc{d}) =_c \textsc{e}$ attached to the attribute A, this attribute must contain the attribute D whose value is E. The minimal structure which satisfies (4.24) is given in (4.25).

[14] A solution along these lines was suggested by Mary Dalrymple after the presentation of Przepiórkowski and Patejuk 2012 at the LFG'12 conference for the purposes of handling unlike category coordination in Polish subjects. All usual disclaimers apply.

4.4.2 Handling unlike category coordination using off-path constraints

Let us now proceed to how off-path constraints may be used as a conservative solution to the issue of unlike category coordination in Polish. On the basis of example (4.12), § 4.3 discussed at length why a simple disjunctive constraint such as the one provided in (4.16) (repeated below as (4.26)) would fail to take such sentences into account.

(4.26) $$\begin{array}{l}(\uparrow \textsc{obl case}) =_c \textsc{inst} \;\vee \\ {[(\uparrow \textsc{obl pform}) =_c \textsc{w} \;\wedge\; (\uparrow \textsc{obl case}) =_c \textsc{acc}]}\end{array}$$

Off-path constraints make it possible to avoid the problem caused by the fact that under coordination the disjunction in (4.26) is interpreted too early. The desired interpretation of (4.26), where the constraint is satisfied if each member of the relevant f-structure independently satisfies any of the disjuncts, was put more formally in (4.19a). This effect may be achieved using the following off-path constraint:

(4.27) $$(\uparrow \textsc{obl} \underset{(\leftarrow \textsc{case}) =_c \textsc{inst} \;\vee\; [(\leftarrow \textsc{pform}) =_c \textsc{w} \;\wedge\; (\leftarrow \textsc{case}) =_c \textsc{acc}]}{\textsc{pred}})$$

The non-off-path part of (4.27), $(\uparrow \textsc{obl pred})$, is an existential constraint – it ensures that the oblique has a semantic form. Since all obliques have a semantic form anyway, this part is vacuous. Its real purpose is to provide an anchor for the off-path constraint. This solution relies on the interaction with distributive features – since PRED is distributive, the attached off-path constraint is passed to all elements of the oblique's f-structure.

This is how the off-path constraint used in (4.27) works: '←' in the off-path constraint attached to PRED points to the f-structure which contains PRED attribute, i.e. the value of OBL. The f-structure selected in this way, a simple f-structure or a member of a coordinate structure, must either satisfy the condition that it is specified for the instrumental case ($(\leftarrow \textsc{case}) =_c \textsc{inst}$) or it must contain a prepositional phrase consisting of the preposition W and a nominal marked for the accusative case ($[(\leftarrow \textsc{pform}) =_c \textsc{w} \wedge (\leftarrow \textsc{case}) =_c \textsc{acc}]$). This is the interpretation of (4.26) formalised in (4.19a). Unlike (4.26), (4.27) achieves the desired result regardless of whether the oblique is a coordinate structure or not.

4.5 More data, complex interactions

After discussing a simple example of unlike category coordination it is finally possible to proceed to further data. Sag *et al.* 1985 provide examples which demonstrate that it is possible to coordinate a nominal with a clause:

(4.28) Pat remembered the appointment and that it was important to be on time.
(Sag *et al.* 1985, p. 165, ex. (123a))

(4.29) That Himmler appointed Heydrich and the implications thereof frightened many observers. (Sag *et al.* 1985, p. 165, ex. (123b))

In these examples the coordinate phrase consisting of unlike categories corresponds to the object, see (4.28), or subject grammatical function, as in (4.29).

The following subsections present similar examples from Polish, focusing on the interaction between unlike category coordination and case assignment to subjects and objects. Let us start with the former.

4.5.1 Unlike category subjects

Basic principles of subject-verb agreement in Polish and case assignment to subjects were discussed in § 3.1 and § 3.2.2 – subjects of finite verbal forms can be marked for one of two following structural cases: nominative or accusative. The case of the subject has direct influence on subject-verb agreement: while subjects marked for the nominative case trigger full agreement (in number, gender and person), see (3.41) repeated as (4.30) for an example, there are subjects which bear the accusative case and require default agreement (singular person, neuter gender, third person) – this class is constituted by non-agreeing numerals (the object of the numeral is marked for the genitive case, see § 3.1.2.1 for discussion), see (3.42) repeated as (4.31) for comparison.

(4.30) Ludzie pracowali ciężko po 16-18 godzin na dobę.
man.NOM.PL.M1 worked.3.PL.M1 hard DISTR 16-18 hours for day
'People worked hard 16-18 hours a day.' (NKJP)

(4.31) Ośmiu ludzi pracowało po dwanaście godzin przez cztery
eight.ACC.PL.M1 man.GEN.PL.M1 worked.3.SG.N DISTR twelve hours for four
miesiące.
months
'Eight people worked twelve hours a day each for four months.' (NKJP)

Constraints which ensure appropriate structural case marking of subjects in examples provided above were first formalised in (3.18)–(3.19) in § 3.1.2.1 (see also § 3.2.2 for discussion); they are repeated below for convenience – (4.32) accounts for (4.30), while (4.33) handles (4.31).

(4.32) $\text{AGR-CASE-NOM} \equiv (\uparrow \text{SUBJ CASE}) =_c \text{NOM}$

(4.33) $\text{AGR-CASE-NUMACC} \equiv (\uparrow \text{SUBJ ACM}) =_c \text{REC} \wedge (\uparrow \text{SUBJ CASE}) =_c \text{ACC}$

There are, however, examples such as (3.32), repeated below with bracketing added in (4.34), showing that it is possible to have a subject consisting of a noun phrase marked for the nominative case and an accusative numeral phrase:

(4.34) [Pan Mirosław] i [czternastu ludzi] pracowało dzień i noc.
Mr Mirosław.NOM.SG and fourteen.ACC.PL man.GEN.PL worked.3.SG.N day and night
'Mr Mirosław and fourteen people worked night and day.' (NKJP)

The f-structure representing this sentence is provided below:

(4.35)

$$\begin{bmatrix} \text{PRED} & \text{`WORK}\langle\boxed{1}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{`Mr Mirosław'} \\ \text{CASE} & \text{NOM} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{`FOURTEEN}\langle\boxed{2}\rangle\text{'} \\ \text{OBJ} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`MAN'}\end{bmatrix} \\ \text{CASE} & \text{ACC} \\ \text{ACM} & \text{REC} \end{bmatrix} \right\} & \\ \text{COORD-FORM} \quad \text{I} & \end{bmatrix} \\ \text{ADJ} & \left\{ \begin{bmatrix} \left\{ \begin{bmatrix}\text{PRED} & \text{`DAY'}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`NIGHT'}\end{bmatrix} \right\} \\ \text{COORD-FORM} \quad \text{I} \end{bmatrix} \right\} \end{bmatrix}$$

Even though (4.34) uses the single conjunct agreement strategy (SCA, discussed in § 3.1.3), whereby the verb agrees with one of the conjuncts (in (4.34) it is the closest conjunct), every element of the phrase corresponding to the subject grammatical function must individually satisfy relevant subjecthood constraints. This is demonstrated in the example provided below:

(4.36)	*[Pana Mirosława]	i	[czternastu	ludzi]	pracowało	dzień	i	noc.
	Mr Mirosław.ACC.SG	and	fourteen.ACC.PL	man.GEN.PL	worked.3.SG.N	day	and	night

The first conjunct in (4.36) is marked for the accusative case, which results in ungrammaticality because the verb PRACOWAĆ requires appropriate structural case marking from its subject: accusative for non-agreeing numerals, and nominative otherwise. Intuitively, (4.37) (repeated from (3.17)), a disjunction of the constraints provided in (4.32)–(4.33), should achieve this goal:

(4.37) $\text{AGR-CASE} \equiv \text{AGR-CASE-NOM} \vee \text{AGR-CASE-NUMACC}$

However, as explained at length in § 4.3, such a constraint would make correct predictions about sentences with non-coordinate subjects (such as (4.30) and (4.31)) or coordinate subjects were every conjunct is specified for the same case, but it would fail if the subject is a coordinate phrase consisting of elements bearing distinct case values, as in (4.34).

To avoid the issue related to the fact that the disjunction in (4.37) is understood too early (see the discussion of (4.19) in § 4.3), the subjecthood constraint holding for verbs taking structurally case-marked subjects can be formalised using off-path constraints – (4.38) is an off-path version of (4.37). In order to make it easier to find corresponding[15] templates, off-path templates defined in (4.39)–(4.41) have -OFFPATH suffix added to the names of respective plain templates (so that (4.40) is an off-path counterpart of (4.32)). (4.40)–(4.41) contain definitions of templates called by template AGR-CASE-OFFPATH[16] defined in (4.39), while (4.42) is a fully expanded version of (4.38).

(4.38) $(\uparrow \text{SUBJ} \quad \underset{\text{AGR-CASE-OFFPATH}}{\text{PRED}} \quad)$

(4.39) $\text{AGR-CASE-OFFPATH} \equiv \text{AGR-CASE-NOM-OFFPATH} \vee \text{AGR-CASE-NUMACC-OFFPATH}$

[15] It must be noted, however, that off-path templates cannot be interpreted without the constraint which is their anchor – for templates defined in (4.39)–(4.41) the anchor is the first line of (4.38), (↑ SUBJ PRED).

[16] While in LFG grammars implemented in XLE it is not possible to use templates in off-path constraints, there seems to be no such restriction in theoretical LFG. Therefore, this work uses templates in off-path constraints in order to improve the readability of complex constraints by making it possible to easily refer to relevant fragments.

(4.40) $\textsc{agr-case-nom-offpath} \equiv (\leftarrow \textsc{case}) =_c \textsc{nom}$

(4.41) $\textsc{agr-case-numacc-offpath} \equiv (\leftarrow \textsc{acm}) =_c \textsc{rec} \wedge (\leftarrow \textsc{case}) =_c \textsc{acc}$

(4.42) $$(\uparrow \textsc{subj} \underset{(\leftarrow \textsc{case}) =_c \textsc{nom} \ \vee \ [(\leftarrow \textsc{acm}) =_c \textsc{rec} \ \wedge \ (\leftarrow \textsc{case}) =_c \textsc{acc}]}{\textsc{pred}})$$

As in (4.27), in (4.38) the PRED attribute serves as an anchoring point for the off-path constraint in (4.39) because PRED is a distributive feature according to LFG assumptions. As a result, the off-path constraint in (4.39), the disjunction of two other templates, (4.40) and (4.41), attached to PRED attribute, is distributed to all elements of the relevant structure – the off-path version of the structural case assignment statement (4.37) is evaluated separately for each element of the subject f-structure. This makes it possible for (4.38) to account for non-coordinate sentences such as (4.30) and (4.31), sentences with coordinate subjects which have identical specification (two nominative NPs or two accusative numeral phrases, for instance) and, finally, for sentences with a coordinate subject bearing non-identical specifications, as in (4.34).

The statement provided in (4.42) may be used as the default subjecthood constraint for finite verbs taking structurally case-marked subjects.

Let us now see how such statements may be used to account for more complex examples where the subject consists of a nominal phrase bearing structural case and a clause:[17]

(4.43) Jana dziwiło, [że Maria wybiera Piotra], i [jej brak gustu].
Jan.ACC puzzled.3.SG.N that Maria chooses Piotr and her lack.NOM.SG.M3 taste
'(The fact) that Maria prefers Piotr and her lack of taste puzzled Jan.'
(Świdziński 1992, 1993)

This example is similar to the English (4.29), but it is considerably more challenging in that structural case assignment must be taken into consideration. The f-structure representation of (4.43) is provided below:[18]

(4.44)
$$\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{puzzle}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\textsc{subj} & \boxed{1}\left[\begin{array}{l}
\left\{\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{choose}\langle\boxed{3},\boxed{4}\rangle\text{'} \\
\textsc{subj} & \boxed{3}\left[\textsc{pred}\ \ \text{`}\textsc{Maria}\text{'}\right] \\
\textsc{obj} & \boxed{4}\left[\textsc{pred}\ \ \text{`}\textsc{Piotr}\text{'}\right] \\
\textsc{comp-form} & \textsc{że}
\end{array}\right],
\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{lack\_of\_taste}\text{'} \\
\textsc{case} & \textsc{nom}
\end{array}\right]\right\} \\
\textsc{coord-form} \quad \textsc{i}
\end{array}\right] \\
\textsc{obj} & \boxed{2}\left[\textsc{pred}\ \ \text{`}\textsc{Jan}\text{'}\right]
\end{array}\right]$$

As in the case of example (4.34), (4.43) uses the SCA agreement strategy – the verb picks the clause, the closest conjunct, as the agreement controller and therefore it appears in the default, non-agreeing form (third person, singular number, neuter gender). If the verb agreed with the second conjunct in (4.43), a different verb form would be required, as shown in (4.45):

[17] Sag 2002, p. 277, fn. 10, discussed a slightly different version of (4.43).

[18] The f-structure fragment corresponding to *jej brak gustu* 'her lack of taste' in (4.44) is simplified as its internal structure is irrelevant to the discussion.

(4.45) Jana dziwił/*dziwiło jej brak gustu.
Jan.ACC puzzled.3.SG.M3/N her lack.NOM.SG.M3 taste
'Her lack of taste puzzled Jan.'

In spite of the fact that the verb in (4.43) agrees with the closest conjunct, the nominal conjunct must still be marked for an appropriate case:

(4.46) *Jana dziwiło, [że Maria wybiera Piotra], i [jej brakiem gustu].
Jan.ACC puzzled.3.SG.N that Maria chooses Piotr and her lack.INST.SG.M3 taste

According to the constraint defined in (4.42), the nominal conjunct BRAK 'lack' should be marked for the nominative case (the accusative is restricted to non-agreeing numerals exclusively), as in (4.43) (see (4.44) for the corresponding f-structure), instead of the instrumental used in (4.46).

For the verb DZIWIĆ 'puzzle, surprise' to allow for a non-canonical clausal subject, its lexical entry would be required to contain the following constraint as an additional disjunct:

(4.47) (↑ SUBJ COMP-FORM) $=_c$ ŻE

This constraint allows for sentences whose subject is clausal (coordinate or non-coordinate) as long as the appropriate complemeniser form is used. The complementiser type required lexically by DZIWIĆ is of the ŻE class (i.e. *że* or *iż*) – it must be the value of the COMP-FORM attribute. The following sentences satisfy the constraint in (4.47):

(4.48) Naszych gości dziwiło, że mamy tak dużo obowiązków.
our guests.ACC puzzled.3.SG.N that have so many duties
'(The fact) that we have so many duties puzzled our guests.' (NKJP)

(4.49) Może tylko trochę dziwiło, [że jego firma nie ma strony internetowej]
maybe only little puzzled.3.SG.N that SELF company NEG has page Internet
i [że zdjęcia swoich prac stolarskich przesyła drogą elektroniczną].
and that photos his works carpenter's sends way electronic
'Maybe (the fact) that his company has no website and that he sends photos of his carpentry works by e-mail surprised a little.' (NKJP)

In order to take into account unlike category coordination featuring clausal subjects such as in (4.43), an off-path version of the constraint provided in (4.47) needs to be added to the off-path constraint handling structural case assignment for subjects shown in (4.38) – the constraint in (4.50) results; see (4.51) for its expanded version:

(4.50) (↑ SUBJ PRED)
AGR-CASE-OFFPATH ∨ (← COMP-FORM) $=_c$ ŻE

(4.51) (↑ SUBJ PRED)
(← CASE) $=_c$ NOM ∨ [(← ACM) $=_c$ REC ∧ (← CASE) $=_c$ ACC]
∨ (← COMP-FORM) $=_c$ ŻE

The first disjunct of AGR-CASE-OFFPATH, (← CASE) $=_c$ NOM, allows for nominative case-marked elements, while the second one, [(← ACM) $=_c$ REC ∧ (← CASE) $=_c$ ACC], takes care of non-agreeing numerals marked for the accusative case. Finally, (← COMP-FORM) $=_c$ ŻE, the off-path version of (4.47), is dedicated to clauses containing a ŻE-type complementiser.

It must be emphasised that subjecthood requirements imposed by particular verbs are highly lexicalised: while most verbs accept structurally case-marked subjects, only some selected verbs may take a clausal subject. Such information is stored in the lexical entries of relevant verbs: for instance, forms of PRACOWAĆ would contain the constraint defined in (4.38), which only allows for the subject to be marked for structural case (as in (4.30), (4.31) and (4.34)), while forms of DZIWIĆ would use instead the constraint provided in (4.50), which additionally allows for clausal subjects (as in (4.43), (4.48) and (4.49)).

4.5.2 Unlike category objects

Let us now proceed to unlike category objects, starting with an example from the literature (bracketing, glosses and translation were added); its f-structure is given in (4.53).

(4.52) Doradził mu [wyjazd] i [żeby nie wracał].
advised him departure.ACC and that NEG come back
'He advised him to leave and not to come back.' (Kallas 1993, p. 92, ex. (48a))

(4.53)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`ADVISE}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\
\text{OBJ} & \boxed{2}\left[\begin{array}{l} \left\{\begin{array}{l} \left[\begin{array}{ll}\text{PRED} & \text{`LEAVE'}\\ \text{CASE} & \text{ACC}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`COME\_\_BACK}\langle\boxed{4}\rangle\text{'}\\ \text{SUBJ} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right]\\ \text{COMP-FORM} & \text{ŻEBY}\\ \text{NEG} & +\end{array}\right]\end{array}\right\} \\ \text{COORD-FORM} \quad \text{I}\end{array}\right] \\
\text{OBJ}_{\theta} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`HE'}\\ \text{CASE} & \text{DAT}\end{array}\right]
\end{array}\right]$$

Now, let us consider another example, based on (4.52) – its f-structure is provided in (4.55).

(4.54) (Wcale) nie doradził mu [wyjazdu] ani [żeby nie wracał].
not at all NEG advised him departure.GEN nor that NEG come back
'He did not advise him to leave nor not to come back.'

(4.55)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`ADVISE}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\
\text{OBJ} & \boxed{2}\left[\begin{array}{l} \left\{\begin{array}{l} \left[\begin{array}{ll}\text{PRED} & \text{`LEAVE'}\\ \text{CASE} & \text{GEN}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`COME\_\_BACK}\langle\boxed{4}\rangle\text{'}\\ \text{SUBJ} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right]\\ \text{COMP-FORM} & \text{ŻEBY}\\ \text{NEG} & +\end{array}\right]\end{array}\right\} \\ \text{COORD-FORM} \quad \text{ANI}\end{array}\right] \\
\text{OBJ}_{\theta} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`HE'}\\ \text{CASE} & \text{DAT}\end{array}\right] \\
\text{NEG} & +
\end{array}\right]$$

In both examples the object is a coordinate phrase which consists of a nominal (a form of the noun WYJAZD 'departure') and a clause (*żeby nie wracał* 'not to come back'). However, these examples differ consistently depending on whether negation (the negation marker NIE) is present – in (4.52) the nominal conjunct takes the accusative case, while in (4.54) it is marked for the genitive case. As discussed in § 3.2.3, such variation is attributable to the fact that the verb DORADZIĆ 'advise' may take a structurally case-marked object: its case marking depends on the syntactic environment, namely on the part of speech of the head assigning case and on the availability of sentential negation.

Taking only local case assignment to objects into consideration,[19] its principles could be summarised as follows: when sentential negation is not available, the object is marked for the accusative case (as in (4.52); see its f-structure in (4.53)), when it is available, the object bears the genitive case (see (4.54), the corresponding f-structure is given in (4.55)).

The constraint handling local case assignment to objects was provided in (3.57) in § 3.2.3.1 and it is repeated below:

(4.56) $\textsc{strcase} \equiv [\textsc{affirmative} \ \vee \ \textsc{negative}]$

(4.57) $\textsc{affirmative} \equiv [\neg(\uparrow \textsc{neg}) \ \wedge \ (\uparrow \textsc{obj case}) =_c \textsc{acc}]$

(4.58) $\textsc{negative} \equiv [(\uparrow \textsc{neg}) =_c + \ \wedge \ (\uparrow \textsc{obj case}) =_c \textsc{gen}]$

To allow for clausal objects in (4.52) and (4.54), a constraint checking that the object contains the complementiser of appropriate type (ŻEBY, realised by forms such as *żeby*, *by*, *aby*, *ażeby* and *iżby*) must be added:

(4.59) $(\uparrow \textsc{obj comp-form}) =_c \textsc{żeby}$

Intuitively, (4.60), a disjunction of constraints defined in (4.56) and (4.59), should account for unlike category coordination in (4.52) and (4.54):

(4.60) $\textsc{strcase} \ \vee \ (\uparrow \textsc{obj comp-form}) =_c \textsc{żeby}$

This is not the case, however, because, again, as discussed in § 4.3, the disjunction is understood too early and, rather than being resolved for each element of the relevant f-structure (here: the object) individually, it is resolved once and applied to all elements. As a result, it handles correctly simple cases (no coordination, coordination of elements bearing the same specification), but it cannot account for unlike category coordination in (4.52) and (4.54). In order to achieve the effect of evaluating the disjunction in (4.60) individually for each conjunct under coordination, the constraint in (4.60) must be rewritten so as to use off-path constraints. Let us start with its first disjunct, STRCASE – examples (4.61)–(4.64) show how STRCASE defined in (4.56) should be rewritten so as to use off-path constraints (see (4.65) for the fully expanded version of (4.61)).

(4.61) $(\uparrow \textsc{obj} \quad \underset{\textsc{strcase-offpath}}{\textsc{pred}} \quad)$

(4.62) $\textsc{strcase-offpath} \equiv [\textsc{affirmative-offpath} \ \vee \ \textsc{negative-offpath}]$

[19] See § 3.2.3 for a more detailed discussion of principles of structural case assignment to objects in Polish.

(4.63) $$\textsc{affirmative-offpath} \equiv [\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{acc}]$$

(4.64) $$\textsc{negative-offpath} \equiv [((\textsc{obj} \leftarrow)\ \textsc{neg}) =_c +\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{gen}]$$

(4.65) $$(\uparrow \textsc{obj} \underset{\substack{[\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{acc}]\ \vee \\ [((\textsc{obj} \leftarrow)\ \textsc{neg}) =_c +\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{gen}]}}{\textsc{pred}})$$

(4.61) is placed in the lexical entry of the relevant verb assigning structural case to its object. The off-path constraint is attached to (↑ OBJ PRED), a simple existential constraint ensuring that the verb's object has a semantic form. Since the PRED attribute is distributive, the off-path constraint STRCASE-OFFPATH attached to it is distributed to potential conjuncts of the object.

Let us start with the first disjunct of (4.62): it is a call to the template AFFIRMATIVE-OFFPATH defined in (4.63) as a conjunction of two simple constraints. The first, ¬((OBJ ←) NEG), uses a complex path: (OBJ ←). Using this inside-out path (see § 2.2.3 for discussion of this device), it points to the structure which contains the OBJ f-structure, i.e. to the f-structure of the verb, and ensures that there is no sentential negation there (as in (4.53)). The other conjunct, $(\leftarrow \textsc{case}) =_c \textsc{acc}$, checks that the object bears the accusative case (that the value of CASE of the structure which contains the attribute PRED is ACC).

The second off-path disjunct of (4.62) calls the template NEGATIVE-OFFPATH defined in (4.64). It employs the same mechanisms as described above: its first conjunct, $((\textsc{obj} \leftarrow)\ \textsc{neg}) =_c +$, is an off-path constraint with a complex inside-out path which checks that the verbal head (the f-structure which contains the OBJ attribute) is negated. If the negation marker NIE is present, it acts as the verb's co-head, contributing a positive value (+) of NEG attribute ((↑ NEG) = +) to the f-structure of the verb (as in (4.55)). The second conjunct of NEGATIVE-OFFPATH, $(\leftarrow \textsc{case}) =_c \textsc{gen}$, ensures that the object bears the appropriate case – genitive, due to the availability of sentential negation.

In order to handle examples such as (4.52) and (4.54), the constraint provided in (4.61) needs to be extended by adding one more disjunct – the off-path version of (4.59), the constraint which allows for clausal objects. This yields (4.66) (see (4.67) for its expanded version), i.e. the off-path version of the plain constraint provided in (4.60):

(4.66) $$(\uparrow \textsc{obj} \underset{\textsc{strcase-offpath}\ \vee\ (\leftarrow \textsc{comp-form}) =_c \textsc{żeby}}{\textsc{pred}})$$

(4.67) $$(\uparrow \textsc{obj} \underset{\substack{[\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{acc}]\ \vee \\ [((\textsc{obj} \leftarrow)\ \textsc{neg}) =_c +\ \wedge\ (\leftarrow \textsc{case}) =_c \textsc{gen}]\ \vee \\ (\leftarrow \textsc{comp-form}) =_c \textsc{żeby}}}{\textsc{pred}})$$

This constraint is capable of handling non-coordinate objects (either marked for structural case or clausal), simple instances of coordination (where all conjuncts bear identical specification: they are either marked for appropriate structural case or they are clauses containing the same complementiser) and, finally, the coordination of unlike objects such as in (4.52)–(4.54), requiring appropriate structural case from relevant conjuncts.

4.5.2.1 Interaction with verb chains

As explained in § 3.2.3.2, structural case assignment to objects is not strictly local in Polish: in verb chains, when negation is present in a higher clause, it may influence the case assigned by the lower predicate – in such a situation the verb assigning case may assign genitive to its object, as if negation was present locally. It may, however, assign the object the accusative case (assuming that there is no local negation). See (4.68) for an illustration of these possibilities (it was constructed on the basis of (4.52) and (4.54)):[20]

(4.68)

(Wcale)	nie	próbował	doradzić	mu	[wyjazdu/wyjazd]	ani	[żeby	nie	wracał].
not at all	NEG	tried	advise.INF	him	departure.GEN/ACC	nor	that	NEG	come back

‘He did not advise him to leave nor not to come back.’

The f-structure in (4.69) provides a representation of (4.68) where genitive of negation is optional – the object of ADVISE may be marked for the accusative or genitive case:[21]

(4.69)

$$\begin{bmatrix}
\text{PRED} & \text{`TRY}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\end{bmatrix} \\
\text{XCOMP} & \boxed{2}\begin{bmatrix}
\text{PRED} & \text{`ADVISE}\langle\boxed{1},\boxed{3},\boxed{4}\rangle\text{'} \\
\text{SUBJ} & \boxed{1} \\
\text{OBJ} & \boxed{3}\begin{bmatrix}
\left\{\begin{bmatrix}\text{PRED} & \text{`LEAVE'} \\ \text{CASE} & \text{ACC} \vee \text{GEN}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`COME\_BACK}\langle\boxed{5}\rangle\text{'} \\ \text{SUBJ} & \boxed{5}\begin{bmatrix}\text{PRED} & \text{`PRO'}\end{bmatrix} \\ \text{COMP-FORM} & \text{ŻEBY} \\ \text{NEG} & +\end{bmatrix}\right\} \\
\text{COORD-FORM} \quad \text{ANI}
\end{bmatrix} \\
\text{OBJ}_\theta & \boxed{4}\begin{bmatrix}\text{PRED} & \text{`HE'} \\ \text{CASE} & \text{DAT}\end{bmatrix}
\end{bmatrix} \\
\text{NEG} & +
\end{bmatrix}$$

To account for the optionality of the genitive of negation under transferred negation in examples such as (4.68), the definitions of templates AFFIRMATIVE-OFFPATH and NEGATIVE-OFFPATH called by STRCASE-OFFPATH (see (4.62) for its definition) in (4.66) must be modified so that they are off-path versions of corresponding templates used in (3.67) discussed in § 3.2.3.2. The result is a redefined version of STRCASE-OFFPATH in (4.71), which is used by (4.70) (see (4.78) for its expanded version):

(4.70) $$(\uparrow \text{OBJ} \quad \underset{\text{STRCASE-OFFPATH} \vee (\leftarrow \text{COMP-FORM}) =_c \text{ŻEBY}}{\text{PRED}} \quad)$$

(4.71) $$\text{STRCASE-OFFPATH} \equiv [\text{AFFIRMATIVE-OFFPATH} \ \vee \ \text{NEGATIVE-OFFPATH}]$$

[20] Some speakers find the accusative in (4.68) considerably worse than the genitive or even completely unacceptable. The variation in the acceptability of the optionality of long distance GoN among speakers of Polish is a known issue – it was discussed in Przepiórkowski 2000.

[21] The f-structure in (4.69) uses a representational shortcut, namely a disjunctive value of CASE (ACC ∨ GEN), in order to express the optionality of GoN in (4.68). As a result, (4.69) stands for two almost identical f-structures, differing only in CASE value (accusative and genitive, respectively).

(4.72) $\textsc{affirmative-offpath} \equiv [\neg((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) =_{c} \textsc{acc}]$

(4.73) $\textsc{negative-offpath} \equiv [\textsc{anyneg-offpath}\ \wedge\ \textsc{negtype-offpath}]$

(4.74) $\textsc{anyneg-offpath} \equiv ((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg}) =_{c} +$

(4.75) $\textsc{negtype-offpath} \equiv [\textsc{locneg-offpath}\ \vee\ \textsc{nonlocneg-offpath}]$

(4.76) $\textsc{locneg-offpath} \equiv [((\textsc{obj} \leftarrow)\ \textsc{neg}) =_{c} +\ \wedge\ (\leftarrow \textsc{case}) =_{c} \textsc{gen}]$

(4.77) $\textsc{nonlocneg-offpath} \equiv [\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) \in_{c} \{\textsc{acc},\ \textsc{gen}\}]$

(4.78)
$$\begin{array}{c}
(\uparrow \textsc{obj} \qquad\qquad\qquad\qquad \textsc{pred} \qquad\qquad\qquad\qquad) \\
[\neg((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) =_{c} \textsc{acc}]\ \vee \\
[((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg}) =_{c} +\ \wedge \\
[[((\textsc{obj} \leftarrow)\ \textsc{neg}) =_{c} +\ \wedge\ (\leftarrow \textsc{case}) =_{c} \textsc{gen}]\ \vee \\
[\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow \textsc{case}) \in_{c} \{\textsc{acc},\ \textsc{gen}\}]]]\ \vee \\
(\leftarrow \textsc{comp-form}) =_{c} \textsc{żeby}
\end{array}$$

Though the constraint defined in (4.70) looks the same as (4.66), their expansions are different – see (4.78) and (4.67), respectively. Since all changes apply to the definition of templates called by STRCASE-OFFPATH, only the relevant fragments are discussed below.

The template AFFIRMATIVE-OFFPATH defined in (4.72) contains a minor change with respect to the previous version defined in (4.63): $\textsc{xcomp}^{*}$ was added to the beginning of the inside-out path. As a result of this change, the off-path constraint may point not only to the f-structure of the verb which contains the OBJ attribute (ADVISE in (4.69)), but it may also point further, to the f-structure corresponding to the verb which subcategorises for the former verb as its open infinitival complement, XCOMP (TRY in (4.69)). In theory, there may be any number of subsequent infinitival clauses (including zero) – this is indicated by the use of the asterisk (*) operator to express functional uncertainty (see § 2.4 for discussion). In the path used for (4.68) one XCOMP would be used, see the f-structure in (4.69), while in (4.54) there would be none as there is no infinitival clause involved – compare the f-structure in (4.55).

Using functional uncertainty, AFFIRMATIVE-OFFPATH, the template defined in (4.72), ensures that there is no sentential negation in the relevant domain – there may neither be local negation nor negation transferred along a verb chain from some higher clause. In this situation the object is assigned the accusative case, as in sentence (4.52), represented in (4.53).

The template NEGATIVE-OFFPATH defined in (4.73) (see (4.64) for its previous version) is a complex constraint handling cases where sentential negation is available at some level (such as (4.54) and (4.68)) – this is formalised using its first conjunct, the template ANYNEG-OFFPATH defined in (4.74). Negation may be local to the predicate assigning case (the asterisk in (4.74) allows for XCOMP to be omitted), as in (4.54) represented in (4.55), but it may also be transferred from some higher predicate in a verb chain (at least one XCOMP in the path), see (4.68) and its f-structure in (4.69). The template NEGTYPE-OFFPATH, the second conjunct of (4.73), determines which case should be assigned depending on whether negation is local or only transferred.

According to its definition in (4.75), NEGTYPE-OFFPATH is a disjunction of two templates: its first disjunct, LOCNEG-OFFPATH defined in (4.76), is the same as (4.64) in the previous version of STRCASE-OFFPATH (see (4.62)) – it handles situations when negation is local to the verb assigning case and its object must bear the genitive case, see (4.54) and its f-structure in (4.55).

By contrast, the second disjunct of (4.75), the template NONLOCNEG-OFFPATH defined in (4.77), is dedicated to the case in which negation is not available locally – the verb to which the entire case assignment statement is attached must not be negated. However, for this constraint to be satisfied, the constraint provided in the template ANYNEG-OFFPATH (see (4.74)) must also be satisfied. As a result, although there is no local negation, some higher verb in the verb chain must be negated, so that transferred negation is available. If this is the case, the object may be assigned the accusative or genitive case – in this context GoN is optional, as in example (4.68) and its corresponding f-structure in (4.69).

To summarise: the constraint provided in (4.70) (see (4.78) for its expanded version), where a redefined version of STRCASE-OFFPATH template is used, imposes restrictions on the verb's object – it assigns appropriate structural case (using STRCASE-OFFPATH template defined in (4.71)) or allows clauses containing a ŻEBY-type complementiser. If there is no negation in the relevant domain, it assigns the accusative case. If the verb assigning case is negated locally, the genitive case is assigned. If there is no local negation but transferred negation is available, GoN is optional and the object may be assigned the accusative or genitive case.

Finally, (4.70), fully expanded in (4.78), is capable of handling sentences where the object is simple (it is not coordinated), it accounts for coordinate objects displaying strong parallelism (all conjuncts consistently bear structural case or all conjuncts are clauses with the appropriate complementiser) and it also takes into consideration the coordination of unlike categories – particular conjuncts may bear structural case or be clausal. Furthermore, it models correctly case assignment not only taking into account local syntactic context, but also verb chains, including the optionality of GoN in the latter environment (see the discussion in § 3.2.3.2).

4.5.2.2 Lexical case, other clause types

Having presented how off-path constraints may be used to handle structural case assignment to objects, let us see how this device may be applied to account for coordination of unlikes where some elements of the object require lexical case:

(4.79) Dorośli parlamentarzyści powinni się od was uczyć [kultury politycznej] i
adult MPs should REFL from you learn culture.GEN political.GEN and
[jak należy obradować].
how should sit
'Adult MPs should learn from you political culture and how to sit.' (NKJP)

(4.80) Boisz się [bezrobocia] i [że zabraknie Ci środków na utrzymanie]?
be.afraid.2.SG REFL unemployment.GEN and that lack you means for subsistence
'Are you afraid of unemployment and that you will lack the means of subsistence?'
(NKJP)

As opposed to structural case, lexical case assignment does not depend on the syntactic environment in any way (it is also known as inherent case), so the formalisation of relevant constraints is very simple in comparison to arguments which require structural case:

(4.81) $$(\uparrow \text{OBL}_{gen} \underset{(\leftarrow \text{CASE}) =_c \text{GEN} \;\vee\; (\leftarrow \text{CLAUSE-TYPE}) =_c \text{INT}}{\text{PRED}})$$

(4.82) $$(\uparrow \text{OBL}_{gen} \underset{(\leftarrow \text{CASE}) =_c \text{GEN} \;\vee\; (\leftarrow \text{COMP-FORM}) =_c \text{ŻE}}{\text{PRED}})$$

(4.81) allows objects realised as nominals marked for the genitive case or interrogative clauses ($(\leftarrow$ CLAUSE-TYPE$) =_c$ INT), or, as in (4.79), their coordination. The example in (4.80) is handled by the constraint in (4.82), which allows objects marked for the genitive case or clauses containing the ŻE-type complementiser ($(\leftarrow$ COMP-FORM$) =_c$ ŻE).

The f-structures corresponding to simplified versions of (4.79) and (4.80) are provided below: (4.83)[22] corresponds to the following fragment of (4.79): *parlamentarzyści powinni się od was uczyć [kultury] i [jak obradować]*; (4.84) represents the following part of (4.80): *Boisz się [bezrobocia] i [że zabraknie Ci środków].*

(4.83) $$\left[\begin{array}{ll}
\text{PRED} & \text{`SHOULD}\langle\boxed{2}\rangle\boxed{1}\text{'}\\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`MP'}\end{array}\right]\\
\text{XCOMP} & \boxed{2}\left[\begin{array}{ll}
\text{PRED} & \text{`LEARN}\langle\boxed{1},\boxed{3},\boxed{4}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\\
\text{OBL}_{gen} & \boxed{3}\left[\begin{array}{l}
\left\{\left[\begin{array}{ll}\text{PRED} & \text{`CULTURE'}\\ \text{CASE} & \text{GEN}\end{array}\right],
\left[\begin{array}{ll}\text{PRED} & \text{`SIT}\langle\boxed{1}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{`HOW'}\end{array}\right]\right\}\\ \text{CLAUSE-TYPE} & \text{INT}\end{array}\right]\right\}\\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]\\
\text{OBL} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`YOU'}\\ \text{CASE} & \text{GEN}\\ \text{PFORM} & \text{OD}\end{array}\right]
\end{array}\right]\\
\text{NEG} & +
\end{array}\right]$$

(4.84) $$\left[\begin{array}{ll}
\text{PRED} & \text{`FEAR}\langle\boxed{1},\boxed{2}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right]\\
\text{OBL}_{gen} & \boxed{2}\left[\begin{array}{l}
\left\{\left[\begin{array}{ll}\text{PRED} & \text{`UNEMPLOYMENT'}\\ \text{CASE} & \text{GEN}\end{array}\right],
\left[\begin{array}{ll}\text{PRED} & \text{`LACK}\langle\boxed{3},\boxed{4}\rangle\text{'}\\
\text{OBJ}_{\theta} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`YOU'}\\ \text{CASE} & \text{DAT}\end{array}\right]\\
\text{OBL}_{gen} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`MEANS'}\\ \text{CASE} & \text{GEN}\end{array}\right]\\
\text{COMP-FORM} & \text{ŻE}\end{array}\right]\right\}\\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]
\end{array}\right]$$

[22] Note that (4.83) involves control into selected conjuncts: the subject of SIT is controlled by the subject of LEARN (and SHOULD), while CULTURE is not controlled. This issue is discussed in § 4.7.

4.6 Alternative solution

A potential disadvantage of LFG is that some fundamental mechanisms such as the application of disjunctive statements have undesired effects due to the way they are interpreted under coordination, which makes it difficult to account for phenomena such as the coordination of unlike categories. As demonstrated at length in previous sections of this chapter (see § 4.3 especially), taking it into account requires adopting a special way of writing constraints, namely rewriting plain disjunctive constraints into off-path constraints. As a result, parallelism between imposing constraints on non-coordinate structures (and simple instances of coordination where conjuncts are nearly identical) and coordination of unlikes is lost – for constraints to be effective as desired in all these environments, more complex off-path constraints must be used at the moment.

An alternative to the "conservative" solution to the problem of unlike category coordination described above was offered in Przepiórkowski and Patejuk 2012, which discussed the discovered problems with the LFG formalism and offered a "liberal solution", which requires introducing some modifications to the formalism.

The proposal is to make entire statements distributive or non-distributive rather than individual features. If this change was introduced, statements using plain (non-off-path) constraints would have the same effect under coordination as their off-path counterparts. The gains are worth considering: coordination of unlikes could be handled in the same way as other phenomena in the grammar – there would be no need to adopt the off-path solution, which uses considerably more complicated notation.

Paraphrasing Sag 2002,[23] also a paper on coordination: most (if not all) of these problems can be dealt with in LFG by a simple change to the framework's foundational assumptions.

4.7 Control into selected conjuncts[24]

In standard LFG there is a distinction between closed and open grammatical functions (it was introduced in § 2.2.1): while closed ones are self-contained, open ones must be controlled by some other grammatical function – their subject must be structure-shared with their controller.

A problem arises with a certain type of coordination of unlikes, namely when categories which correspond to grammatical functions which prototypically differ with respect to the closed/open classification are coordinated – as in (4.85), an example from the literature:

(4.85) Chcę pić i papierosa.
want drink.INF and cigarette.ACC
'I want to drink and (I want) a cigarette.' (Kallas 1993, p. 123, ex. (102))

Without coordination, the nominal conjunct would typically be classified as OBJ,[25] a closed grammatical function – the f-structure corresponding to (4.87) is provided in (4.89). By contrast,

[23] Sag 2002, p. 268: "In this paper I explore the idea that most (if not all) of these problems can be dealt with in HPSG by a simple change to the framework's foundational assumptions."

[24] This section is based on Patejuk and Przepiórkowski 2014a.

[25] Since it is marked for structural case and it cannot passivise, it would correspond to OBL_{str} grammatical function in Polish LFG (see § 4.2.1 for discussion). However, in order to make it easier to compare constraints presented in this section with earlier constraints handling structural case assignment, the OBJ grammatical function will be assumed here instead of OBL_{str}.

the infinitival conjunct by itself would be a prototypical instance of XCOMP, an open grammatical function – this is shown in (4.88), the f-structure which corresponds to (4.86).

(4.86) Chcę pić.
want drink.INF
'I want to drink.'

(4.87) Chcę papierosa.
want cigarette.ACC
'I want a cigarette.'

(4.88)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`WANT}\langle\boxed{1},\boxed{2}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`I'}\end{array}\right]\\
\text{XCOMP} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`DRINK}\langle\boxed{1}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\end{array}\right]
\end{array}\right]$$

(4.89)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`WANT}\langle\boxed{1},\boxed{2}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`I'}\end{array}\right]\\
\text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`CIGARETTE'}\\ \text{CASE} & \text{ACC}\end{array}\right]
\end{array}\right]$$

The first issue with "unlike control" coordination such as that in (4.85) is the choice of the grammatical function: does the coordinated argument bear the grammatical function of the closed conjunct (OBJ), of the open conjunct (XCOMP), or some entirely different function? The second issue is how to establish the control relations appropriately – the nominal conjunct cannot be controlled (there is nothing to be controlled), while the infinitival conjunct (more specifically its subject) must be controlled. The following two subsections, § 4.7.1 and § 4.7.2, deal with these two issues in turn. Then, § 4.7.3 shows how to account for structural case assignment into selected conjuncts under unlike category coordination and selective control.

4.7.1 Which grammatical function?

It was suggested during the OBJ vs COMP debate (Dalrymple and Lødrup 2000, Alsina *et al.* 2005, Forst 2006, Berman 2007, Börjars and Vincent 2008) that it may be reasonable to get rid of the COMP grammatical function because it distorts the distinction between f-structure and c-structure: COMP is defined categorially, as the grammatical function assigned to closed clausal complements. Moreover, since clausal complements can be coordinated with uncontroversial nominal objects (see (4.28) repeated below as (4.90), with bracketing added following Dalrymple and Lødrup 2000), they should – at least in some cases – also bear the grammatical function OBJ, and their clausal categorial status should be ensured by other constraints.

(4.90) Pat remembered [the appointment] and [that it was important to be on time].
(Dalrymple and Lødrup 2000, ex. (5) = Sag *et al.* 1985, ex. (123a))

While Dalrymple and Lødrup 2000 maintain the COMP vs OBJ distinction, treating only some clausal arguments as OBJ, Alsina *et al.* 2005 get rid of COMP altogether. Further arguments against the mixed approach of Dalrymple and Lødrup 2000 are provided by Forst (2006), who also shows the grammar engineering advantages of getting rid of COMP; other work (Berman 2007, Börjars and Vincent 2008) also supports this move.[26]

Alsina *et al.* 2005, p. 41 go further, advancing the idea that XCOMP should also be removed from the repertoire of grammatical functions: "If we abandon the function COMP in LFG, the obvious question is, what about the function XCOMP? Given that they are both clausal complements, and that XCOMP may be considered a special case of COMP, XCOMP should probably go the same way as COMP."

[26] But see Lødrup 2012 for a voice of dissent.

However, it is not so obvious whether XCOMP should be removed on the grounds that it is the same case as COMP since there are two approaches to what can correspond to this grammatical function. In theoretical LFG, XCOMP usually represents both infinitival and predicative complements (as mentioned in § 2.2.1). Though both can be classified as clausal complements, as in the above quote from Alsina *et al.* 2005, they may correspond to various c-structure categories (infinitival phrases, predicative nominals and adjectives), unlike COMP, which corresponds to CP, a clause introduced by a complementiser or an interrogative item. Hence, XCOMP is defined by c-structure categories to a much lesser extent than COMP, and the postulate to get rid of it is less justified. By contrast, implemented grammars such as those developed within the ParGram project (Butt *et al.* 2002) tend to use distinct grammatical functions for infinitival complements (XCOMP) and predicative complements (XCOMP-PRED). Under such a definition of XCOMP, it is indeed a special case of COMP, differing only in the fact that it is the grammatical function corresponding to infinitival clauses.

Regardless of which of these two definitions of XCOMP is adopted, coordination such as in (4.85) causes a problem under the standard LFG approach to coordination, where conjuncts in a coordinate phrase correspond to the same grammatical function. Even if the lexical entry of a verb has a disjunctive specification where particular disjuncts correspond to various valence schemata, one with XCOMP, the other with OBJ, as in the lexical entry of *chcę* in (4.91), only one disjunct can be chosen. As a result, a multiclausal analysis where the first conjunct is a dependent of a verb taking an infinitival complement (see (4.92)), while the second conjunct is a dependent of the verb taking a nominal complement (as in (4.93)), is not possible.

(4.91) chcę V [PRED-XCOMP $\vee$ PRED-OBJ]
(↑ SUBJ NUM)= SG
(↑ SUBJ PERS)= 1

(4.92) PRED-XCOMP ≡ (↑ PRED)=‘WANT<(↑ SUBJ)(↑ XCOMP)>’

(4.93) PRED-OBJ ≡ (↑ PRED)=‘WANT<(↑ SUBJ)(↑ OBJ)>’

These considerations provide some justification for adopting the assumption that the common grammatical function assigned to the coordinate phrase in (4.85) is OBJ,[27] treating XCOMP as another elsewhere grammatical function (alongside COMP discussed in § 4.2.2). Moreover, following Arka and Simpson (1998), who convincingly argue for the possibility of control into SUBJ in Balinese, it is assumed that control into OBJ is also allowed in principle.

4.7.2 Establishing control

Now the problem is how to establish control appropriately – while the subject of the infinitival conjunct (*pić*) in (4.85) is controlled by the subject of *Chcę*, it is not the case with the nominal conjunct (*papierosa*), where such control would result in a violation of the coherence condition (see § 2.2.2 for discussion) – *papierosa* has no subject to be controlled.

In order to account for (4.85), the (pro-dropped) subject of the main verb must control the subject of the infinitival. Usually control is expressed via equations such as (4.94).

[27] This is the decision for the particular instance of coordination in (4.85). Note that it does not imply that OBJ should be used as the common grammatical function for all instances of unlike category coordination.

(4.94) $(\uparrow \textsc{subj}) = (\uparrow \textsc{obj subj})$

The grammatical function OBJ is used in (4.94) instead of XCOMP, because OBJ was chosen in § 4.7.1 as the common grammatical function of the coordinated argument in (4.85).

However, the control equation in (4.94) will not work in case of the coordinated object in (4.85). The difficulty stems from the fact that grammatical functions are distributive: "[G]overnable grammatical functions like OBJ are *distributive features*" (Dalrymple 2001, 365). This assumption is the basis of the analysis of coordination of heads which share arguments, as in (4.95), and for the explanation of the contrast – provided by Hall (1965, 66) – between (4.96) and (4.97). When the shared dependent has the same grammatical function in relation to coordinated heads (oblique, OBL, in (4.96)), it is grammatical. However, it is ungrammatical when the shared argument would fill different valence slots of relevant heads, as in (4.97), where *in the garage* would be an adjunct of WASH and an oblique for KEEP (for discussion, see Dalrymple 2001, 363–366).

(4.95) Chris selected and hired David.

(4.96) John washes and polishes his car in the garage.

(4.97) *John washes and keeps his car in the garage.

A control equation such as (4.94) will not yield the desired result under coordination of unlike categories such as in (4.85) because it would also distribute to the nominal conjunct, resulting in a violation of the coherence condition in this conjunct. This is illustrated in (4.98), where $\boxed{3}$ is the f-structure fragment corresponding to *papierosa* 'cigarette', where incoherence occurs – according to its PRED attribute, *papierosa* takes no arguments, yet a SUBJ attribute is introduced to its f-structure as the result of (4.94).

(4.98)
$$* \left[\begin{array}{ll} \textsc{pred} & \text{`}\textsc{want}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{`I'}\end{array}\right] \\ \textsc{obj} & \boxed{2}\left[\left\{\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{drink}\langle\boxed{1}\rangle\text{'}\\ \textsc{subj} & \boxed{1}\end{array}\right], \boxed{3}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{cigarette}\text{'}\\ \textsc{subj} & \boxed{1}\end{array}\right]\right\}\right]\end{array}\right]$$

4.7.2.1 First attempt (failed)

Unfortunately, using a disjunction such as in (4.99) to restrict the application of the control equation to the infinitival conjunct will not work either:[28]

(4.99) $[(\uparrow \textsc{obj cat}) =_c \textsc{inf} \;\wedge\; (\uparrow \textsc{subj}) = (\uparrow \textsc{obj subj})] \;\vee\; (\uparrow \textsc{obj cat}) \neq \textsc{inf}$

The disjunctive constraint in (4.99) will not have the desired effect under coordination since, as explained in § 4.3 (see the discussion of (4.19) repeated below as (4.100)), instead of being evaluated independently for each conjunct (as formalised in (4.100a)), it is evaluated once and distributed to all elements of the relevant structure (compare its formalisation in (4.100b)).

[28] This analysis uses the CAT attribute, which stores information about the part of speech of the head.

(4.100) a. $\forall x \in (\uparrow \textsc{gf})[A(x) \vee B(x)]$ (intended)

b. $\forall x \in (\uparrow \textsc{gf})A(x) \vee \forall x \in (\uparrow \textsc{gf})B(x)$ (actual)

In other words, the effect of the constraint provided in (4.99), interpreted as in (4.100b), is equivalent to having two separate lexical entries for the given verb: one which introduces the control equation if the complement is an infinitive (resulting in incoherence with the nominal conjunct of (4.85)) and another one which checks that the complement is not an infinitive (causing incompleteness in the infinitival conjunct of (4.85) as its SUBJ attribute is not filled). By contrast, the desired interpretation of (4.99) is the one provided in (4.100a) – it is satisfied under coordination if each element of the given structure satisfies any of the two constraints.

As explained in § 4.6, the cause of this issue is that, in LFG, what is distributive is features, not statements. The solution to this problem proposed there is to treat statements, rather than features, as distributive. If this modification was adopted, the statement in (4.99) would be able to handle control into selected conjuncts in sentences such as (4.85).

4.7.2.2 Second attempt (failed in XLE)

Another idea is to use off-path constraints (see § 4.4) to distribute the disjunction checking the category of the conjuncts and use the relevant control equation only with infinitival conjuncts – (4.101) below is an off-path version of (4.99):

(4.101)
$$(\uparrow \textsc{obj} \underset{\substack{[(\leftarrow \textsc{cat}) =_c \textsc{inf} \;\wedge\; (\leftarrow \textsc{subj}) = ((\textsc{obj} \leftarrow)\ \textsc{subj})] \\ \vee \\ (\leftarrow \textsc{cat}) \neq \textsc{inf}}}{\textsc{pred}})$$

The off-path constraint is attached to the PRED attribute of the OBJ grammatical function (which corresponds to the coordinate phrase in (4.85)) – the fact that PRED is a distributive feature ensures that the off-path constraint is distributed to all elements of the relevant f-structure. The first off-path disjunct of (4.101), $[(\leftarrow \textsc{cat}) =_c \textsc{inf} \wedge (\leftarrow \textsc{subj}) = ((\textsc{obj} \leftarrow)\ \textsc{subj})]$, checks whether the given element (conjunct) of the OBJ f-structure is an infinitive ($(\leftarrow \textsc{cat}) =_c \textsc{inf}$) and establishes control between the subject of this element (of OBJ f-structure) and the subject of the verb WANT: $(\leftarrow \textsc{subj}) = ((\textsc{obj} \leftarrow)\ \textsc{subj})$. A complex path is used for this purpose: $((\textsc{obj} \leftarrow)\ \textsc{subj})$ where $(\textsc{obj} \leftarrow)$ is an inside-out path pointing to the structure which contains OBJ and then to the SUBJ of this structure. This control equation is an off-path equivalent of the plain (non-off-path) control equation provided in (4.94). The second off-path disjunct in (4.101), $(\leftarrow \textsc{cat}) \neq \textsc{inf}$, ensures that the relevant element of OBJ f-structure is not an infinitive. For the f-structure to be well-formed, every element of OBJ f-structure must satisfy one of the disjuncts in the off-path constraint in (4.101).

Unfortunately, this solution will not work in implemented grammars since off-path constraints are non-constructive in the XLE system (`http://www2.parc.com/isl/groups/nltt/xle/`; Crouch *et al.* 2011) implementing LFG – they cannot introduce new attribute-value pairs to the f-structure, they can only act as constraining equations. By contrast, control equations must take the form of a defining equation to be effective: structure-sharing of the subject of WANT with the subject of its infinitival complement requires assignment rather than checking of identity – checking will not provide a value of the subject of the infinitive. This is shown in the plain control equation in (4.94): it uses the assignment operator '=', not the checking operator '$=_c$'. While both operators are available in XLE in plain constraints, the former is not available in off-path constraints.[29] For this reason, the constraint provided in (4.101) is not effective – it uses the assignment operator which is not available in this type of constraints in XLE.

However, it must be mentioned that this difficulty does not necessarily affect theoretical LFG analyses. There exist works which use off-path constraints in a constructive way, introducing new attribute-value pairs: these include Ash Asudeh's talk on *Reflexives in the Correspondence Architecture*.[30] More importantly, the currently prepared new edition of Dalrymple 2001 explicitly mentions that there are various types of off-path constraints: "Using the f-structure variables ← and →, any kind of constraint can be written as an off-path constraint; defining equations, constraining equations, existential constraints, and other kinds of f-descriptions may be specified." Similarly, the draft of the new edition of Bresnan 2000 also discusses the contrast between defining and constraining equality in the context of off-path constraints.[31]

Nevertheless, if a solution can be found which is implementable in XLE, it should perhaps be preferred to the one sketched here.

4.7.2.3 Third attempt (successful)[32]

A working solution is available which takes a slightly different approach to control – instead of a control equation such as in (4.94), subject control verbs are annotated with its modified version in (4.102), which uses a new attribute – CONTROLLER:

(4.102) $(\uparrow \textsc{subj}) = (\uparrow \textsc{obj controller})$

Placed in the lexical entry of the controlling verb, (4.102) introduces the attribute CONTROLLER – whose value is the f-structure of SUBJ of this verb – into the f-structure of OBJ of this verb. As a result, the f-structure of the subject of the main verb is available in the f-structure of the object of this verb. If needed, this information can be used for establishing control.

In order to account for (4.85), c-structure rules like those in (4.103)–(4.105) are needed, apart from rules (4.106)–(4.107), which handle non-coordinated objects:

(4.103) $$\begin{array}{lccc} \text{VP} & \rightarrow & \text{V} & \text{OBJ-ARG} \\ & & \uparrow=\downarrow & (\uparrow \textsc{obj}) = \downarrow \end{array}$$

(4.104) $\text{OBJ-ARG} \rightarrow \{ \text{OBJ-ARG}_{unlike} \mid \text{OBJ-ARG}_{infp} \mid \text{OBJ-ARG}_{np} \}$

[29] Thanks are due to John Maxwell for confirming this issue and discussion.

[30] Thanks are due to Dag Haug for drawing attention to this talk. The slides are available at http://users.ox.ac.uk/~cpgl0036/slides/asudeh-iceland09-reflexives.pdf (accessed on 6 October 2014); see the "Drip" on slide 51.

[31] "The off-path constraint in (66) is a defining equation, so even if no other part of the functional description for the f-structure had specified the feature [TENSE PAST], the off-path constraint would result in an f-structure that contains the feature. In contrast, if the equality in the off-path equation had been the constraining equality, $=_c$, the off-path constraint would only be satisfied if the f-structure independently contained the feature, as is the case with f-structure (65)."

[32] This is a modified version of the solution suggested by Mary Dalrymple after the presentation of this problem at the ParGram meeting in 2013 ("Revisiting grammatical functions?", with Adam Przepiórkowski).

(4.105)
$$\text{OBJ-ARG}_{unlike} \quad \equiv \quad \begin{array}{ccc} \text{INFP} & \text{Conj} & \text{NP} \\ \downarrow\in\uparrow & & \downarrow\in\uparrow \\ (\downarrow \text{CONTROLLER}) = (\downarrow \text{SUBJ}) & & \end{array}$$

(4.106)
$$\text{OBJ-ARG}_{infp} \quad \equiv \quad \begin{array}{c} \text{INFP} \\ \downarrow=\uparrow \\ (\downarrow \text{CONTROLLER}) = (\downarrow \text{SUBJ}) \end{array}$$

(4.107)
$$\text{OBJ-ARG}_{np} \quad \equiv \quad \begin{array}{c} \text{NP} \\ \downarrow=\uparrow \end{array}$$

Rule (4.103) assigns the grammatical function OBJ to the single argument in the VP, OBJ-ARG. Rule (4.104) rewrites OBJ-ARG to a disjunction of three rules whose left-hand side is not shown in the tree because ‘≡’ is used instead of ‘→’ as the rewrite operator in (4.105)–(4.107).[33] Rule (4.105) handles the coordination of unlike categories where an infinitival phrase (INFP) is coordinated with a nominal phrase (NP):[34] as in the standard analysis of coordination, conjuncts are added to a set. The non-standard element is the equation (↓ CONTROLLER) = (↓ SUBJ), which structure-shares the controller of INFP with the subject of INFP. Since this annotation is attached to INFP exclusively, the f-structure of the other conjunct, NP, is unaffected – it is not controlled, hence the coherence condition is satisfied.

The f-structure corresponding to (4.85) according to this analysis is provided in (4.108) below. It is produced by rules (4.103)–(4.105) in conjunction with the annotation (4.102) placed in the lexical entries of forms of the verb CHCIEĆ ‘want’.

(4.108)
$$\left[\begin{array}{ll} \text{PRED} & \text{‘WANT}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{‘I’} \end{array}\right] \\ \text{OBJ} & \boxed{2}\left[\left\{\left[\begin{array}{ll} \text{PRED} & \text{‘DRINK}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{CONTROLLER} & \boxed{1} \end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘CIGARETTE’} \\ \text{CONTROLLER} & \boxed{1} \end{array}\right]\right\}\right] \end{array}\right]$$

4.7.3 Interaction with structural case assignment

When discussing such coordination, one more issue needs to be addressed: the interaction with structural case assignment. Consider the examples below:

(4.109) Nie chciał pić ani kanapki.
NEG wanted drink.INF nor sandwich.GEN
‘He didn’t want to drink nor (did he want) a sandwich.’ (Kallas 1993, p. 92, ex. (49))

(4.110) Chciał pić i kanapkę.
wanted drink.INF and sandwich.ACC
‘He wanted to drink and (he wanted) a sandwich.’

[33] In LFG (and XLE) the convention is to have only one rule with a given left-hand side. This formal device is used in order to make OBJ-ARG more readable as well as to make it easier to refer to its particular disjuncts. The result is equivalent to having OBJ-ARG rewrite directly to respective right-hand sides of (4.105)–(4.107).

[34] To handle examples with different ordering of conjuncts or a different number of conjuncts than in (4.85), the rule in (4.105) must be modified accordingly.

As explained earlier, the structural case assigned to a nominal object depends on the syntactic environment. Simplifying a little (see § 3.2.3 or § 4.5.2 for discussion): when the verb is negated, the object takes the genitive case (as in (4.109), see the f-structure in (4.111)), when there is no negation, the object is in the accusative case (compare (4.110), whose f-structure representation is provided in (4.112)).

(4.111)
$$\left[\begin{array}{ll} \text{PRED} & \text{‘WANT}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\left[\text{PRED} \quad \text{‘PRO’}\right] \\ \text{OBJ} & \boxed{2}\left[\begin{array}{l} \left\{ \left[\begin{array}{ll} \text{PRED} & \text{‘DRINK}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{CONTROLLER} & \boxed{1}\end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘SANDWICH’} \\ \text{CASE} & \text{GEN} \\ \text{CONTROLLER} & \boxed{1}\end{array}\right]\right\} \\ \text{COORD-FORM} \quad \text{ANI}\end{array}\right] \\ \text{NEG} & + \end{array}\right]$$

(4.112)
$$\left[\begin{array}{ll} \text{PRED} & \text{‘WANT}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\left[\text{PRED} \quad \text{‘PRO’}\right] \\ \text{OBJ} & \boxed{2}\left[\begin{array}{l} \left\{ \left[\begin{array}{ll} \text{PRED} & \text{‘DRINK}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{CONTROLLER} & \boxed{1}\end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘SANDWICH’} \\ \text{CASE} & \text{ACC} \\ \text{CONTROLLER} & \boxed{1}\end{array}\right]\right\} \\ \text{COORD-FORM} \quad \text{I}\end{array}\right] \end{array}\right]$$

Case assignment under coordination could be handled using off-path constraints – a modified version of (4.66) provided in (4.113) (see (4.114) for its expanded version) would be used to handle simple cases (strictly local case assignment, see the definition of STRCASE-OFFPATH provided in (4.62)):

(4.113)
$$(\uparrow \text{OBJ} \quad \underset{\text{STRCASE-OFFPATH} \ \vee \ (\leftarrow \text{CAT}) =_c \text{INF}}{\text{PRED}} \quad)$$

(4.114)
$$(\uparrow \text{OBJ} \quad \underset{\begin{array}{c} [\neg((\text{OBJ} \leftarrow) \ \text{NEG}) \ \wedge \ (\leftarrow \text{CASE}) =_c \text{ACC}] \ \vee \\ [((\text{OBJ} \leftarrow) \ \text{NEG}) =_c + \ \wedge \ (\leftarrow \text{CASE}) =_c \text{GEN}] \ \vee \\ (\leftarrow \text{CAT}) =_c \text{INF} \end{array}}{\text{PRED}} \quad)$$

The only difference with respect to the constraint provided in (4.66) is the last off-path disjunct: instead of allowing for a clause featuring a ŻEBY-type complementiser ($(\leftarrow$ COMP-FORM$) =_c$ ŻEBY), the second disjunct in (4.113) allows for an infinitival conjunct ($(\leftarrow$ CAT$) =_c$ INF). The first disjunct, the call to the template STRCASE-OFFPATH, takes care of structural case assignment to nominal objects, as explained when discussing (4.62) used in (4.66).

To take case assignment in verb chains into account (see § 4.5.2.1 for discussion), a modified version of STRCASE-OFFPATH (see (4.71) for its definition) used in (4.70) must be used in (4.113). The following expanded constraint results:

(4.115)

$$
\begin{array}{c}
(\uparrow \textsc{obj} \qquad\qquad\qquad\qquad \textsc{pred} \qquad\qquad\qquad\qquad) \\
[\neg((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow\ \textsc{case}) =_c \textsc{acc}]\ \vee \\
[((\textsc{xcomp}^{*}\ \textsc{obj} \leftarrow)\ \textsc{neg}) =_c +\ \wedge \\
[[((\textsc{obj} \leftarrow)\ \textsc{neg}) =_c +\ \wedge\ (\leftarrow\ \textsc{case}) =_c \textsc{gen}]\ \vee \\
[\neg((\textsc{obj} \leftarrow)\ \textsc{neg})\ \wedge\ (\leftarrow\ \textsc{case}) \in_c \{\textsc{acc}, \textsc{gen}\}]]]\ \vee \\
(\leftarrow\ \textsc{cat}) =_c \textsc{inf}
\end{array}
$$

The constraint in (4.115) is almost the same as the one provided in (4.78) (see the accompanying discussion for detailed explanation of its fragments) – the only difference is the last off-path disjunct: as explained above, the last disjunct of (4.113) allows for an infinitival clause instead of a clause with ŻEBY-type complementiser, which is the case in (4.66).

4.8 Examples of lexical entries

This section gathers the lexical entries of the verbs discussed earlier in this chapter – note that the entries focus on constraints related to arguments which allow unlike category coordination, so only these arguments are discussed in detail. Though information about the requirements of other arguments is also provided, information about tense, aspect, mood and verbal agreement is omitted.

(4.116) is the entry used in (4.43), where the subject consists of a clause containing the ŻE-type complementiser and a nominative nominal phrase. The call to the template AGR-CASE-OFFPATH defined in (4.39) handles structural case assignment to the nominal conjunct of the subject, while the statement (← COMP-FORM) $=_c$ ŻE allows the clausal conjunct (the first conjunct in (4.43)). Structural case assignment to the object is handled using the template STRCASE defined in (3.67).

(4.116) dziwiło V

$$
\begin{array}{l}
(\uparrow \textsc{pred})=\text{‘}\textsc{puzzle}<(\uparrow \textsc{subj})(\uparrow \textsc{obj})>\text{’} \\
(\uparrow \textsc{subj} \qquad\qquad\qquad\qquad \textsc{pred} \qquad\qquad\qquad\qquad) \\
\qquad\qquad \textsc{agr-case-offpath}\ \vee\ (\leftarrow\ \textsc{comp-form}) =_c \textsc{że} \\
\textsc{strcase}
\end{array}
$$

The entries in (4.117) and (4.118) contain an object which may be realised as an instance of unlike category coordination: (4.117) is used in (4.52)–(4.54) and (4.68), where a nominal is coordinated with a clause containing the ŻEBY-type complementiser. While the entry in (4.118) also allows a nominal realisation of the object, the other possibility is an infinitival realisation, as in (4.85) (see also (4.109)–(4.110), which use a different form: *chciał*), which involves controlling into selected conjuncts (only infinitival) under coordination. Note that both entries use calls to the template STRCASE-OFFPATH defined in (4.71) in order to handle structural case assignment to the object (including long distance GoN). Finally, both entries take a subject – the relevant constraints are imposed by the call to the template AGR-CASE defined in (4.37).

(4.117) doradził V

$$
\begin{array}{l}
(\uparrow \textsc{pred})=\text{‘}\textsc{advise}<(\uparrow \textsc{subj})(\uparrow \textsc{obj})(\uparrow \textsc{obj}_{\theta})>\text{’} \\
\textsc{agr-case} \\
(\uparrow \textsc{obj} \qquad\qquad\qquad\qquad \textsc{pred} \qquad\qquad\qquad\qquad) \\
\qquad\qquad \textsc{strcase-offpath}\ \vee\ (\leftarrow\ \textsc{comp-form}) =_c \textsc{żeby} \\
(\uparrow \textsc{obj}_{\theta}\ \textsc{case}) =_c \textsc{dat}
\end{array}
$$

$$\begin{array}{lll}
\text{(4.118)} & \text{chcę} \quad \text{V} & (\uparrow \text{PRED})=\text{`WANT}<(\uparrow \text{SUBJ})(\uparrow \text{OBJ})>\text{'} \\
 & & \text{AGR-CASE} \\
 & & (\uparrow \text{OBJ} \underset{\text{STRCASE-OFFPATH} \;\vee\; (\leftarrow \text{CAT}) \;=_c\; \text{INF}}{\text{PRED}}) \\
 & & (\uparrow \text{SUBJ}) = (\uparrow \text{OBJ CONTROLLER})
\end{array}$$

The entries defined in (4.119) and (4.120)[35] take an argument which can be realised as a coordination of a nominal phrase marked for the lexical genitive case and an interrogative clause (see (4.79)) or a subordinate clause containing the ŻE-type complementiser (see (4.80)), respectively. As in (4.117)–(4.118) above, calls to the template AGR-CASE impose case constraints on the subject of relevant predicates.

$$\begin{array}{lll}
\text{(4.119)} & \text{uczyć} \quad \text{V} & (\uparrow \text{PRED})=\text{`LEARN}<(\uparrow \text{SUBJ})(\uparrow \text{OBL}_{gen})(\uparrow \text{OBL})>\text{'} \\
 & & \text{AGR-CASE} \\
 & & (\uparrow \text{OBL}_{gen} \underset{(\leftarrow \text{CASE}) \;=_c\; \text{GEN} \;\vee\; (\leftarrow \text{CLAUSE-TYPE}) \;=_c\; \text{INT}}{\text{PRED}}) \\
 & & (\uparrow \text{OBL PFORM}) =_c \text{OD} \;\wedge\; (\uparrow \text{OBL CASE}) =_c \text{GEN}
\end{array}$$

$$\begin{array}{lll}
\text{(4.120)} & \text{boisz} \quad \text{V} & (\uparrow \text{PRED})=\text{`FEAR}<(\uparrow \text{SUBJ})(\uparrow \text{OBL}_{gen})>\text{'} \\
 & & \text{AGR-CASE} \\
 & & (\uparrow \text{OBL}_{gen} \underset{(\leftarrow \text{CASE}) \;=_c\; \text{GEN} \;\vee\; (\leftarrow \text{COMP-FORM}) \;=_c\; \text{ŻE}}{\text{PRED}})
\end{array}$$

Finally, (4.121) is the entry of the verb used in (4.12), where the oblique object consists of a prepositional phrase coordinated with an instrumental nominal phrase. The remaining constraints, related to the subject and the object, are handled using calls to templates AGR-CASE and STRCASE, respectively.

$$\begin{array}{lll}
\text{(4.121)} & \text{owinął} \quad \text{V} & (\uparrow \text{PRED})=\text{`WRAP}<(\uparrow \text{SUBJ})(\uparrow \text{OBJ})(\uparrow \text{OBL})>\text{'} \\
 & & \text{AGR-CASE} \\
 & & \text{STRCASE} \\
 & & (\uparrow \text{OBL} \underset{(\leftarrow \text{CASE}) \;=_c\; \text{INST} \;\vee\; [(\leftarrow \text{PFORM}) \;=_c\; \text{W} \;\wedge\; (\leftarrow \text{CASE}) \;=_c\; \text{ACC}]}{\text{PRED}})
\end{array}$$

4.9 Summary

This chapter discussed unlike category coordination on the basis of examples from the literature as well as from NKJP. Numerous examples of coordination of unlikes are available in Walenty (discussed in § 8), which provides an explicit account of this phenomenon.

While the issue of unlike category coordination has been discussed in the LFG framework before (Peterson 2004, Dalrymple and Lødrup 2000, Alsina *et al.* 2005, for instance), there has been no formal analysis that would show how to impose constraints under such coordination.

[35] Constraints related to SIĘ marker, required by both predicates used in appropriate examples ((4.79) and (4.80), respectively), were omitted in their lexical entries as irrelevant to the discussion here. See Patejuk and Przepiórkowski 2015a for an LFG analysis taking into account various functions of SIĘ as well as multifunctionality.

This chapter fills this gap by presenting how unlike category coordination can be handled in LFG, providing an explicit formalisation of the analysis offered.

Moreover, the proposed analysis covers interactions with phenomena such as structural case assignment (for subject and object, the latter including long-distance GoN) and control, offering probably the first analysis of control into selected conjuncts in LFG.

Finally, it is perhaps worth noting that the analysis presented in this chapter was successfully implemented in XLE and it is a part of a large-scale LFG grammar of Polish – see the discussion in ch. 8 and ch. 9.

Chapter 5

Coordination of unlike grammatical functions

5.1 Introduction[1]

In LFG, as in other theories, it was assumed for a long time that coordinated items should belong to the same c-structure category. When coordination of unlikes came to the attention of LFG, the new assumption was that it is possible to coordinate different categories but the coordinate structure is assigned the grammatical function as a whole, so that all conjuncts correspond to the same grammatical function. Dalrymple and Lødrup 2000 discusses an example from Sag *et al.* 1985 where a nominal is coordinated with a clause and together they correspond to the object grammatical function; see (4.28) provided in ch. 4.

Peterson 2004, p. 643, advances a strong claim along these lines: "Coordination is subject to the condition that items can be conjoined if and only if they satisfy the condition of functional equivalence." and he does so with extreme certitude: "This condition does not have to be stipulated; it follows as an axiom from the general principles of functional application to sets."

However, over the years it was noticed in different formalisms that, under certain circumstances, it is possible to coordinate dependents which bear different grammatical functions. This phenomenon was first discussed in Sannikov 1979, 1980 on the basis of Russian data, its existence was mentioned (though largely disregarded) in Mel'čuk 1988 and later a dependency-like analysis was sketched for Polish by Kallas (1993); other analyses include Chaves and Paperno 2007 for Russian (where this phenomenon is referred to as "hybrid coordination") and Bîlbîie and Gazdik 2012 for Hungarian and Romanian in HPSG, Paperno 2012 for Russian in Categorial Grammar and Gazdik 2010, 2011 for French and Hungarian in LFG. There is also a range of papers in Chomskyan, non-constraint-based frameworks: these include (among others) Lipták 2001, 2012 and Citko and Gračanin-Yüksek 2013.

This chapter presents attested examples selected from abundant data extracted mainly from NKJP, demonstrating the diversity of lexico-semantic coordination in Polish, and shows how generalisations stemming from presented data can be formalised in an LFG grammar of Polish. It demonstrates, providing relevant corpus evidence, that it is not only possible to coordinate

[1] This chapter is based on the following papers: Patejuk and Przepiórkowski 2012b,a, Przepiórkowski and Patejuk 2014, Patejuk and Przepiórkowski 2014c, Patejuk 2015.

different grammatical functions in Polish, but also to coordinate dependents which correspond to grammatical functions belonging to various levels in the f-structure, whereby particular conjuncts depend on different heads. Furthermore, it provides arguments in support of using different formal representations of lexico-semantic coordination, monoclausal vs multiclausal, depending on which items are involved in such coordination. It also takes into account complex issues such as the representation of embedding of coordination of different grammatical functions, as well as interactions with verbal coordination. Finally, it provides some discussion of less common types of pronouns involved in such coordination, including the range of possible modifiers.

5.2 Basic data and generalisations

Though probably most examples of lexico-semantic coordination discussed in the literature include interrogative items, as in (5.1), where the first conjunct (*Kogo*) is the object, while the other (*komu*) is an indirect object marked for the dative case, this phenomenon is by no means limited to such items.

(5.1) Kogo i komu przedstawił?
who.ACC and who.DAT introduced
'Who did he introduce to whom?' (Kallas 1993, p. 121, ex. (241))

(5.2) czy komukolwiek, kiedykolwiek i do czegokolwiek przydał się poradnik
PART anybody.DAT anytime and for anything come in handy guide
'Has a(ny) guide ever come in handy to anybody for anything?' (NKJP)

(5.3) Obiecać można wszystko i wszystkim.
promise may everything.ACC and everyone.DAT
'One may promise everything to everyone.' (NKJP)

In (5.2) all coordinated items contain the following forms of free choice pronouns: *komukolwiek*, *kiedykolwiek*, *czegokolwiek*. This is the only similarity: particular conjuncts belong to different categories (noun phrase, adverbial phrase and a prepositional phrase, respectively) and bear distinct grammatical functions (see § 4.2.1): indirect object (OBJ_θ), adjunct (ADJ) and oblique object (OBL), respectively. In (5.3) both conjuncts are pronouns expressing a universal quantifier: the first corresponds to the direct object (OBJ), while the other is the indirect object (OBJ_θ). Unlike in (5.2), both conjuncts belong to the same category (noun phrase).

It is also possible to coordinate phrases containing pronouns which belong to yet another semantic class, namely *n*-words:

(5.4) nikogo i nic nie może tłumaczyć.
nobody.GEN and nothing.NOM NEG can excuse
'Nothing can excuse anybody.' (NKJP)

This example is interesting because particular conjuncts not only correspond to distinct grammatical functions but they are also dependents of different predicates: the second conjunct (*nic*) is the subject (SUBJ) of the main clause verb *może*,[2] while the other (*nikogo*) is the object (OBJ)

[2] As a result of structure-sharing under raising, it is also the subject of *tłumaczyć* at the same time.

of *tłumaczyć* in the embedded infinitival clause (XCOMP). There are further, more sophisticated examples of coordination where conjuncts depend on different heads:

(5.5) Skąd i jakie otrzymujemy informacje?
whence and what.ACC receive information.ACC
'What information and from where do we receive?' (NKJP)

In (5.5) conjuncts are modifiers of different predicates: the first conjunct, *Skąd*, is an adjunct[3] (ADJ) of the verb (*otrzymujemy*), while the other, *jakie*, modifies the verb's object (*informacje*). Furthermore, it is possible that one lexico-semantic conjunct may be the head of the other one:

(5.6) Ile i czego znaleźli?
how much.ACC and what.GEN found
'How much, and (of) what, did they find?' (NKJP)

As mentioned in § 3.1.2.1, Polish numeral phrases are headed by the numeral while the accompanying nominal is analysed as its dependent. In (5.6) the first conjunct (*ile*) is a numeral, analysed as the object (OBJ) of the verb (*znaleźli*), while the other (*czego*) is the object of this numeral – together they constitute a complete numeral phrase (*ile czego*) with the following f-structure:

(5.7)
$$\begin{bmatrix} \textsc{pred} & \text{`}\textsc{how\_much}\langle\boxed{1}\rangle\text{'} \\ \textsc{obj} & \boxed{1}\begin{bmatrix} \textsc{pred} & \text{`}\textsc{what}\text{'} \end{bmatrix} \end{bmatrix}$$

Conjuncts taking part in lexico-semantic coordination in (5.1), (5.5) and (5.6) belong to the semantic class mentioned in the beginning, namely interrogative items. Let us consider one more example featuring such conjuncts:

(5.8) Nie wiadomo było, czy *(i) kiedy wróci.
NEG know was PART and when returns
'It was not clear whether and when he would return.' (NKJP)

At first glance (5.8) appears similar to previous examples as all conjuncts represent the same broad semantic class, interrogative words in this case: the first conjunct is a *yes/no* question particle (*czy*), the other is an adverb (*kiedy*). The particle is analysed as a marker (marking *yes/no* interrogative clauses), the other conjunct is treated as an adjunct of the verb (but see fn. 3). There is a crucial difference, though: when the conjunction (*i*) is removed, (5.8) becomes ungrammatical,[4] while all other examples presented so far remain grammatical without the conjunction. It is possible, however, to use a multiclausal construction as an alternative to (5.8), with roughly the same meaning:

(5.9) Nie wiadomo było, [czy wróci] i [kiedy wróci].
NEG know was PART returns and when returns
'It was not clear whether he would return and when he would return.'

[3] Note that it could also be analysed as an oblique (ablative).

[4] Under the reading where *kiedy* is interrogative rather than indefinite – see the discussion in fn. 21.

§ 5.4.2.16 is devoted to the discussion of examples such as (5.8) and how they differ from the remaining ones.

Before proceeding, let us briefly summarise the properties of lexico-semantic coordination presented so far: particular conjuncts bear distinct grammatical functions (arguments, adjuncts) or no grammatical function at all (as in the case of *czy*, the *yes/no* question marker), they may also belong to different levels of f-structure as long as each conjunct represents the same semantic type (pronouns expressing a universal quantifier, free choice pronouns, *n*-words or *wh*-words). Finally, particular conjuncts may correspond to different categories at the level of c-structure.

5.3 Is this coordination?

Since lexico-semantic coordination is a potentially very surprising variety of coordination, it is natural to question whether it is indeed an instance of coordination. While typical tests such as agreement seem inapplicable, there is fortunately some other potentially convincing evidence.

First, it is possible to use such constructions with items which are unambiguous and uncontroversial conjunctions in Polish:

(5.10) kto oraz kiedy miałby płacić za postawiony budynek
who.NOM and when should pay for erected building
'Who and when would be supposed to pay for the erected building?' (NKJP)

(5.11) Rozmowa z dr. Stanem Nagorskim, belgijskim lekarzem dokonującym eutanazji,
talk with doctor Stan Nagorski, Belgian physician performing euthanasia
o tym, dlaczego oraz komu to robi i kiedy odmawia
about this why and who.DAT this does and when refuses
'An interview with dr Stan Nagorski, a Belgian physician performing euthanasia, about why and to whom he does this and when he refuses.' (Polityka 5, 29.01–4.02.2014)

(5.10)–(5.11) feature *oraz* ('and'), an entirely unambiguous conjunction – there is no other interpretation of this word in Polish.

There exist examples with *ani*, a Polish conjunction which is an *n*-word – it can occur when sentential negation is available. As shown in (5.12)–(5.13), these examples are ungrammatical without negation, because *ani* cannot be licensed, which strongly suggests that it is *ani*, the *n*-word conjunction, which occurs in these examples.[5]

(5.12) Nigdy nie wyjeżdżałyśmy na wakacje, bo *(nie) miałyśmy z kim ani
never NEG leave for holidays because NEG had with who.INST nor
za co...
for what.ACC
'We would never go on holiday because there was nobody we could go with and there was no money to go.' (Joanna Bator, *Ciemno, prawie noc*, p. 119)

[5] Note that such sentences would be grammatical without negation if a different conjunction was used, such as *i*, for instance, which does not require negative concord.

(5.13) Rząd USA *(nie) ujawnia, kogo ani dlaczego umieścił na liście osób,
government USA NEG discloses who.ACC and why put on list people
których nie należy wpuszczać do samolotów.
who NEG should let in to planes
'The US government does not disclose who and why they put on the list of people who should not be allowed on board of planes.' (Gazeta Wyborcza, 19–21.04.2014, p. 9)

Furthermore, it is possible to find examples where a preconjunction is used, as in 'both… and… ' coordinate structures:[6]

(5.14) A jest i co, i gdzie eksportować.
and is and what.ACC and where export
'There (certainly) is what and where to export to.' (NKJP)

While all examples presented so far featured conjoining *and*-type conjunctions (mostly *i*), there exist examples with alternative conjunctions:

(5.15) Mile widziane odpowiedzi merytoryczne, bez przypuszczeń kto lub czego
welcome responses substantive without speculating who.NOM or what.GEN
będzie w Wikipedii szukał.
AUX in Wikipedia seek
'Welcome are substantive responses, without speculating who will seek what in Wikipedia.' (NKJP)

While the word *lub* 'or' is not perfectly unambiguous, its other interpretation, the imperative form of the verb *lubić* 'like', is not an option in this context, leaving the conjunction interpretation.

5.4 Monoclausal or multiclausal: a critical review of tests for determining representation

This section discusses the issue of how conjuncts in constructions known as lexico-semantic or hybrid coordination should be represented in Polish. It concentrates on examples such as (5.16), while examples where conjuncts are non-adjacent (see (5.17) where the last conjunct is placed after the verb) remain outside of its scope.

(5.16) Kto i komu zaufał?
who.NOM and who.DAT trusted
'Who trusted whom?'

(5.17) Kto zaufał i komu?
who.NOM trusted and who.DAT
'Who trusted and whom?'

Monoclausal analyses argue that conjuncts belong to the same clause, which means that (5.16) is treated similarly to multiple questions (as in (5.18)), see the corresponding f-structure in (5.19). By contrast, multiclausal analyses (often referred to in the literature as biclausal, though more than two clauses may be involved) treat (5.16) as consisting of two different clauses, as in (5.20) which consists of two distinct questions; compare the f-structure in (5.21).

[6] Thanks are due to Tracy Holloway King for suggesting the use of such constructions.

(5.18) Kto komu zaufał?

(5.19)
$$\left[\begin{array}{ll} \text{PRED} & \text{‘TRUST}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ}\ \boxed{1} & \left[\begin{array}{ll} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{NOM} \end{array}\right] \\ \text{OBJ}_{\theta}\ \boxed{2} & \left[\begin{array}{ll} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{DAT} \end{array}\right] \end{array}\right]$$

(5.20) Kto ~~zaufał~~ i komu zaufał?

(5.21)
$$\left\{\begin{array}{l} \left[\begin{array}{ll} \text{PRED} & \text{‘TRUST}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ}\ \boxed{1} & \left[\begin{array}{ll} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{NOM} \end{array}\right] \\ \text{OBJ}_{\theta}\ \boxed{2} & \left[\begin{array}{ll} \text{PRED} & \text{‘PRO’} \\ \text{CASE} & \text{DAT} \end{array}\right] \end{array}\right], \\ \left[\begin{array}{ll} \text{PRED} & \text{‘TRUST}\langle\boxed{3},\boxed{4}\rangle\text{’} \\ \text{SUBJ}\ \boxed{3} & \left[\begin{array}{ll} \text{PRED} & \text{‘PRO’} \\ \text{CASE} & \text{NOM} \end{array}\right] \\ \text{OBJ}_{\theta}\ \boxed{4} & \left[\begin{array}{ll} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{DAT} \end{array}\right] \end{array}\right] \end{array}\right\}$$

§ 5.4.1 provides a brief discussion of derivational multiclausal analyses of lexico-semantic coordination offered for Polish. It is followed in § 5.4.2 by a critical review of a range of tests proposed for languages employing lexico-semantic coordination (these include, apart from Polish, also Bulgarian, Hungarian, Romanian, Russian and West Armenian) aiming to determine which representation should be adopted for lexico-semantic coordination in Polish: monoclausal or multiclausal.

An initial LFG analysis of such coordination, capable of producing both monoclausal and multiclausal representation, is presented in § 5.5. Problems with this analysis are presented in § 5.6. An improved analysis, accounting for problematic issues, is provided in § 5.7.

Finally, § 5.8 discusses less frequent types of conjuncts taking part in lexico-semantic coordination, together with observed modification patterns.

5.4.1 Multiclausal analyses

This section discusses two recent multiclausal analyses of lexico-semantic coordination, namely those of Tomaszewicz 2011a and Citko and Gračanin-Yüksek 2013. These analyses were chosen because they were originally applied to Polish data and they were also accompanied by an explanation of the resulting representation.

Although the analyses of Tomaszewicz 2011a and Citko and Gračanin-Yüksek 2013 differ with respect to formal devices they use – Tomaszewicz 2011a operates with the notion of deletion (under identity) while Citko and Gračanin-Yüksek 2013 use multidominance (sharing certain branches by clauses) – the intuition behind these analyses is strikingly similar: both offer two-fold analyses (in order to account for two different readings: single pair as opposed to pair list) which cover nearly equivalent cases.

There are two ways of answering multiple questions (such as ‘Who left when?’), which ask about more than one variable (here: subject, time) and expect answers consisting of tuples (here: 2-tuples), usually referred to as a “pair” (probably because 2-tuples are common). There are “pair list” answers, which consist of a list (of more than one element) of tuples (‘Mary left yesterday, John left two days ago.’). On the other hand, there are “single pair” answers, which

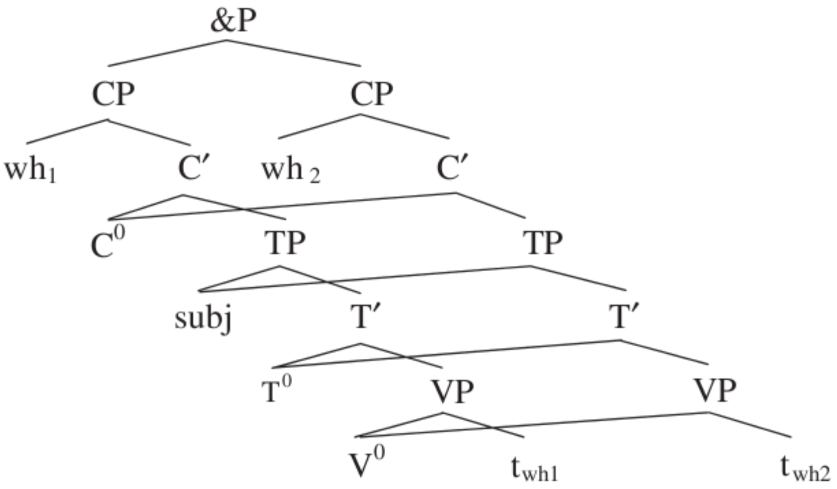


Figure 5.1: Non-bulk sharing structure (Citko and Gračanin-Yüksek 2013, ex. (6c))

consist of a single tuple (‘Mary left yesterday.’). Similarly, single questions, which ask about only one variable (‘Who left on Monday?’), require single answers (‘Mary left on Monday.’).

The first analysis in Tomaszewicz 2011a (henceforth T_1) is the one where two conjuncts belong to two distinct clauses and missing arguments (if any) are filled using implicit pronouns – the representation provided in (5.23)[7] corresponds to (5.22), more precisely, to its single pair reading, while the interpretation accompanying this analysis is provided in (5.24).

(5.22) Kto i co kupił?
who and what bought

(Tomaszewicz 2011a, ex. (1))

(5.23) [who [$_{TP}$ ~~who~~ bought something]] & [what [$_{TP}$ *pro* bought ~~what~~]

(Tomaszewicz 2011a, ex. (4a))

(5.24) Who bought something? And what did they buy? (Tomaszewicz 2011a, ex. (4b))

Since both questions in (5.23) are single questions (this is reflected in the interpretation in (5.24)), Tomaszewicz 2011a uses this strategy for obtaining the single pair reading.

The counterpart of this strategy is what Citko and Gračanin-Yüksek 2013 call the non-bulk sharing strategy (henceforth CGY_1), where “*wh*-words are NEVER shared between the two CPs (while everything else in the structure is)” – their representation is shown in Figure 5.1.

The second analysis proposed by Tomaszewicz 2011a (henceforth T_2) involves a coordination of two questions: a single question in the first clause (containing the *wh*-word corresponding to the first conjunct of lexico-semantic coordination) and a multiple question in the second clause (containing both *wh*-words). As explained in Tomaszewicz 2011a, “the two identical wh-phrases in the two conjuncts undergo ATB movement”,[8] while the second *wh*-phrase stays in the second clause. (5.25) provides the representation (including implicit pronouns) of the pair list reading of (5.22); its interpretation is provided in (5.26). Note that both include two questions:

[7] Brackets in (5.23) are mismatched (the closing bracket is missing) following Tomaszewicz 2011a.

[8] Trask 1993, p. 5 defines ATB as: “Denoting certain extraction phenomena in which the extracted constituent is simultaneously related to a gap in every conjunct of coordination: *That book you were reading* e *and Lisa wanted to buy* e *is out in paperpack.*”

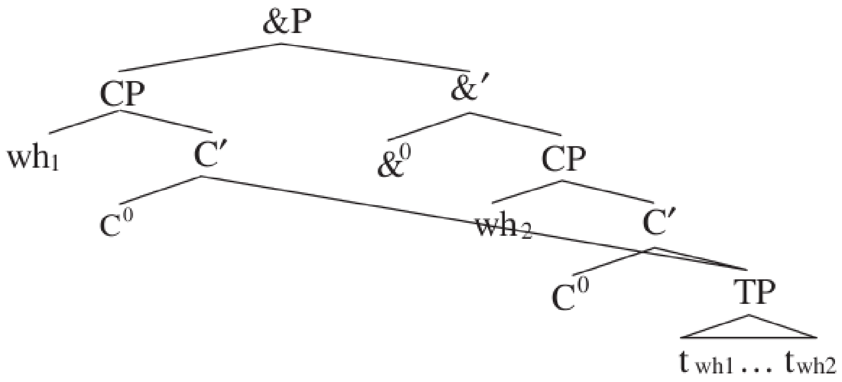


Figure 5.2: Bulk sharing structure (Citko and Gračanin-Yüksek 2013, ex. (6b))

(5.25) [who [~~who~~ [$_{TP}$ ~~who~~ bought something]] & [~~who~~ what [$_{TP}$ ~~who~~ bought ~~what~~]]] (Tomaszewicz 2011a, ex. (6a))

(5.26) Who bought something? And who bought what? (Tomaszewicz 2011a, ex. (6c))

Citko and Gračanin-Yüksek 2013 use the bulk-sharing strategy (henceforth CGY$_2$) which is a near counterpart of the pair list multiclausal analysis of Tomaszewicz 2011a. Under this analysis "there IS a point in the derivation when the two *wh*-phrases belong to both CPs, even though in the final representation each *wh*-phrase occupies a specifier of a different CP", see Figure 5.2 for an illustration. The difference with respect to the analysis of Tomaszewicz 2011a is that only one of the clauses, the second one, contains a multiple question on the analysis of Tomaszewicz 2011a, while in the analysis of Citko and Gračanin-Yüksek 2013 this is the case with both clauses. As a consequence, the multidominance bulk sharing analysis does not assume implicit arguments as there are no missing arguments in the representation.

5.4.2 Critical review of selected arguments

This section provides a critical review of the main arguments found in the literature supporting the monoclausal or the multiclausal analysis of lexico-semantic coordination when applied to data from Polish.

5.4.2.1 Sentence-level adverbs

Tomaszewicz 2011a claims that lexico-semantic coordination is multiclausal due to the fact that it is possible to split the conjuncts using a sentence-level adverbial. After providing examples from Bulgarian, she notes: "In Polish the facts are exactly the same as in Bulgarian and the speaker-oriented adverbs include: *najważniejsze* 'most importantly', *zwłaszcza* 'importantly', *niestety* 'unfortunately', *na szczęście* 'fortunately', *o dziwo* 'surprisingly'."

For each of the "sentence-level adverbials" listed above, a counterexample was found in NKJP – examples provided below feature such adverbs inside plain NP coordination:

(5.27) Zdemontowane zostały [piece, maszyny i co najważniejsze pompy].
removed were furnaces machines and what most important pumps
'Furnaces, machines and, what is most important, pumps were removed.' (NKJP)

(5.28) W domu po prostu zamęczał [matkę i zwłaszcza ojca]
at home simply pestered mother and especially father
‘At home he would simply pester his mother and especially his father.’ (NKJP)

(5.29) Z tego tytułu zachowanie [posłów SLD i niestety posłów PSL] jest wyjątkowo
for this reason behaviour MPs SLD and unfortunately MPs PSL is particularly
złe, naganne
bad reprehensible
‘For this reason the behaviour of SLD MPs and, unfortunately, PSL MPs is particularly bad, reprehensible.’ (NKJP)

(5.30) Dali mi [trochę forsy i na szczęście samochód]
gave me some money and fortunately car
‘They gave me some money and, fortunately, a car.’ (NKJP)

(5.31) [Włochy, Grecja, Francja, Niemcy i o dziwo Węgry wraz ze Słowacją] są
Italy Greece France Germany and surprisingly Hungary with Slovakia are
nastawione prorosyjsko.
disposed pro-Russian
‘Italy, Greece, France, Germany and, surprisingly enough, Hungary with Slovakia have pro-Russian attitude.’ (NKJP)

Examples provided above contain precisely the “sentence-level adverbials” listed by Tomaszewicz 2011a – they can clearly occur in plain coordination, where particular conjuncts correspond to the same grammatical function.

It is dubious whether these examples should be analysed as multiclausal simply because a “sentence-level adverbial” is present – examples where such adverbials are placed between conjuncts of the coordinate subject seem to provide strong counterevidence to such claims as the verb displays plural agreement, which would be unexpected under the multiclausal analysis. Examples where the subject is split using such adverbials include (5.27) and (5.31), though an example with singular conjuncts could make a stronger argument – see constructed (5.32), a modified version of (5.31):

(5.32) [Francja i o dziwo Słowacja] są nastawione prorosyjsko.
France.SG and surprisingly Slovakia.SG are disposed.PL pro-Russian

It seems more likely that “sentence-level adverbials” may split conjuncts, making coordination discontinuous.

5.4.2.2 Clausal coordinators

Tomaszewicz 2011a claims that *a* is a strictly clausal coordinator in Polish (“*a* never conjoins constituents smaller than a full clause”) and provides the following examples as evidence:

(5.33) Kto a najważniejsze co mówił o tobie?
who.NOM and most importantly what.ACC said about you
‘Who said something about you and what did they say?’
(Tomaszewicz 2011a, ex. (11))

(5.34) Jan i/*a Maria
Jan and Maria
(Tomaszewicz 2011a, ex. (12a))

(5.35) wąski i/*a długi mostek
narrow and long bridge
(Tomaszewicz 2011a, ex. (12b))

First, it is worth noting that, as shown in (5.36), (5.33) is ungrammatical without *najważniejsze* – Tomaszewicz 2011a mentions in footnote 3 that “the adverb is needed here since “a” is contrastive, and the adverb provides the needed contrast”.

(5.36) Kto a *(najważniejsze) co mówił o tobie?
who.NOM and most importantly what.ACC said about you

If so, the judgement in (5.34) is controversial – an isolated fragment does not prove that NPs cannot be coordinated using *a* – it seems that once appropriate adverbials are added, as in constructed (5.37) and authentic (5.38), relevant examples are grammatical:[9]

(5.37) Jan i/a zwłaszcza Maria głośno chrapią.
Jan.SG and especially Maria.SG loudly snore.PL
‘Jan and especially Maria snore loudly.’

(5.38) Życie, a zwłaszcza śmierć Angeliki de Sancé
life and especially death Angélique de Sancé
‘Life and especially death of Angélique de Sancé’ (Google)

Secondly, the judgement provided in (5.35) seems to be wrong when confronted with corpus data – there are numerous examples in NKJP where *a* is used when coordinating adjectives in a constrastive manner, so again it does not follow that *a* is an exclusively clausal coordinator:

(5.39) Latem umarł ksiądz Józef Tischner – piękny człowiek i piękny mężczyzna:
summer died reverend Józef Tischner beautiful human and beautiful man
[[wrażliwy i delikatny] a mocny], [uśmiechnięty a poważny], [[bardzo mądry]
sensitive and delicate but strong smiling but serious very wise
a prosty].
but simple
‘This summer reverend Józef Tischner passed away – a beautiful human and a beautiful man: sensitive and delicate yet strong, smiling yet serious, very wise yet simple.’
(NKJP)

(5.40) padł ofiarą sprytnego podstępu i przy pomocy [zręcznych a fałszywych] argumentów
fell victim cunning trick and with help clever but false arguments
został nakłoniony do udziału
was persuaded to participation
‘He fell victim to a cunning trick and using clever yet false arguments he was persuaded to participate.’ (NKJP)

[9] Note that plural agreement in (5.37) shows that it is monoclausal, see the discussion in § 5.4.2.1.

The examples provided above show that the conjunction *a* may be used in Polish in plain AP coordination – there is no reason to claim that such examples are multiclausal. As a result, such examples provide evidence against the claim that *a* is a strictly clausal coordinator in Polish and that structures which contain it such as the lexico-semantic coordination example in (5.33) must be multiclausal.

5.4.2.3 Degree questions

Another argument in support of the multiclausal analysis is based on "degree questions" and "null measure phrases": Tomaszewicz 2011b claims that the coordinated question is not felicitous "if the interpretation of the null argument in the first conjunct as an indefinite is not felicitous". Tomaszewicz 2011b builds an argument in favour of the multiclausal representation of lexico-semantic coordination on the basis of observations concerning "measure phrase arguments", which are "infelicitous with predicates such as *mieć n wzrostu* ('to have *n* in height', meaning 'to be *n* tall', where *n* stands for a numeral and a unit of measurement) or *mieć n lat* ('to have *n* years' meaning 'to be *n* years old')". To support this line of argument, Tomaszewicz 2011b uses examples such as (5.41) and (5.42), which she claims to be infelicitous.

(5.41) #Kto i ile ma wzrostu?
who.NOM and how much.ACC has height.GEN
'#Who measures something in height and how much do they measure in height?'
'#Who measures something in height and who measures how much in height?'
(Tomaszewicz 2011b, ex. (15))

(5.42) #Kto i ile ma lat?
who.NOM and how many.ACC has years.GEN
'#Who is some years old and how old are they?'
'#Who is some years old and who is how old?' (Tomaszewicz 2011b, ex. (16))

Judgements provided by Tomaszewicz 2011b are difficult to defend in the light of search results returned by `google.com` for the following simple query `"kto i ile ma lat"`:[10]

(5.43) Sonda: kto i ile ma lat?
poll who.NOM and how many.ACC has years.GEN
'Poll: who is how old?' (Google)

(5.44) Jakie durniu ma znaczenie kto i ile ma lat?
what fool has importance who.NOM and how many.ACC has years.GEN
'What does it matter, you fool, who is how old?' (Google)

(5.45) Dlaczego w Polsce przywiązuje się tak wielkie znaczenie temu kto i
why in Poland attaches REFL so great importance this who.NOM and
ile ma lat?
how many.ACC has years.GEN
'Why so much importance is attached in Poland to the issue of who is how old?'
(Google)

[10] Note that quotes are part of the query – this way the search engine looks for the entire phrase.

These counterexamples undermine the argument of Tomaszewicz 2011b in favour of the multiclausal analysis. This, however, does not provide constructive evidence in favour of the monoclausal analysis.

5.4.2.4 Focus pronominal

Tomaszewicz 2011a claims that it is possible to use TO, an element which she refers to as a "focus pronominal", inside lexico-semantic coordination with *wh*-words in Bulgarian and Polish. She provides an example from Bulgarian (good according to her), see (5.46),[11] and claims that "The facts are the same in Polish". However, this claim is unjustified – examples such as (5.47), with *to*, are not acceptable to consulted native speakers of Polish and there are no examples in NKJP which would match the query `[base="kto|co"] i to [base="kto|co"]`:[12]

(5.46) Koj i to kakvo kupi?
who and FOCUS what bought
'Who bought something and what was it that they bought?'
(Tomaszewicz 2011a, ex. (22))

(5.47) Kto i (*to) co kupił?
who and FOCUS what bought

5.4.2.5 Rhetorical questions

Paperno 2012 investigates the availability of a rhetorical reading in lexico-semantic coordination of interrogative words, multiple *wh*-questions and single questions, as in the examples below:

(5.48) Kto i kogda platit nalogi?
who and when pays taxes
rhetorical reading available: 'Nobody ever pays taxes.' (Paperno 2012, ex. (6a))

(5.49) Kto kogda platit nalogi?
who when pays taxes
rhetorical reading not available, only 'Who pays taxes when?'
(Paperno 2012, ex. (6b))

(5.50) Kto platit nalogi?
who pays taxes
rhetorical reading available: 'Nobody pays taxes.' (Paperno 2012, ex. (6c))

On the basis of examples and judgements related to their interpretation provided above, Paperno 2012 concludes that "coordinated wh-words behave like single wh-words" with respect to the possibility of being used as rhetorical questions. On the other hand, unlike lexico-semantic coordination and single questions, multiple *wh*-questions "usually can not function as rhetorical questions".

[11] In (5.46) the focus pronominal TO is glossed as FOCUS, unlike in Tomaszewicz 2011a.

[12] See `http://nkjp.pl/poliqarp/help/ense3.html` for the description of the query language of Poliqarp, one of tools for searching NKJP.

Although Paperno 2012 does not pursue this idea, this contrast could suggest that lexico-semantic coordination is multiclausal – since such questions pattern with single questions rather than multiple questions, it could be the case that there are two underlying single questions which were coordinated, as in the T_1 analysis of Tomaszewicz 2011a or the CGY_1 analysis of Citko and Gračanin-Yüksek 2013 (see § 5.4.1 for details).

Such a test can be applied to Polish, though it does not seem very reliable as there are attested examples of multiple *wh*-questions which function as rhetorical questions. The following examples from NKJP were discussed in Przepiórkowski and Patejuk 2014:

(5.51) Kto kiedy zrozumie kobiety?
who.NOM when understand.FUT women
'Who will ever understand women?' (NKJP)

(5.52) Kto i kiedy rzucił takie hasło?
who and when threw such signal
'Who and when gave such a starting signal?' (NKJP)

Przepiórkowski and Patejuk 2014 observe that multiple questions such as (5.51) were found to have a rhetorical taste, while true questions featured lexico-semantic coordination, as in (5.52).

When applied to Polish data discussed in Przepiórkowski and Patejuk 2014, the observation of Paperno 2012 made for Russian is not reliable enough to be considered a test for choosing an appropriate syntactic representation of lexico-semantic coordination because the observation is based on a preference rather than a clear contrast – there are attested counterexamples, as demonstrated above.

Finally, a test based on the availability of rhetorical readings would be of very limited use as it would be inapplicable to most semantic classes which take part in lexico-semantic coordination – it could only be applied to *wh*-words.

5.4.2.6 Voice mismatch

Paperno 2012 discusses the following example[13] from West Armenian (a similar example is provided for Q'anjob'al (Mayan), see Paperno 2012, ex. (45)):

(5.53) ov jev vorun oknutjamp as ʃenk-əbidi ʃinvi
who.NOM and who.GEN help.INST this building-DEF FUT build.PASS.3SG
'Who and with whose help is this building going to build?' (Paperno 2012, ex. (43))

Paperno 2012 claims that this sentence is multiclausal because the verb form used in this example is passive while the first conjunct, *ov* marked for the nominative case, is expected to be the subject of an active sentence, not a passive one, as shown below:

(5.54) *ov as ʃenk-əbidi ʃinvi
who.NOM this building-DEF FUT build.PASS.3SG
'Who is going to build this building?' (Paperno 2012, ex. (44))

[13] Examples (5.53)–(5.54) were taken from Paperno 2012 without any changes in gloss alignment or translation.

As a result, Paperno 2012 treats this mismatch in voice as “a clear sign of a biclausal, elliptical structure”.

Though the passivisation test can be applied to Polish, its results are not very helpful when trying to determine the structure of the following example:

(5.55) *Kto i z czyją pomocą ten dom zostanie zbudowany?
who.NOM and with who.INST help.INST this house AUX.FUT built
‘Who and with whose help is this building going to build?’ (intended)

Though the sentence provided above is clearly ungrammatical, it is difficult to interpret such a negative result in a constructive way: while the voice mismatch test can be used to argue that some structure is multiclausal if such a mismatch is possible, failing this test does not provide evidence to the contrary – ellipsis is usually said to be possible under identity of relevant verbs, so it is not surprising to find it impossible without verb identity.

5.4.2.7 Obligatory arguments

Bîlbîie and Gazdik 2012 discuss an argument based on the fact that in certain languages some arguments are obligatory – they cannot be omitted. If a question featuring lexico-semantic coordination is grammatical with one ordering of conjuncts but ungrammatical when their order is changed, it suggests that the structure is multiclausal because one clause features an instance of argument dropping which is not allowed:

(5.56) Cine și unde locuiește?
who and where lives
(Bîlbîie and Gazdik 2012, ex. (15a))

(5.57) *Unde și cine locuiește?
where and who lives
(Bîlbîie and Gazdik 2012, ex. (15b))

According to Bîlbîie and Gazdik 2012 these examples suggest that while Romanian allows implicit subjects, the locative argument cannot be dropped. This explanation, however, does not seem sound under closer scrutiny: if lexico-semantic coordination was monoclausal in Romanian, (5.57) could only be ungrammatical due to word order preferences – this could be verified by removing the conjunction from examples (5.56) and (5.57) and checking the respective grammaticality judgements. No such examples are considered by Bîlbîie and Gazdik 2012, though a simple Google query seems to support the word order preferences hypothesis: there are no hits for the string *Unde cine locuiește?*, while there are numerous results for *Cine unde locuiește?*. It may therefore be the case that the reason for the ungrammaticality of (5.57) is the word order that is not accepted in Romanian in this configuration, which has nothing to do with the representation of such coordination.

If, however, multiclausal representation was assumed for lexico-semantic coordination in Romanian, the contrast between (5.56) and (5.57) would be surprising. If it was the case that *unde*, the locative argument, cannot be dropped, (5.56) and (5.57) would be expected to be equally ungrammatical under analyses which put question words into separate clauses (such as

T_1 and CGY_1). This is because the locative argument would be missing from one of the clauses: in (5.56) it would be absent from the first clause (where the verb is not realised overtly), while in (5.57) it would not be present in the second clause (where the verb is expressed overtly). Let us proceed to analyses which use movement from a multiple question: under T_2 (5.56)–(5.57) should be equally ungrammatical (locative argument cannot be implicit), while under CGY_2 these examples should be consistently grammatical or ungrammatical because both *wh*-phrases are shared by both clauses (see Figure 5.2 in § 5.4.1).

It seems therefore that it is difficult to argue in favour of the multiclausal analysis using the data of Bîlbîie and Gazdik 2012 – their argument seems flawed.

On the other hand, Paperno 2012 discusses similar data from Russian, arguing against the ellipsis analysis of hybrid coordination in (5.58):

(5.58) Kto i kuda napravljaetsja?
who.NOM and where.to is.directed
'Who is going where?' (Paperno 2012, ex. (56a))

(5.59) ??Kto napravljaetsja i kuda? (Paperno 2012, ex. (56b))

Paperno 2012 demonstrates that both arguments of the verb, the subject and the directional argument, are obligatory: see (5.60) and (5.61), respectively.

(5.60) *Kto napravljaetsja?
who.NOM is.directed
(Paperno 2012, ex. (38b))

(5.61) *Kuda napravljaetsja?
where.to is.directed
(Paperno 2012, ex. (38c))

When considering the examples presented above, it is perhaps worth mentioning that Russian, unlike Polish, does not freely allow subject pro-drop, although it does not entirely forbid it (see Paperno 2012, p. 95, fn. 5).

Following these facts, Paperno 2012 draws the conclusion that both arguments in (5.58) must be dependents of the same clause, so its representation is monoclausal. By contrast, (5.59) is less acceptable because its representation is multiclausal – both question words are dependents of different clauses, each of which misses an argument.

Since this test seems to be vulnerable to word order preferences (as suggested by the analysis of data from Romanian), it will not be taken into consideration in the discussion of Polish data.

5.4.2.8 Object agreement

A very convincing argument was advanced for Hungarian following Lipták 2001 by Bîlbîie and Gazdik 2012 (nearly identical examples were discussed by Paperno 2012 following Gazdik 2011):

(5.62) Mit készítesz és hogyan (készíted)?
what prepare.2.INDEF and how prepare.2.DEF
(Bîlbîie and Gazdik 2012, ex. (53a))

(5.63) Mit és hogyan készítesz / *készíted?
what and how prepare.2.INDEF prepare.2.DEF
'What are you preparing and how (are you preparing it)?'
(Bîlbîie and Gazdik 2012, ex. (53b))

(5.62) shows that *Mit*, the interrogative pronoun, triggers indefinite object agreement on the verb (first clause), while the verb in the second clause would bear definite object agreement – the lexical object is missing and it is interpreted as a definite implicit object. This difference in verb agreement suggests a multiclausal structure of (5.62).

By contrast, in (5.63) only indefinite object agreement is possible, which strongly suggests that the sentence has monoclausal structure – if there was ellipsis in the first conjunct (containing the interrogative pronoun, *Mit*), as in multiclausal analyses of lexico-semantic coordination involving deletion presented in § 5.4.1, the verb would be expected to display definite agreement (with an implicit object), which is not the case.

Lipták 2012 brings to light two important details which seem to have been omitted in other discussions of Hungarian object agreement in the context of the representation of coordinated *wh*-questions in spite of the fact that they are very relevant to this issue. First, Lipták 2012 shows that only singular objects may be dropped in Hungarian:

(5.64) Itt van a könyv. Péter már elolvasta pro_{sg}.
here is the book Péter already PV-read.DEF.3.SG
'Here is the book. Péter has read it.' (Lipták 2012, ex. (28a))

(5.65) Itt vannak a könyvek. *Péter már elolvasta pro_{pl}.
here are the book.PL Péter already PV-read.DEF.3.SG
'Here are the books. Péter has already read them.' (Lipták 2012, ex. (28b))

Secondly, Hungarian has a morphologically plural variant of 'what':

(5.66) Miket és hol javítottál meg?
what.PL.ACC and where repaired.INDEF.2.SG PV
(Lipták 2012, ex. (29))

Together, these facts provide an important argument against most[14] multiclausal analyses of the coordination of *wh*-words presented in § 5.4.1. This is because in examples such as (5.66) the first conjunct, *Miket*, is morphologically plural and only singular objects may be dropped in Hungarian.

Unfortunately, this test is inapplicable to Polish as it displays verb agreement with the subject but not with the object.

5.4.2.9 Distribution of question particles

Tomaszewicz 2011a mentions a test based on the distribution of question particles and discusses it using data from Romanian cited after Raţiu 2009:

[14] It seems that CGY_2, the bulk sharing analysis whereby *wh*-words are shared by both clauses, would survive.

(5.67) Oare cine ce va spune?
PART who what will say
(Tomaszewicz 2011a, ex. (13a))

(5.68) Oare cine *(și) oare ce va spune?
PART who and PART what will say
'Who will say something and what will he say?' (Tomaszewicz 2011a, ex. (13b))

As the examples show, in Romanian the question particle *oare* can be used only once with multiple *wh*-questions, but it can be used with each conjunct under lexico-semantic coordination. This suggests that while multiple *wh*-questions are monoclausal, the structure of lexico-semantic coordination of *wh*-words is multiclausal in Romanian.

While the test itself seems convincing, it cannot be applied directly to Polish as it does not use any question particle which could be a counterpart of the Romanian one. However, it is possible to consider the distribution of other elements which may occur only once per clause in Polish. Potential candidates include mood markers such as BY (conditional) and NIECH (imperative), the reflexive marker SIĘ, the negative particle NIE and agglutinate forms of the verb BYĆ 'be' (such as *-ś* in *Coś zrobił?* 'What have you done?').

To verify whether items listed above may be used with each *wh*-word conjunct under lexico-semantic coordination, the following base query was used for searching NKJP:[15]

(5.69)
```
[base="kto|co|gdzie|jak|kiedy" & (case=$1 | case!=".*")] VAR
i [base="kto|co|gdzie|jak|kiedy" & (case!=$1 | case!=".*")] VAR
```

`VAR` is a variable which is to be substituted (twice) for a relevant query element from the list provided above (mood marker, reflexive marker, negative particle, agglutinate verb form). The fragment `[base="kto|co|gdzie|jak|kiedy" & (case=$1 | case!=".*")]` matches a segment whose base form is KTO 'who', CO 'what', GDZIE 'where', JAK 'how' or KIEDY 'when' (`base="kto|co|gdzie|jak|kiedy"`); its case value is assigned to the `$1` variable (`case=$1`) or it has no case at all (`case!=".*"`). While `[base="kto|co|gdzie|jak|kiedy" & (case!=$1 | case!=".*")]` matches the same base forms, it requires that the case of this segment is not the same as the one assigned to the variable `$1` (`case!=$1`) or that the segment has no case whatsoever (`case!=".*"`).

A sample query resulting from substituting `by` for `VAR` in (5.69) is provided below:

(5.70)
```
[base="kto|co|gdzie|jak|kiedy" & (case=$1 | case!=".*")] by
i [base="kto|co|gdzie|jak|kiedy" & (case!=$1 | case!=".*")] by
```

Some constructed examples that would match the query in (5.70) are given below:

(5.71) *Kto by i kogo by uderzył?
who.NOM COND and who.ACC COND hit
'Who would hit whom?' (intended)

(5.72) *Kto by i kiedy by uderzył?

[15] As mentioned in fn. 12, the description of the query language of Poliqarp is available here: `http://nkjp.pl/poliqarp/help/ense3.html`.

(5.73) *Gdzie by i kiedy by uderzył?

The following table provides a short summary of the results of relevant NKJP queries (the entire corpus, NKJP1800M, was searched for results):

(5.74)

variable used in (5.69)	`by`	`niech`	`się`	`nie`	`[pos=aglt]`
NKJP results	0	0	0	0	0

The results summarised in (5.74) show that there is no evidence that it is possible to use elements which normally occur only once per clause (markers, particles, clitics) in Polish with each element of questions featuring lexico-semantic coordination of interrogative items. This suggests that there is no evidence supporting the multiclausal analysis.

While, in theory, it might be the case that counterexamples exist in larger text collections, constructed examples, such as in (5.71)–(5.73), are unacceptable.

5.4.2.10 Auxiliary between *wh*-phrases

Bîlbîie and Gazdik 2012 advance an argument based on the following example from Hungarian:

(5.75) Mit akarunk és hol vacsorázni?
what want.3 and where eat for dinner
'What do we want to eat for dinner and where?' (Bîlbîie and Gazdik 2012, ex. (48))

They claim that "it can be argued that *akar* 'want' is an auxiliary in Hungarian" because "it can interrupt the infinitive following it and appear between the verbal particle (if there is one) and the verbal stem" – if it is assumed, the argument goes, that the auxiliary and the main verb must belong to the same clause, it follows that the structure of such examples must be monoclausal.

Let us see how this test can be applied to Polish. Since CHCIEĆ 'want' is a control verb in Polish, forms of BYĆ 'be', the prototypical auxiliary, will be used instead:

(5.76) Kto będzie komu pomagać?
who.NOM AUX who.DAT help
'Who will help whom?'

(5.77) Komu będzie kto pomagać?
who.DAT AUX who.NOM help

(5.78) *Kto będzie i komu pomagać?
who.NOM AUX and who.DAT help

(5.79) *Komu będzie i kto pomagać?
who.DAT AUX and who.NOM help

Judgements for sentences (5.76)–(5.79) seem to find support in NKJP: the query `[base=kto & case=$1] [pos=bedzie] i [base=kto & case!=$1]` matches the coordination of two *wh*-words whose lemma is KTO 'who', requiring that the case of the first conjunct (assigned to the variable `$1`: `case=$1`) must be different than the case of the second conjunct (`case!=$1`).

This query yielded 0 results in the entire corpus. By contrast, its modified version with the conjunction removed did return one good result (more were found using Google), supporting judgements which accept splitting multiple *wh*-questions with an auxiliary:

(5.80) chodzi jednak o to, kto będzie komu służył
matters still about this who.NOM AUX who.DAT serve
'It's more about who will serve whom.' (NKJP)

(5.81) Pytanie kto będzie kogo spłacał. ?
question who.NOM AUX who.ACC pay
'The question is who is going to pay whom.' (Google)

It is not clear, however, how the results of this test should be interpreted, especially when other contexts are considered with respect to whether they allow being split with an auxiliary:

(5.82) Janek i Marysia będą biegli.
Janek and Marysia AUX.3.PL run.3.PL.M1
'Janek and Marysia will run.'

(5.83) *Janek będą i Marysia biegli.
Janek AUX.3.PL and Marysia run.3.PL.M1

Taking these examples into consideration, it seems[16] to be the case that coordinate phrases in Polish do not allow being split by an auxiliary in general. Lexico-semantic coordination also features a coordinate phrase, so the fact that splitting these with an auxiliary results in ungrammaticality is expected and is caused by reasons independent of whether such constructions are monoclausal or multiclausal.

5.4.2.11 Overt pronouns

Kazenin 2001 advances an argument in favour of the monoclausal analysis on the basis of coreference effects with overt pronouns:

(5.84) [Kogo$_i$ Petja izbil] i [za čto Petja ego$_i$/??pro$_i$ izbil]?
whom Peter beat and for what Peter him beat
'Whom did Peter beat and what for did Peter beat him?' (Kazenin 2001, ex. (50))

(5.85) *Kogo$_i$ i za čto Petja ego$_i$ izbil?
whom and for what Peter him beat
(Kazenin 2001, ex. (52))

Kazenin 2001 notes that under a coordination of two questions such as in (5.84) the *wh*-word in the first clause (*Kogo*) may be coreferential with an overt pronoun (*ego*) or, though this

[16] Sentences with a coordinate phrase consisting of predicative adjectives seem to be grammatical, though:

(i) Miły będzie i czuły.
kind AUX and tender
'He will be kind and tender.'

In such examples, however, BYĆ is a lexical raising verb, not an auxiliary.

option is glossed as worse, an implicit pronominal (pro) in the second clause. By contrast, an overt pronoun coreferential with one of the conjuncts cannot be used under lexico-semantic coordination, as shown in (5.85). According to Kazenin 2001, this difference suggests that the structure of such examples is monoclausal – unlike in (5.84), *Kogo* and *za čto* belong to the same clause in (5.85), which makes it impossible to use an overt pronoun as the object of *izbil* – this argument position is already filled by *Kogo*.

This test is applicable to Polish and the facts are very similar to Russian:

(5.86) Kogo i za co Piotr (*go) zbił?
who.ACC and for what Piotr he.ACC beat
'Who did Piotr beat and what did Piotr beat him for?'

However, note that this test does not exclude multiclausal analyses (see § 5.4.1): under T_1 and CGY_1 null pronouns could be claimed to block the use of lexical pronouns, while under T_2 there is a multiple *wh*-question in the second clause and CGY_2 uses multidominance, whereby the multiple *wh*-question is shared by both clauses. On the other hand, there seems to be no constructive evidence which would support using these multiclausal analyses instead of the monoclausal one.

5.4.2.12 Left-branch extraction (LBE)

Tomaszewicz 2011b proposes a syntactic argument in support of the multiclausal analysis based on the unavailability of a certain type of extraction in this environment, namely left-branch extraction (LBE). This name is used to refer to the phenomenon whereby the leftmost constituent of an NP can be extracted out of this NP, consider the examples below:

(5.87) Marek kupił jaki samochód?
Marek bought what.ACC car.ACC
'What car did Marek buy?'

(5.88) Jaki samochód kupił Marek?

(5.89) Jaki kupił Marek samochód?

In (5.87) the object containing an interrogative modifier (*jaki*) is not fronted. While in (5.88) the entire object is fronted, in (5.89) it is only the interrogative modifier of the object that is fronted, while the head nominal is in the same position as in (5.87) – this is an instance of LBE.

The argument of Tomaszewicz 2011b is that while LBE is grammatical with multiple questions (see (5.90)), which are monoclausal, such extraction is ungrammatical when lexico-semantic coordination is involved (compare (5.91)). Tomaszewicz 2011b attributes this alleged contrast in grammaticality to the fact that the structure of lexico-semantic coordination is multiclausal.

(5.90) Jaki kto kupił samochód swojej żonie?
which.ACC who.NOM bought car.ACC SELF.DAT wife.DAT
(Tomaszewicz 2011b, ex. (27a))

(5.91) *Jaki i kto kupił samochód swojej żonie?
which.ACC and who.NOM bought car.ACC SELF.DAT wife.DAT
(Tomaszewicz 2011b, ex. (27b))

However, judgements in Tomaszewicz 2011b are dubious – counterexamples may be found in the literature discussing similar phenomena:

(5.92) Jakie i skąd zdobywał informacje?
what.ACC and from where obtained information.ACC
'What information and where from did he obtain?' (Kallas 1993, p. 141, ex. (108))

Moreover, numerous attested examples may be found:

(5.93) Jakie i kto miał rzucane kłody pod nogi?
what.ACC and who.NOM had thrown logs.ACC under legs
'Who has been put what obstacles in their way?' (NKJP)

(5.94) Czy wiadomo jaki i kto będzie grał schwarzcharakter?
PART known what.ACC and who.NOM AUX play villain.ACC
'Do we know who is going to play which villain?' (NKJP)

(5.95) Jakie i kto podjął w tej sprawie działania?
what.ACC and who.NOM took in this matter actions.ACC
'Who took what action in this matter?' (Google)

(5.96) Jakie i kto może ponieść konsekwencje?
what.ACC and who.NOM can bear consequences.ACC
'Who can suffer what consequences?' (Google)

Since the examples listed above provide rich counterevidence to the judgements of Tomaszewicz 2011b, the conclusion drawn on the basis of her judgements does not hold – there is no contrast in grammaticality between LBE with multiple *wh*-questions and under lexico-semantic coordination, so there is no reason to claim that the latter is multiclausal.

It must be noted, however, that undermining the argument of Tomaszewicz 2011b does not provide strong, constructive evidence in support of the monoclausal representation of lexico-semantic coordination: LBE is possible in this environment, whatever the representation.

5.4.2.13 Stranding

Paperno 2012 advanced an argument in favour of the monoclausal analysis of lexico-semantic coordination based on the fact that there is a Russian pronoun, namely *čto*, which requires adjectival modifiers to appear in a non-agreeing genitive case form, unlike other nominals which take adjectival modifiers fully agreeing in case. Paperno 2012 offers a test based on the phenomenon of stranding,[17] which is, in transformational terms, a kind of "partial *wh*-movement", and illustrates it using the following example from Russian:

(5.97) Čto i komu on xorošego sdelal?
what.ACC and who.DAT he good.GEN did
'What good did he do, and to whom?' (Paperno 2012, ex. (49))

[17] Examples featuring stranding discussed here are different from LBE discussed in § 5.4.2.12 in that the latter involve extracting the modifier from the NP, while the former involve extraction of the head of the NP. The common feature in these two cases is that it is always the interrogative element that is extracted.

Paperno 2012 argues that the structure of this example must be non-elliptical (monoclausal) due to the fact that the adjectival modifier *xorošego* must depend on – and hence belong to the same clause as – the first conjunct, *Čto*. He provides the following example to demonstrate that the modifier cannot occur in the non-agreeing genitive form on its own:

(5.98) *Komu on xorošego sdelal?
who.DAT he good.GEN did
'To whom did he do good?' (Paperno 2012, ex. (50))

Furthermore, Paperno 2012 shows that this argument is immune to those multiclausal analyses which claim that there is ellipsis in one of the conjuncts coupled with the use of an indefinite pronoun (as in T_1, for instance):

(5.99) Komu on čto-libo xorošee sdelal?
who.DAT he something good.ACC did
'To whom did he do something good?' (Paperno 2012, ex. (51))

This example features an indefinite pronoun, *čto-libo*, which triggers full modifier agreement. The multiclausal analysis of Tomaszewicz 2011a features an indefinite pronoun represented as *something*, see (5.23) and (5.25), which correspond to (5.22). However, assuming that the indefinite pronoun of Tomaszewicz 2011a behaves in the same way as *čto-libo*, (5.97) could not be an instance of ellipsis of an indefinite pronoun – if this was the case, the modifier would be expected to appear in the agreeing form, as in (5.99).

The facts in Polish are similar: while Polish adjectival modifiers usually fully agree in case with their nominal heads, there are certain pronominal forms ((non-)agreement depends on the case of the head) which require the modifier to appear in a non-agreeing case, namely genitive. The following example is analogous to the one provided in (5.97):

(5.100) Co i komu ona ciekawego/*ciekawe powiedziała?
what.ACC and who.DAT she interesting.GEN/ACC said
'What interesting did she say, and to whom?'

There are, however, certain differences with respect to facts from Russian, as Polish has more pronouns which display the (non-)agreement pattern shown above for the interrogative *co*. This class also includes elements such as *coś* (indefinite), *cokolwiek* (free choice pronoun) and *nic* (*n*-word). Since, unlike in Russian, such (non-)agreement is possible in Polish with the indefinite pronoun (*coś*, see (5.101)), this test is vulnerable to claims that one of the conjuncts features an implicit indefinite pronoun (as in T_1).

(5.101) *(Coś) ciekawego się stało.
something.NOM interesting.GEN REFL happened
'Something interesting happened.'

While this test does not provide a definite argument against multiclausal analyses (it seems that at least T_2 and CGY_2 would be technically able to account for such data), again, there seems to be no motivation to use such accounts instead of the monoclausal analysis.

5.4.2.14 Governing numerals

Examples such as the following might provide new[18] evidence supporting monoclausal analyses of lexico-semantic coordination:

(5.102) Kto, ile i kiedy dostał unijnych dotacji?
who how much.ACC and when got EU subsidies.GEN
'Who got how much EU subsidies and when?' (NKJP)

(5.103) Nie wiem w ogóle, ile i kiedy dostanę pieniędzy na naszą działalność.
NEG know at all how much.ACC and when get.FUT money.GEN for our operation
'I have no idea how much money I will get for our operation and when.' (NKJP)

Both examples provided above contain governing numeral forms – as discussed in § 3.1.2.1, the distinctive feature of such forms is that they assign the genitive case to the accompanying nominal: the head numeral *ile* is marked for the accusative case (structural case assigned by the verb) while its nominal object bears the genitive case: *dotacji* in (5.102) and *pieniędzy* in (5.103).

This feature of governing numerals makes it is difficult to argue that ellipsis is at work in such examples because their hypothetical multiclausal base sentences would lack identity across clauses, as shown in (5.104), a multiclausal paraphrase of (5.103):

(5.104) Nie wiem w ogóle, ile pieniędzy dostanę i kiedy dostanę pieniądze.
NEG know at all how much.ACC money.GEN get.FUT and when get.FUT money.ACC
'I have no idea how much money I will get and when I will get the money.'

(5.104) shows that ellipsis analyses which postulate deletion under identity in the first clause (such as T_1) are impossible in such cases due to the fact that the case found in the second clause (accusative *pieniądze* required by the verb as structural case in this context) does not match the case found in the first clause (genitive *pieniędzy* required by the numeral head *ile*). If the example using lexico-semantic coordination, (5.103), was multiclausal, the genitive *pieniędzy* would be unexpected as the numeral (*ile*) would be placed in the first clause, while the verb in the second clause requires an object marked for the accusative case (*pieniądze*, as in (5.104)).

Finally, though theoretically the verb DOSTAĆ 'get' can assign the genitive case (as a realisation of structural case) to its object under the partitive reading, this does not seem to be an option in (5.102)–(5.103). Such interference can be eliminated by using predicates such as ROZWIĄZAĆ 'solve', where such a reading is unavailable (as shown by the contrast between (5.105) and (5.106)). (5.107) is a constructed example featuring lexico-semantic coordination where the partitive reading is not possible:

(5.105) Kto rozwiązał zadania?
who.NOM solved tasks.ACC
'Who did solve the tasks?'

(5.106) *Kto rozwiązał zadań?
who.NOM solved tasks.GEN
'Who did solve some tasks?' (intended)

(5.107) Ile i kto rozwiązał zadań?
how many.ACC and who.NOM solved tasks.GEN
'How many tasks did who solve?'

[18] This argument is, however, similar to stranding presented in § 5.4.2.13.

While other multiclausal accounts such as T_2 and CGY_2 could probably handle such data technically, there seems to be no reason which would justify adopting these accounts instead of the monoclausal analysis.

5.4.2.15 Binding

Lipták 2012 discusses the following example[19] from Polish, attributing it (together with judgements) to Citko 2013:[20]

(5.108) Który profesor (*i) ilu ze swoich studentów przeegzaminował?
which professor and how many of his students examined
'Which professor examined how many of his students?'
(Lipták 2012, endnote 15, ex. (ii))

According to Lipták 2012, this example is Citko's "argument for the presence of two CP projections hosting the wh-phrases" because "the variable inside the second wh-phrase cannot be bound by the first wh-phrase, unlike in multiple fronting", which, according to judgements provided in the example above, leads to ungrammaticality when the conjunction is present.

However, Lipták 2012 notes that her informants do not support Citko's judgements – they did not confirm the difference in grammaticality which would depend on the presence of the conjunction, as claimed by Citko. Lipták 2012 adds that the same judgements, namely supporting no difference in grammaticality between the multiple *wh*-question and question featuring coordination of *wh*-words in (5.108), hold in Bulgarian, Romanian and Hungarian.

Since binding is possible under lexico-semantic coordination – judgements of Citko 2013 presented in (5.108) are not confirmed by native speakers – the argument for the necessity of multiclausal representation in this context does not hold.

The fact that binding is possible in this environment does not, however, prove the opposite – it does not provide any constructive evidence that the representation of such examples must be monoclausal. It is possible to imagine a multiclausal analysis where sentences such as (5.108) contain a null subject in the second clause which binds the anaphor locally. To account for the fact that semantically it is *profesor* that binds the anaphor *swoich*, it could be argued that the implicit subject in the second clause is coindexed with the overt subject in the first clause.

Furthermore, as mentioned above, Lipták 2012 reports that examples such as (5.108) with the conjunction present are good in Romanian, which casts doubts as to whether anaphoric binding can prove that lexico-semantic coordination is monoclausal in a given language – there are arguments such as the distribution of the question particle *oare* (see § 5.4.2.9 for discussion), which strongly suggest multiclausal representation of such coordination in Romanian. This seems to further undermine binding as an argument in favour of the monoclausal analysis of lexico-semantic coordination.

As a result, it is concluded that while binding is possible under lexico-semantic coordination in Polish, this does not constitute evidence supporting either analysis: monoclausal or multiclausal.

[19] Polish diacritics, missing from the original example, were restored in (5.108).

[20] The time of publication of Lipták 2012 and Citko 2013 does not reflect the order in which these were written.

5.4.2.16 Coordination with *yes/no* question particle

Polish *yes/no* question particle CZY can be coordinated with *wh*-words:

(5.109) Tytuł brzmiał prosto i uczciwie: "Czy i jaki jest Bóg"
title sounded simply and honestly PART and what is God
'The title sounded simple and honest: "Does God exist and what is he like?"'
(NKJP)

(5.110) Nie wiemy wreszcie, czy i co kto chowa w rękawie.
NEG know besides PART and what.ACC who.NOM hides in sleeve
'Besides, we don't know if they got something up their sleeves and who keeps what up their sleeve.' (NKJP)

While such examples are common and their grammaticality is rather uncontroversial, it is worth noting that removing the conjunction results in ungrammaticality:[21]

(5.111) *Czy co kto chowa w rękawie?
PART what.ACC who.NOM hides in sleeve

This suggests that the *yes/no* question particle CZY cannot be used with *wh*-words as dependents of the same predicate.[22] However, sentences where *wh*-words depend on a different predicate are grammatical, as shown below:

(5.112) Czy wiesz, co jesz?
PART know what.ACC eat
'Do you know what you are eating?' (NKJP)

(5.113) Kto wie, czy Abraham nie był czarny?
who.NOM knows PART Abraham NEG was black
'Who knows whether Abraham was not black?' (NKJP)

In these examples the *yes/no* question particle CZY and *wh*-words belong to distinct clauses. In (5.112) *Czy* is placed in the main clause, where WIEDZIEĆ 'know' is the main verb, while *co* is the object of JEŚĆ 'eat' in the subordinate clause. By contrast, in (5.113) *Kto* is the subject of the main verb (WIEDZIEĆ), while the *yes/no* question particle *czy* belongs to the subordinate clause (featuring BYĆ 'be'). As a result, these examples satisfy the requirement that there be no *wh*-words in the clause which contains CZY.

If this constraint is accepted, it follows that the structure of lexico-semantic coordination featuring CZY as one of the conjuncts cannot be monoclausal. However, this constraint is satisfied under multiclausal analyses where CZY and *wh*-words never belong to the same clause (these include T_1 and CGY_1 discussed in § 5.4.1).

[21] (5.111) can be judged as grammatical under the reading where *co* and *kto* are interpreted as indefinite pronouns (existential). This, however, does not affect the presented argument, since it is concerned with the interpretation where these are *wh*-words.

[22] This observation was also made by Tomaszewicz 2011a: "In Polish the clause-initial marker *czy* cannot co-occur with wh-phrases, yet it is allowed in Coordinated-WHs, which provides evidence for the clausal character of the conjuncts."

Lexico-semantic coordination with CZY is a special case due to the fact that removing the conjunction in other lexico-semantic environments does not lead to ungrammaticality – the result of such an operation with coordinated *wh*-words is a monoclausal structure, a multiple question. However, this is not possible with CZY, which provides the only constructive argument in favour of adopting a multiclausal analysis – it is at the same time the only environment where the monoclausal analysis is not appropriate.

5.4.3 Summary and conclusion

The previous subsection provided a critical review of selected arguments for monoclausal or multiclausal representation of lexico-semantic coordination applied to Polish. It showed that, while there is evidence suggesting that structures with CZY should be analysed as multiclausal in Polish (see § 5.4.2.16), there is no evidence supporting such an analysis when CZY is not involved in such coordination. On the other hand, while it was demonstrated that some multiclausal analyses could not account for some phenomena considered, there seems to be no strong evidence which would make it possible to reject the remaining multiclausal accounts.

Some multiclausal analyses use ellipsis (such as T_1 and T_2), but it is possible to argue against them since they postulate ellipsis under identity. However, if the identity requirement is abandoned, ellipsis becomes an extremely powerful operation, which is starkly visible when considering phenomena such as gapping – for instance, the head of a clause may be removed and there seems to be no requirement of strict identity of verb forms (singular *lubi* vs plural *lubią* in (5.114)); besides, the dependent of the gapped clause may bear different case than in the full clause (accusative *Marysię* vs genitive *Marysi*, triggered by negation – realisations of structural case assigned to the object, see § 3.2.3):

(5.114) Janek lubi Marysię, a jego rodzice nie (lubią Marysi).
Janek like.3.SG Marysia.ACC and his parents NEG like.3.PL Marysia.GEN
'Janek likes Marysia, but his parents don't (like Marysia).'

The example featuring gapping serves to show that multiclausal analyses assuming ellipsis can be saved by stipulating the use of extra devices (such as the use of implicit pronouns) to account for relevant data.

While both analyses, monoclausal and multiclausal, are available in theory, it seems preferable to choose the more economic and simple analysis if there is no reason to do otherwise. As a consequence, the monoclausal analysis emerges as the default analysis – it does not require the use of implicit pronouns and coindexing, it does not use ellipsis mechanisms which are hard to justify in other syntactic contexts, it does not require multidominance. The multiclausal analysis seems to be motivated only for cases when one of the conjuncts is the *yes/no* question particle CZY (see § 5.4.2.16).

Such a split analysis of Polish lexico-semantic coordination is presented in the rest of this chapter: the multiclausal analysis is only used for coordination with CZY as one of the conjuncts, while the monoclausal analysis is used elsewhere.

5.5 Formalisation

Lexical entries of pronouns of a particular semantic type bear the attribute TYPE,[23] which may take the following values (for examples discussed so far): ANY (free choice pronoun; cf. (5.2)), ALL (universal quantifier; cf. (5.3)), NEG (*n*-word; cf. (5.4) repeated below as (5.115)) or INT (question word; cf. (5.5), (5.6) and (5.8)). This feature has independent motivation: it is used for the purposes of handling direct and embedded questions, free relatives and negative concord.[24]

(5.115)

nikogo	i	nic	nie	może	tłumaczyć.
nobody.GEN	and	nothing.NOM	NEG	can	excuse

‘Nothing can excuse anybody.’ (NKJP)

Simplified lexical entries of selected *n*-words (including words from (5.115)) are provided below:

(5.116) nic N (↑ PRED)=‘NOTHING’
(↑ CASE)= NOM
(↑ TYPE)= NEG

(5.117) nigdy ADV (↑ PRED)=‘NEVER’
(↑ TYPE)= NEG

(5.118) nikogo N (↑ PRED)=‘NOBODY’
(↑ CASE)= GEN
(↑ TYPE)= NEG

(5.119) nigdzie ADV (↑ PRED)=‘NOWHERE’
(↑ TYPE)= NEG

Such elements rewrite to corresponding phrases (as in (5.120)) and then, using parameterised c-structure rules, they are rewritten to phrases whose name contains, apart from category, a parameter whose value corresponds to its semantic type (represented as a subscript in italics):

(5.120) NP → N

(5.121) NP_{neg} → NP
(↓ TYPE)$=_c$ NEG

(5.122) ADVP → ADV

(5.123) ADVP_{neg} → ADVP
(↓ TYPE)$=_c$ NEG

Parameters make it possible to use such information about the semantic type of relevant lexical items at the level of c-structure without resorting to checking f-structure attributes – they may be used to ensure that certain categories in a given rule represent the same type:

(5.124) XPextr_{type} → XP_{type}
(↑ XPATH GF^{+})=↓

The rule in (5.124) is also independently motivated as it is used for the purposes of handling extraction (see § 2.4 for discussion). Its left-hand side rewrites to a disjunction of phrases of the same type; the XP_{type} category used in (5.124) is in fact a metacategory (see fn. 33 in § 4.7.2.3); its expansion rule is provided in (5.125), while the definition of allowed types is given in (5.126):[25]

(5.125) $\mathrm{XP}_{type} \equiv \{\mathrm{NP}|\mathrm{PP}|\mathrm{ADVP}|\mathrm{AP}\}_{type}$

(5.126) $type \equiv \{\ all \mid any \mid int \mid neg\ \}$

[23] The *yes/no* question particle CZY does not have the TYPE attribute, since it does not have a PRED attribute of its own. Instead, it introduces the CLAUSE-TYPE attribute to the f-structure of the relevant verb (see the structures in § 5.5.2).

[24] See Patejuk and Przepiórkowski 2014b for a discussion of the licensing of negative concord in Polish.

[25] Note that ‘≡’ does not indicate template definitions here, as it did in § 2.2.3, but is rather used to define abbreviations for some expressions.

The annotation attached to XP$_{type}$ in (5.124) makes it possible for dependents representing relevant semantic types to appear at the level of c-structure outside the clause containing their f-structure head (see the discussion of (5.129)). There are two important elements of this annotation: XPATH, defined in (5.127), provides the extraction path, while GF, defined in (5.128), corresponds to grammatical functions which may be assigned to the relevant element:[26]

(5.127) $\text{XPATH} \equiv \text{XCOMP}^{*}$

(5.128) $\text{GF} \equiv \{\text{SUBJ}|\text{OBJ}|\text{OBJ}_{\theta}|\text{OBL}|\text{ADJ} \in\}$

Together, these allow the dependent to be extracted[27] from infinitival clauses:

(5.129)	uśmiecha	się	nieśmiało,	bo	nikogo	nie	chce	krępować
	smiles	REFL	shyly	because	nobody.GEN	NEG	wants	intimidate

‘She smiles shyly as she does not want to intimidate anybody.’ (NKJP)

In (5.129) it is *nikogo* that undergoes extraction: even though it belongs at the level of c-structure to the clause containing the verb *chce*, it is an argument (OBJ) of the embedded infinitival clause (XCOMP) headed by the verb *krępować*. This is formalised as the following annotation[28] of the extracted element which corresponds to XP$_{type}$ in (5.124):

(5.130) $(\uparrow \text{XCOMP OBJ}){=}\downarrow$

The constraint assigning a grammatical function to the extracted phrase used in (5.124), $(\uparrow \text{XPATH GF}^{+}){=}\downarrow$, has a path which includes two variables: XPATH defined in (5.127), which is realised as XCOMP in (5.130), and GF defined in (5.128), which rewrites to OBJ above.

The following f-structure is obtained for (5.129):

(5.131)
$$\left[\begin{array}{ll}
\text{PRED} & \text{‘SMILE}\langle\boxed{1}\rangle\text{’} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right] \\
\text{ADJ} & \left\{ \left[\begin{array}{ll}\text{PRED} & \text{‘SHYLY’}\end{array}\right], \left[\begin{array}{ll}
\text{PRED} & \text{‘BECAUSE}\langle\boxed{2}\rangle\text{’} \\
\text{COMP} & \boxed{2}\left[\begin{array}{ll}
\text{PRED} & \text{‘WANT}\langle\boxed{3},\boxed{4}\rangle\text{’} \\
\text{SUBJ} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right] \\
\text{XCOMP} & \boxed{4}\left[\begin{array}{ll}
\text{PRED} & \text{‘INTIMIDATE}\langle\boxed{3},\boxed{5}\rangle\text{’} \\
\text{SUBJ} & \boxed{3} \\
\text{OBJ} & \boxed{5}\left[\begin{array}{ll}
\text{PRED} & \text{‘NOBODY’} \\
\text{CASE} & \text{GEN} \\
\text{TYPE} & \text{NEG}
\end{array}\right]
\end{array}\right] \\
\text{NEG} & +
\end{array}\right]
\end{array}\right] \right\}
\end{array}\right]$$

[26] The range of grammatical functions defined in (5.128) is restricted so as to only include those which are used in further discussion. However, note that it does not include all the grammatical functions used by the Polish LFG grammar – COMP and XCOMP are excluded, which has the effect of precluding clausal complements (infinitival and sentential) from being the target of extraction.

[27] As explained in fn. 14 in § 2.4, the use of the term *extraction* does not mean that extraction (illustated in (5.132)–(5.133); as opposed to scrambling) is involved. This term is used in this work to refer to LDDs.

[28] Note that, for the sake of simplicity, (5.130) does not use the UDF discourse function mentioned in § 2.4. However, it is used later, starting with § 5.6.5.

Sometimes, however, dependents may be extracted from sentential complements:

(5.132) Kogo powiedziała, że nie chce więcej widzieć?
who.GEN said that NEG wants more see
‘Who did she say she does not want to see anymore?’

(5.133) a na konwencji PiS to na kogo powiedziała że zagłosuje?
and on convention PiS FOCUS on who said that vote.FUT
‘And on the convention of PiS, who did she say she would vote for?’ (Google)

In (5.132) the *wh*-word *kogo* is placed in the main clause while in terms of f-structure it is an argument (OBJ) of *widzieć*, the infinitival complement (XCOMP) of *chce*, the sentential complement (COMP) of the main verb *powiedziała* – see the f-structure in (5.134):

(5.134)
$$\left[\begin{array}{ll}
\text{PRED} & \text{‘SAY}\langle\boxed{1},\boxed{2}\rangle\text{’}\\
\text{SUBJ} & \boxed{1}\left[\text{PRED}\quad\text{‘PRO’}\right]\\
\text{COMP} & \boxed{2}\left[\begin{array}{ll}
\text{PRED} & \text{‘WANT}\langle\boxed{3},\boxed{4}\rangle\text{’}\\
\text{SUBJ} & \boxed{3}\left[\text{PRED}\quad\text{‘PRO’}\right]\\
\text{XCOMP} & \boxed{4}\left[\begin{array}{ll}
\text{PRED} & \text{‘SEE}\langle\boxed{3},\boxed{5}\rangle\text{’}\\
\text{SUBJ} & \boxed{3}\\
\text{OBJ} & \boxed{5}\left[\begin{array}{ll}
\text{PRED} & \text{‘WHO’}\\
\text{CASE} & \text{GEN}\\
\text{TYPE} & \text{INT}
\end{array}\right]\\
\text{ADJ} & \left\{\left[\text{PRED}\quad\text{‘ANYMORE’}\right]\right\}
\end{array}\right]\\
\text{COMP-FORM} & \text{ŻE}\\
\text{NEG} & +
\end{array}\right]
\end{array}\right]$$

To account for such data, the extraction path defined in (5.127) is extended for relevant items, namely for (phrases containing) *wh*-words:[29,30]

(5.135) $\text{XPATH} \equiv \{\text{COMP}|\text{XCOMP}\}^*$

Note that in (5.124) GF is followed by Kleene plus, GF^+, which means that at least one instance of GF must be used. Paths consisting of more than one GF are used, for instance, when the relevant item is a modifier of an argument, as in (5.136), where *Jaki* is a modifier of *samochód*, the object of *kupić*, which is the infinitival complement of *chciał*. In order to produce the f-structure in (5.137) for (5.136), XPATH must be realised as XCOMP, while the path corresponding to GF^+ is OBJ ADJ.

[29] Since closer investigation of Polish extraction phenomena remains outside of the scope of this work, the provided extraction path is trivial. To account for attested data, it may require certain adjustments, including imposing additional constraints on some of its parts.

[30] The redefined extraction path provided in (5.135) should not cause problems with licensing of negative concord – though (5.135) includes COMP, *n*-words have their own, independent path restricting the domain to be checked for the presence of negation. This path would not include COMP because in Polish negation is transferred in infinitival verb chains, but not across clauses (see § 3.2.3.2 and Patejuk and Przepiórkowski 2014b for discussion).

(5.136) Jaki chciał kupić samochód?
what.ACC wanted buy.INF car.ACC
‘What car did he buy?’

(5.137)
$$\left[\begin{array}{ll}
\mathrm{PRED} & \text{‘}\mathrm{WANT}\langle\boxed{1},\boxed{2}\rangle\text{’} \\
\mathrm{SUBJ} & \boxed{1}\left[\begin{array}{ll}\mathrm{PRED} & \text{‘}\mathrm{PRO}\text{’}\end{array}\right] \\
\mathrm{XCOMP} & \boxed{2}\left[\begin{array}{ll}
\mathrm{PRED} & \text{‘}\mathrm{BUY}\langle\boxed{1},\boxed{3}\rangle\text{’} \\
\mathrm{SUBJ} & \boxed{1} \\
\mathrm{OBJ} & \boxed{3}\left[\begin{array}{ll}
\mathrm{PRED} & \text{‘}\mathrm{CAR}\text{’} \\
\mathrm{CASE} & \mathrm{ACC} \\
\mathrm{ADJ} & \left\{\left[\begin{array}{ll}\mathrm{PRED} & \text{‘}\mathrm{WHAT}\text{’}\end{array}\right]\right\}
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

5.5.1 Monoclausal coordination

After particular conjuncts have been assigned appropriate functional annotation, they are fed into rules handling lexico-semantic coordination. The rule provided in (5.138) serves the purpose of handling sentences such as (5.2)–(5.6). Such sentences are assumed to have a monoclausal structure (see § 5.4 for discussion of which representation should be adopted), which is achieved by using the co-head functional annotation: ↑=↓.

(5.138)
$$\mathrm{XPlxm}_{type} \rightarrow \underset{\uparrow=\downarrow}{\mathrm{XPextr}_{type}} \quad [, \underset{\uparrow=\downarrow}{\mathrm{XPextr}_{type}}]^{*} \quad \underset{\uparrow=\downarrow}{\mathrm{CONJ}} \quad \underset{\uparrow=\downarrow}{\mathrm{XPextr}_{type}}$$

Let us see how (5.140), the f-structure corresponding to (5.5), repeated in (5.139) below, is constructed in a stepwise manner.

(5.139) Skąd i jakie otrzymujemy informacje?
whence and what.ACC receive information.ACC
‘What information and where from do we receive?’ (NKJP)

(5.140)
$$\left[\begin{array}{ll}
\mathrm{PRED} & \text{‘}\mathrm{RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{’} \\
\mathrm{SUBJ} & \boxed{1}\left[\begin{array}{ll}\mathrm{PRED} & \text{‘}\mathrm{PRO}\text{’}\end{array}\right] \\
\mathrm{OBJ} & \boxed{2}\left[\begin{array}{ll}
\mathrm{PRED} & \text{‘}\mathrm{INFORMATION}\text{’} \\
\mathrm{CASE} & \mathrm{ACC} \\
\mathrm{ADJ} & \left\{\left[\begin{array}{ll}\mathrm{PRED} & \text{‘}\mathrm{WHAT}\text{’} \\ \mathrm{CASE} & \mathrm{ACC} \\ \mathrm{TYPE} & \mathrm{INT}\end{array}\right]\right\}
\end{array}\right] \\
\mathrm{ADJ} & \left\{\left[\begin{array}{ll}\mathrm{PRED} & \text{‘}\mathrm{WHENCE}\text{’} \\ \mathrm{TYPE} & \mathrm{INT}\end{array}\right]\right\}
\end{array}\right]$$

The relevant lexical entries are provided below (note that only information used in the following f-structures is represented; for definitions of templates AGR-CASE and STRCASE called in (5.143), see (3.17) and (3.67), respectively):

(5.141) skąd ADV $(\uparrow \text{PRED}) = \text{‘WHENCE’}$
$(\uparrow \text{TYPE}) = \text{INT}$

(5.142) jakie A $(\uparrow \text{PRED}) = \text{‘WHAT’}$
$(\uparrow \text{CASE}) = \text{ACC}$
$(\uparrow \text{TYPE}) = \text{INT}$

(5.143) otrzymujemy I $(\uparrow \text{PRED}) = \text{‘OBTAIN}\langle(\uparrow \text{SUBJ})(\uparrow \text{OBJ})\rangle\text{’}$
AGR-CASE
STRCASE

(5.144) informacje N $(\uparrow \text{PRED}) = \text{‘INFORMATION’}$
$(\uparrow \text{CASE}) = \text{ACC}$

First, lexical entries of a particular type are rewritten using appropriate parameterised rules – below interrogative items (adverb, adjective) are rewritten to respective interrogative phrases (adverbial, adjectival):

(5.145) $$\text{ADVP}_{int} \rightarrow \underset{(\downarrow \text{TYPE}) =_c \text{INT}}{\text{ADV}}$$

(5.146) $$\text{AP}_{int} \rightarrow \underset{(\downarrow \text{TYPE}) =_c \text{INT}}{\text{A}}$$

In the next step particular conjuncts build their own partial f-structures thanks to the rule provided in (5.124) – it assigns each conjunct its own, independent functional annotation, which is crucial to the analysis as it makes it possible to avoid the problem of distributivity of grammatical functions under coordination: they are assigned to individual conjuncts rather than to the entire coordinate structure. Although the annotation in (5.124) is very general (it may in theory generate a path consisting of the extraction path XPATH and any (non-zero) sequence of grammatical functions GF), one must bear in mind that its output is constrained by the f-structure of the rest of the utterance. As a result, though (5.124) may generate infinitely many structures, only the following f-structures built by individual conjuncts may be unified with the rest of the sentence in (5.139): (5.147) is the rule which produces the f-structure in (5.148) for *skąd*, while (5.149) is the rule which yields the f-structure in (5.150) for *jakie*.

(5.147) $$\text{XPextr}_{int} \rightarrow \underset{(\uparrow \text{ADJ} \in) = \downarrow}{\text{ADVP}_{int}}$$

(5.148) $$\left[\begin{array}{ll} \text{ADJ} & \left\{ \begin{bmatrix} \text{PRED} & \text{‘WHENCE’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \right\} \end{array}\right]$$

(5.149) $$\text{XPextr}_{int} \rightarrow \underset{(\uparrow \text{OBJ ADJ} \in) = \downarrow}{\text{AP}_{int}}$$

(5.150) $$\left[\begin{array}{ll} \text{OBJ} & \left[\begin{array}{ll} \text{ADJ} & \left\{ \begin{bmatrix} \text{PRED} & \text{‘WHAT’} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{bmatrix} \right\} \end{array}\right] \end{array}\right]$$

Now it is time to use the rule in (5.138) to build the f-structure for the fragment corresponding to the entire phrase with lexico-semantic coordination – the particular instance of this rule used in (5.139), where the type is set to *int* and there are only two conjuncts, is given in (5.151):

(5.151) $$\text{XPlxm}_{int} \rightarrow \underset{\uparrow=\downarrow}{\text{XPextr}_{int}} \quad \underset{\uparrow=\downarrow}{\text{CONJ}} \quad \underset{\uparrow=\downarrow}{\text{XPextr}_{int}}$$

Since all conjuncts in (5.151) bear the co-head annotation (↑=↓), unlike under the standard account of coordination (using the ↓∈↑ annotation), no set is created. Instead, f-structure fragments built by particular conjuncts, (5.148) and (5.150), are unified in one f-structure, (5.152):[31]

(5.152)
$$\left[\begin{array}{ll} \text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{‘WHENCE’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\} \\ \text{OBJ} & \left[\begin{array}{ll}\text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{‘WHAT’}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]\end{array}\right]$$

Finally, using the top-level sentence rule in (5.154), the f-structure representing lexico-semantic coordination (*skąd i jakie*), (5.152), is unified with (5.155), the f-structure corresponding to *otrzymujemy informacje*, the remaining part of (5.139), built using the rule in (5.153) (the implicit subject is introduced elsewhere), to yield the full f-structure provided in (5.140):

(5.153)
$$\begin{array}{cccc} \text{IP} & \rightarrow & \text{I} & \text{NP} \\ & & \uparrow=\downarrow & (\uparrow \text{OBJ})=\downarrow \end{array}$$

(5.154)
$$\begin{array}{cccc} \text{S} & \rightarrow & \text{XPlxm}_{int} & \text{IP} \\ & & \uparrow=\downarrow & \uparrow=\downarrow \end{array}$$

(5.155)
$$\left[\begin{array}{lll} \text{PRED} & \multicolumn{2}{l}{\text{‘RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{’}} \\ \text{SUBJ} & \boxed{1} & \left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right] \\ \text{OBJ} & \boxed{2} & \left[\begin{array}{ll}\text{PRED} & \text{‘INFORMATION’}\\ \text{CASE} & \text{ACC}\end{array}\right] \end{array}\right]$$

The resulting f-structure corresponding to (5.139) provided in (5.140) is monoclausal – there is only one main predicate and all lexico-semantic conjuncts (*Skąd*, *jakie*) are unified in its f-structure, though not necessarily as dependents of the same predicate. While the first conjunct (*Skąd*) is a modifier of the main verb (*otrzymujemy*), the other conjunct (*jakie*) is the modifier of the object (*informacje*) of the main verb.

5.5.2 Multiclausal coordination

A slightly different coordination rule, provided in (5.156),[32] where the PART$_{int}$ category corresponds to the *yes/no* question particle CZY (see the lexical entry in (5.157) – it introduces the attribute CLAUSE-TYPE[33] taking the INT value, which signals that the clause in which CZY appears is interrogative), is designed for examples such as (5.8), repeated in (5.160), which are considered multiclausal, as discussed in § 5.4.2.16.

(5.156)
$$\begin{array}{cccccc} \text{XPlxb}_{int} & \rightarrow & \text{PART}_{int} & [,\ \ \text{XPextr}_{int}]^{*} & \text{CONJ} & \text{XPextr}_{int} \\ & & \downarrow\in\uparrow & \downarrow\in\uparrow & \uparrow=\downarrow & \downarrow\in\uparrow \end{array}$$

[31] Note that the contribution of the conjunction is omitted – this issue is discussed later, in § 5.6.2.

[32] This rule accounts for examples where the question particle is the first conjunct. Such examples seem to be most frequent; there exist, however, examples where *czy* serves as the last conjunct:

(i) Będą sprawdzać kto i czy miał zezwolenie
AUX check who and PART had permission
‘They will check whether (they had permission) and who had permission.’ (NKJP)

Such cases may be handled by applying simple word order modifications to the rule in (5.156).

[33] See § 4.5.2.2 for the discussion of (4.79), an example where one of the conjuncts of the object is an interrogative subordinate clause: constraints related to this argument are provided in (4.81); the f-structure corresponding to (a simplified version of) (4.79) is given in (4.83).

(5.157) czy PART$_{int}$ (↑ CLAUSE-TYPE)= INT

(5.158) kiedy ADV (↑ PRED)=‘WHEN’
(↑ TYPE)= INT

(5.159) wróci I (↑ PRED)=‘RETURN<(↑ SUBJ)>’
AGR-CASE

(5.160) Nie wiadomo było, czy *(i) kiedy wróci.
NEG know was PART and when returns
‘It was not clear whether and when he would return.’ (NKJP)

First, the interrogative adverb *kiedy* (see (5.158) for its lexical entry) is rewritten to an interrogative adverbial phrase using the rule in (5.145). Subsequently, the obtained phrase is rewritten to XPextr$_{int}$ category with the help of the rule in (5.124) in the same way as in (5.147), which constructs its partial f-structure in (5.162); the partial f-structure constructed by the lexical entry of *czy* (see (5.157)) is given in (5.161).

(5.161) $\left[\text{CLAUSE-TYPE} \quad \text{INT}\right]$

(5.162)
$$\left[\begin{array}{ll}\text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{‘WHEN’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]$$

Now, the rule in (5.156) can be applied. To represent the fact that utterances such as (5.160) are not monoclausal, all conjuncts bear the set membership annotation (↓∈↑). As a result, partial f-structures constructed by individual conjuncts provided in (5.161)–(5.162) are placed inside a set, as shown in (5.163):

(5.163)
$$\left\{\left[\text{CLAUSE-TYPE} \quad \text{INT}\right], \left[\begin{array}{ll}\text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{‘WHEN’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]\right\}$$

When, as a result of using the rule in (5.154), (5.164), the f-structure corresponding to *wróci* (constructed using (5.159) and (5.165); the implicit subject is introduced elsewhere), the rest of the subordinate clause in (5.160), is unified with (5.163), the f-structure corresponding to *czy i kiedy*, a multiclausal coordinate structure results, as shown in (5.166):[34]

(5.164)
$$\left[\begin{array}{ll}\text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’}\\ \text{SUBJ} & \boxed{1}\left[\text{PRED} \quad \text{‘PRO’}\right]\end{array}\right]$$

(5.165)
$$\begin{array}{ccc}\text{IP} & \to & \text{I}\\ & & \uparrow=\downarrow\end{array}$$

(5.166)
$$\left\{\left[\begin{array}{ll}\text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’}\\ \text{SUBJ} & \boxed{1}\left[\text{PRED} \quad \text{‘PRO’}\right]\\ \text{CLAUSE-TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’}\\ \text{SUBJ} & \boxed{1}\\ \text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{‘WHEN’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]\right\}$$

[34] Note that the implicit subject in (5.166), $\boxed{1}$, is structure-shared: it belongs to both clauses at the same time.

The structure in (5.166) is multiclausal because of the interaction of properties of its partial f-structures: (5.163) is a set and the PRED attribute in (5.164) is a distributive feature. When these partial structures are unified, (5.164) distributes to particular elements of the set in (5.163): ‘copies’ of (5.164) are unified with respective elements of the set in (5.163): the *yes/no* question particle *czy* and the adjunct *kiedy* ‘when’ – the result is the biclausal f-structure in (5.166).

5.5.3 Argument saturation under the multiclausal analysis

The multiclausal analysis of certain instances of lexico-semantic coordination forces the introduction of some changes in order to account for independent argument saturation in coordinated clauses. While modifications are not required by examples such as (5.160) – there is an intransitive predicate whose only argument is shared (the implicit subject in (5.166)) – argument saturation turns out to be an issue with sentences such as the following one:

(5.167) czy *(i) ile będzie mogła zarobić tego typu placówka?
PART and how much.ACC AUX be able earn such institution
‘Will such an institution be able to earn and how much will it be able to earn?’ (NKJP)

Let us consider (5.168), a simplified version of (5.167):

(5.168) czy *(i) ile zarobi placówka?
PART and how much.ACC earn institution
‘Will the institution earn and how much will it earn?’

(5.169) ile NUM (↑ PRED)=‘HOW_MUCH<(↑ OBJ)>’
(↑ CASE)= ACC
(↑ TYPE)= INT

(5.170) zarobi I (↑ PRED)=‘EARN<(↑ SUBJ)(↑ OBJ)>’
AGR-CASE
STRCASE

(5.171) placówka N (↑ PRED)=‘INSTITUTION’
(↑ CASE)= NOM

In (5.168) the *yes/no* interrogative particle (*czy*; see (5.157) for its lexical entry) is coordinated with *ile*, one of the arguments of the verb *zarobi*, which, according to its lexical entry in (5.170), is a two-place predicate, taking a subject and a direct object. The former, the subject, is overt (*placówka*) and it is shared by both clauses resulting from multiclausal representation of (5.168), so it is not a problem from the perspective of argument saturation. However, according to the analysis provided in § 5.5.2 above, dependents coordinated under multiclausal lexico-semantic coordination belong to different clauses. As a result, *ile* may only fill the object grammatical function of one of these clauses. To avoid the violation of the completeness principle (see § 2.2.2 for discussion), the object of the other clause must be filled in some other way. This can be achieved using the following statement to handle implicit argument saturation:

(5.172)
$$\begin{array}{lll}\text{PRODROP} & \equiv & ((\uparrow \text{SUBJ PRED})=\text{`PRO'}) \\ & & ((\uparrow \text{OBJ PRED})=\text{`PRO'}) \\ & & \ldots \\ & & ((\uparrow \text{GF PRED})=\text{`PRO'})\end{array}$$

The statement provided in (5.172) is a template consisting of a set of equations optionally (each of them is enclosed in brackets) filling a given grammatical function with an implicit argument (represented as the PRO value of its PRED attribute). The last line of (5.172), $((\uparrow \text{GF PRED})=\text{`PRO'})$, is a notational shortcut for all other appropriate grammatical functions, as defined in (5.128), with the exception of adjuncts (to avoid introducing implicit adjuncts).

It must be noted that the place of attachment of such statements is of importance – attaching (5.172) to the entire lexico-semantic coordinate structure would give rise to a shared implicit dependent, which can lead to violations of the uniqueness condition – a given grammatical function could be filled with a lexical dependent, leading to a clash with the implicit argument attempting to fill the same slot. This is illustrated for (5.168) in (5.173), where the implicit object (PRO introduced by (5.172)) is structure-shared by both clauses, but the object of the second clause is filled lexically at the same time (by *ile*)[35] – conflicting values of OBJ's PRED attribute are represented in (5.173) as inequality (‘PRO’≠‘HOW_MUCH’).

(5.173)
$$* \left\{ \begin{bmatrix} \text{PRED} & \text{`EARN}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`INSTITUTION'}\end{bmatrix} \\ \text{OBJ} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`PRO'} \\ \text{CASE} & \text{ACC}\end{bmatrix} \\ \text{CLAUSE-TYPE} & \text{INT} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{`EARN}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1} \\ \text{OBJ} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`PRO'}\neq\text{`HOW\_MUCH'} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT}\end{bmatrix} \end{bmatrix} \right\}$$

For this reason (5.172) must not be placed inside the rule adding conjuncts to a set,[36] it should instead be placed so that implicit arguments attach independently inside individual clauses. To achieve this, (5.172) should be attached as in (5.174): at an intermediate level, so that its partial f-structure is unified with the f-structure fragment constructed by a given conjunct – this way both grammatical function assignment and optional introduction of implicit arguments are independent for each conjunct.

(5.174)
$$\begin{array}{lcc}\text{XPextrbicl}_{type} & \rightarrow & \text{XPextr}_{type} \\ & & \uparrow=\downarrow \\ & & \text{PRODROP}\end{array}$$

Furthermore, care must be taken in order to ensure that conjuncts with prodrop statements are only used with multiclausal lexico-semantic coordination. One of possible means to this end is to introduce special categories for multiclausal conjuncts exclusively, as in (5.174) and (5.175):

(5.175)
$$\begin{array}{lcc}\text{PARTbicl}_{type} & \rightarrow & \text{PART}_{type} \\ & & \uparrow=\downarrow \\ & & \text{PRODROP}\end{array}$$

[35] (5.173) is simplified: the numeral is represented as a zero-place predicate for typographical reasons, so its implicit object is not represented. However, the following examples feature the full representation of the numeral.

[36] Similar problems are expected with lexical prodrop (when the implicit argument is introduced by the lexical entry of the verb): such implicit arguments would distribute to all coordinated clauses, leading to potential violations of the uniqueness condition.

Finally, the rule provided in (5.156) must be rewritten as in (5.176), replacing XPextr and PART categories with XPextrbicl and PARTbicl, respectively:

(5.176) $$\begin{array}{llllll} \text{XPlxb}_{int} & \rightarrow & \text{PARTbicl}_{int} & [,\ \text{XPextrbicl}_{int}]^{*} & \text{CONJ} & \text{XPextrbicl}_{int} \\ & & \downarrow\in\uparrow & \downarrow\in\uparrow & \uparrow=\downarrow & \downarrow\in\uparrow \end{array}$$

Let us now construct (5.186), the f-structure representing (5.168), stepwise to see the modifications discussed above at work.

First, individual conjuncts construct their partial f-structures: (5.180) corresponds to *czy* and it is built using the lexical entry in (5.157), while (5.181) is contributed by *ile* and it is constructed using the lexical entry in (5.169) and the following rules: (5.177) (it rewrites the numeral to an NP), (5.178) (it rewrites an interrogative NP to NP$_{int}$) and (5.124) (it rewrites a category of a particular type to XPextr of the same type and assigns the grammatical function – (5.179) shows how this rule was used for *ile* in (5.168)).

(5.177) $$\begin{array}{lll} \text{NP} & \rightarrow & \text{NUM} \\ & & \uparrow=\downarrow \end{array}$$

(5.178) $$\begin{array}{lll} \text{NP}_{int} & \rightarrow & \text{NP} \\ & & (\downarrow \text{TYPE})=_{c} \text{INT} \end{array}$$

(5.179) $$\begin{array}{lll} \text{XPextr}_{int} & \rightarrow & \text{NP}_{int} \\ & & (\uparrow \text{OBJ})=\downarrow \end{array}$$

(5.180) $$\left[\begin{array}{ll} \text{CLAUSE-TYPE} & \text{INT} \end{array}\right]$$

(5.181) $$\left[\begin{array}{ll} \text{OBJ} & \left[\begin{array}{ll} \text{PRED} & \text{`HOW\_MUCH}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{`PRO'} \end{array}\right] \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{array}\right] \end{array}\right]$$

Subsequently, according to the rules in (5.175) and (5.174), for *czy* and *ile*, respectively, optional implicit arguments can be added as a result of calling the template defined in (5.172) inside particular conjuncts. In (5.182) an implicit argument fills the object grammatical function in the f-structure fragment containing the contribution of the *yes/no* question particle *czy*:

(5.182) $$\left[\begin{array}{ll} \text{OBJ} & \left[\begin{array}{ll} \text{PRED} & \text{`PRO'} \\ \text{CASE} & \text{ACC} \end{array}\right] \\ \text{CLAUSE-TYPE} & \text{INT} \end{array}\right]$$

Next, conjuncts are added to a set using the modified rule handling multiclausal lexico-semantic coordination provided in (5.176):

(5.183) $$\left\{ \left[\begin{array}{ll} \text{OBJ} & \left[\begin{array}{ll} \text{PRED} & \text{`PRO'} \\ \text{CASE} & \text{ACC} \end{array}\right] \\ \text{CLAUSE-TYPE} & \text{INT} \end{array}\right], \left[\begin{array}{ll} \text{OBJ} & \left[\begin{array}{ll} \text{PRED} & \text{`HOW\_MUCH}\langle\boxed{1}\rangle\text{'} \\ \text{OBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{`PRO'} \end{array}\right] \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{array}\right] \end{array}\right] \right\}$$

Finally, (5.183), the partial f-structure built by lexico-semantic coordination (*czy i ile*), is unified with (5.184), the f-structure representing the rest of the sentence (*zarobi placówka*) in (5.168) built using the rule in (5.185), which yields the full f-structure in (5.186).

(5.184) $$\left[\begin{array}{ll} \text{PRED} & \text{`EARN}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{`INSTITUTION'} \end{array}\right] \\ \text{OBJ} & \boxed{2} \end{array}\right]$$

(5.185) $$\begin{array}{llll} \text{IP} & \rightarrow & \text{I} & \text{NP} \\ & & \uparrow=\downarrow & (\uparrow \text{SUBJ})=\downarrow \end{array}$$

(5.186)

$$\left\{ \left[\begin{array}{ll} \text{PRED} & \text{`EARN}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`INSTITUTION'}\end{array}\right] \\ \text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{ACC}\end{array}\right] \\ \text{CLAUSE-TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{`EARN}\langle\boxed{1},\boxed{3}\rangle\text{'} \\ \text{SUBJ} & \boxed{1} \\ \text{OBJ} & \boxed{3}\left[\begin{array}{ll} \text{PRED} & \text{`HOW\_MUCH}\langle\boxed{4}\rangle\text{'} \\ \text{OBJ} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT}\end{array}\right]\end{array}\right] \right\}$$

As indicated by appropriate structure-sharing of relevant f-structure fragments, the lexical subject (*placówka*) is shared by both clauses in (5.186), unlike their objects: the object of the first clause (it contains *czy*, the *yes/no* question particle, the first conjunct) is filled with an implicit argument, while the object of the other clause is filled with a lexical argument, the second conjunct (*ile*) – under the multiclausal analysis conjuncts must be placed in separate clauses.

5.6 Problematic issues

5.6.1 Which elements are coordinated?

The first problem with the representation of lexico-semantic coordination presented in § 5.5 is that it does not show which elements were coordinated – the underlying analysis assigns particular conjuncts to relevant fragments of f-structure without representing the lexico-semantic coordinate phrase in any way. To illustrate potential problems with such an analysis, let us consider the following examples:

(5.187)	Kto	i	jaki	sprawił	komu	prezent?
	who.NOM	and	what.ACC	made	who.DAT	gift.ACC
	'Who bought what kind of gift to whom?'					

(5.188)	Kto	i	komu	sprawił	jaki	prezent?
	who.NOM	and	who.DAT	made	what.ACC	gift.ACC

(5.189)	Kto	komu	jaki	sprawił	prezent?
	who.NOM	who.DAT	what.ACC	made	gift.ACC

In (5.187) it is *Kto* (the subject) and *jaki* (the modifier of the object *prezent*) that are coordinated using lexico-semantic coordination, while *komu* (the indirect object) is not part of such coordination. By contrast, in (5.188) lexico-semantic conjuncts include *Kto* and *komu*, while *jaki* does not take part in such coordination. Finally, there is no lexico-semantic coordination in (5.189) – it is a multiple question.

Under the analysis presented in § 5.5, the same f-structure would correspond to (5.187), (5.188) and, if the conjunction has no representation (see § 5.6.2), also to (5.189), making it impossible to see on the basis of the f-structure which interrogative elements take part in lexico-semantic coordination and which do not.

(5.190)
$$\left[\begin{array}{lll}
\text{PRED} & \text{`BUY}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} & \\
\text{SUBJ} & \boxed{1} & \left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right] \\
\text{OBJ} & \boxed{2} & \left[\begin{array}{ll}\text{PRED} & \text{`GIFT'}\\ \text{CASE} & \text{ACC}\\ \text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right] \\
\text{OBJ}_{\theta} & \boxed{3} & \left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{DAT}\\ \text{TYPE} & \text{INT}\end{array}\right]
\end{array}\right]$$

5.6.2 Representing the conjunction

The f-structures provided in § 5.5 do not include the contribution of the annotation of the conjunction. As mentioned in § 5.3, the form of the conjunction, namely whether it belongs to the conjoining or the alternative type, is of importance from the perspective of semantics. Such information is provided using a dedicated attribute, COORD-FORM for instance:[37]

(5.191) a. i CONJ (↑ COORD-FORM)= I

b. lub CONJ (↑ COORD-FORM)= LUB

When conjunctions annotated in this way are used with rules such as (5.138), the rule handling monoclausal lexico-semantic coordination, the conjunction is represented in the f-structure containing the relevant conjuncts. The fragment corresponding to *Skąd i jakie*, the lexico-semantic coordinate phrase from (5.5) (repeated as (5.192)), is provided in (5.193) – compare it with (5.152), where the conjunction was not represented.

(5.192)

Skąd	i	jakie	otrzymujemy	informacje?
whence	and	what.ACC	receive	information.ACC

'What information and from where do we receive?' (NKJP)

(5.193)
$$\left[\begin{array}{ll}
\text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\} \\
\text{OBJ} & \left[\begin{array}{ll}\text{ADJ} & \left\{\left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right] \\
\text{COORD-FORM} & \text{I}
\end{array}\right]$$

When the f-structure fragment in (5.193) is unified with the f-structure of the rest of the utterance (see (5.155)), the following f-structure results (a modified version of (5.140)):

[37] The attribute COORD-FORM was introduced in § 2.5 when discussing (2.69).

(5.194)

$$\begin{bmatrix}
\text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\end{bmatrix} \\
\text{OBJ} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`INFORMATION'}\\ \text{CASE} & \text{ACC}\\ \text{ADJ} & \left\{\begin{bmatrix}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\}\end{bmatrix}\\
\text{ADJ} & \left\{\begin{bmatrix}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\}\\
\text{COORD-FORM} & \text{I}
\end{bmatrix}$$

5.6.3 Interference with verbal coordination

Representation of the conjunction such as in § 5.6.2 is vulnerable to interference with clausal coordination – when verbs are coordinated, as in (5.195), the conjunction is represented in the top-level f-structure, see (5.196):

(5.195)

babcia	rozpaczała	i	płakała.
grandma	despaired	and	cried

‘Grandma despaired and cried.’ (NKJP)

(5.196)

$$\begin{bmatrix}
\left\{\begin{bmatrix}\text{PRED} & \text{`DESPAIR}\langle\boxed{1}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`GRANDMA'}\end{bmatrix}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`CRY}\langle\boxed{1}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\end{bmatrix}\right\}\\
\text{COORD-FORM} \quad \text{I}
\end{bmatrix}$$

As a result a problem arises when lexico-semantic coordination co-occurs with coordination of verb forms, as in (5.197) (a modified version of (5.5)) and attested (5.198):

(5.197)

[Skąd	i	jakie]	[otrzymujemy	lub	kradniemy]	informacje?
whence	and	what.ACC	receive	or	steal	information.ACC

‘What information and where from do we receive or steal?’

(5.198)

[kogo	i	kiedy]	Kamiński	[podsłuchiwał	lub	chciał	podsłuchiwać]?
who.ACC	and	when	Kamiński	bugged	or	wanted	bug.INF

‘Who and when did Kamiński bug or want to bug?’ (NKJP)

When, as in (5.197)–(5.198), verbs are coordinated, the conjunction is represented at the same level as the set containing particular verbal heads. The structure provided in (5.199) represents the following fragment of (5.197): *[otrzymujemy lub kradniemy] informacje.*

(5.199)

$$\begin{bmatrix}
\left\{\begin{bmatrix}\text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\end{bmatrix}\\ \text{OBJ} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`INFORMATION'}\\ \text{CASE} & \text{ACC}\end{bmatrix}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`STEAL}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ} & \boxed{2}\end{bmatrix}\right\}\\
\text{COORD-FORM} \quad \text{LUB}
\end{bmatrix}$$

If *i* ‘and’ was used for coordinating verbs in (5.197) (*otrzymujemy i kradniemy*), this example could[38] be unproblematic: the conjunction used for verb coordination would be the same as in the lexico-semantic coordinate phrase (see (5.193) and (5.194) for comparison) – the values of COORD-FORM introduced by both coordinate phrases would unify.

By contrast, when, as in (5.197), *lub* ‘or’ is used as the conjunction in the coordinate verb phrase, an inconsistent f-structure (see § 2.2.2 for discussion of consistency) is produced due to the clash of values of COORD-FORM in the top-level f-structure in (5.200):[39]

(5.200)
$$* \begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\end{bmatrix} \\ \text{OBJ} & \boxed{2}\begin{bmatrix} \text{PRED} & \text{`INFORMATION'} \\ \text{CASE} & \text{ACC} \\ \text{ADJ} & \left\{\begin{bmatrix}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\}\end{bmatrix} \\ \text{ADJ} & \left\{\boxed{3}\begin{bmatrix}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`STEAL}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ} & \boxed{2}\\ \text{ADJ} & \left\{\boxed{3}\right\}\end{bmatrix}\right\} \\ \text{COORD-FORM} \quad \text{I}\neq\text{LUB} \end{bmatrix}$$

In (5.200) conjunctions used in (5.197) in lexico-semantic (*i*) and verbal (*lub*) coordinate phrases set conflicting values of COORD-FORM attribute, I and LUB, respectively – they are therefore represented as an inequality in (5.200), I≠LUB, indicating inconsistency.

The same problem arises in (5.198), whose f-structure, inconsistent under the current analysis, is provided in (5.201): LUB, the value of COORD-FORM attribute introduced by the coordinate verbal phrase (*[podsłuchiwał lub chciał podsłuchiwać]*), conflicts with I introduced by the lexico-semantic coordination (*[kogo i kiedy]*).

(5.201)
$$* \begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{`BUG}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`KAMIŃSKI'}\end{bmatrix} \\ \text{OBJ} & \boxed{2}\begin{bmatrix} \text{PRED} & \text{`WHO'} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT}\end{bmatrix} \\ \text{ADJ} & \left\{\boxed{3}\begin{bmatrix}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\}\end{bmatrix}, \begin{bmatrix}\text{PRED} & \text{`WANT}\langle\boxed{1},\boxed{4}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{XCOMP} & \boxed{4}\begin{bmatrix}\text{PRED} & \text{`BUG}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ} & \boxed{2}\\ \text{ADJ} & \left\{\boxed{3}\right\}\end{bmatrix}\end{bmatrix}\right\} \\ \text{COORD-FORM} \quad \text{I}\neq\text{LUB} \end{bmatrix}$$

[38] Some implemented LFG grammars treat the attribute holding the conjunction form as an instantiated feature. The value of such features may be set only once, so problems are expected even when the conjunction in lexico-semantic coordination is the same as in the conjoined verbal phrase. Thanks are due to Tracy Holloway King for pointing this out.

[39] Note that in (5.200) the index $\boxed{3}$ is assigned to *Skąd*, which is a shared element of adjunct sets of respective predicates, rather than to the entire set – this reflects the fact that the adjunct sets themselves are not structure-shared: it is possible for each predicate to have additional adjuncts, independently of each other.

5.6.4 Embedded lexico-semantic coordination

A related problem is caused by the embedding of coordination; consider the examples below:

(5.202) Nigdy nie wiadomo, [[kto lub co], skąd i kiedy] zaatakuje.
never NEG know who.NOM or what.NOM whence and when attacks
'You never know who or what, where from and when may attack.' (NKJP)

(5.203) kombinowaniem [[kto, kogo, kiedy i jak], [z kim przeciw komu] albo
plotting who.NOM who.ACC when and how with whom against whom or
[od kogo i za co]]
from whom and for what
'[...] plotting about who, whom, when and how, with whom against whom or from whom and for what [...]' (NKJP)

Two varieties of coordination are involved in (5.202): the first conjunct of lexico-semantic coordination (*kto lub co*) is at the same time a regular coordinate NP (both conjuncts are marked for the nominative case and together they correspond to the subject grammatical function), while the remaining lexico-semantic conjuncts (*skąd i kiedy*) are adjuncts (ablative and temporal). This is unproblematic representationally, because the first conjunction is represented inside the coordinate NP (see (5.204)) while the other, used for lexico-semantic coordination, is represented in the outer f-structure layer, as shown in (5.205).

(5.204)
$$\left[\begin{array}{ll} \multicolumn{2}{l}{\left\{ \left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}}\\ \text{COORD-FORM} & \text{LUB}\end{array}\right]$$

(5.205)
$$\left[\begin{array}{ll} \text{SUBJ} & \left[\begin{array}{ll} \multicolumn{2}{l}{\left\{ \left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}}\\ \text{COORD-FORM} & \text{LUB}\end{array}\right]\\ \text{ADJ} & \left\{ \left[\begin{array}{ll}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\\ \text{COORD-FORM} & \text{I}\end{array}\right]$$

Example (5.203) is more interesting as it presents embedded lexico-semantic coordination: two edge conjuncts are also instances of such coordination. The first conjunct (*[kto, kogo, kiedy i jak]*) contains a subject, an object and two adjuncts (temporal and manner), the middle conjunct (*[z kim przeciw komu]*) contains two *wh*-phrases (obliques) which perhaps are not coordinated, and the last conjunct (*[od kogo i za co]*) consists of another oblique coordinated with an adjunct.

It is possible to construct a less complicated example, though:

(5.206) [[Kto i kiedy] lub [kogo i gdzie]] spotkał?
who.NOM and when or who.ACC and where met
'Where did who meet whom and when?'

In (5.206) the first conjunct (*[Kto i kiedy]*) contains a subject and an adjunct (temporal) while the other (*[kogo i gdzie]*) consists of a direct object and another adjunct (expressing location).

Under the current flat representation, examples such as (5.203) and (5.206) cause at least two problems. First, there is no information about which elements were coordinated; this problem was described in § 5.6.1. The other issue is the representation of conjunctions used in embedded coordination – under flat representation they would be placed in the same fragment of f-structure, leading to a clash of values (I set by *i* vs LUB set by *lub*) similar to the one discussed in § 5.6.3.

5.6.5 Proposed solution

On the analysis presented so far, all conjuncts of lexico-semantic coordination are only represented in the f-structure of their respective head, namely as values of corresponding grammatical functions. There is no feature structure representing the coordinated structure itself, the *Skąd i jakie* of (5.197) or the *Kto i kiedy lub kogo i gdzie* of (5.206).

An alternative is that conjuncts taking part in lexico-semantic coordination, apart from being values of respective grammatical functions, are gathered in the value of a discourse function. Since lexico-semantic coordination can use the mechanism of extraction, it is assumed without discussion that the relevant discourse function is UDF – unbounded dependency function.[40]

More specifically, each coordinate structure is – as in standard LFG – represented as a hybrid feature structure containing an explicit representation of the conjunction and a set gathering relevant conjuncts. Multi-level coordination, as in (5.206), is represented using embedded hybrid structures, each carrying information about the particular conjunction used at a given level. The resulting coordinate structure is then assigned a discourse function (UDF) and its relevant parts (f-structures of particular conjuncts) are structure-shared with values of appropriate grammatical functions. The following f-structure results as the representation of (5.206):

(5.207)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`MEET}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{SUBJ} & \boxed{1} \\
\text{OBJ} & \boxed{2} \\
\text{ADJ} & \left\{\boxed{3},\boxed{4}\right\} \\
\text{UDF} & \left\{\left[\begin{array}{l}
\left\{\left[\begin{array}{l}
\left\{\begin{array}{l}
\boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right], \\
\boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{array}\right]
\end{array}\right\} \\
\text{COORD-FORM} \quad \text{I}
\end{array}\right],
\left[\begin{array}{l}
\left\{\begin{array}{l}
\boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right], \\
\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`WHERE'}\\ \text{TYPE} & \text{INT}\end{array}\right]
\end{array}\right\} \\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]\right\} \\
\text{COORD-FORM} \quad \text{LUB}
\end{array}\right]\right\}
\end{array}\right]$$

§ 5.7 shows how this modified account of lexico-semantic coordination may be formalised in LFG.

[40] See § 2.4 for discussion of UDF. The analysis does not, however, hinge on the choice of this particular discourse function – it could be changed without affecting the core analysis.

5.7 Improved analysis

First of all, the rule which assigns grammatical functions to particular conjuncts under lexico-semantic coordination, (5.124) repeated below as (5.208), is modified by replacing ↑ at the beginning of the path with an inside-out path, ($\textsc{udf} \in^{*} \uparrow$). See (5.209) for its modified version:

(5.208) $$\text{XPextr}_{type} \quad \rightarrow \quad \begin{array}{c}\text{XP}_{type}\\ (\uparrow \textsc{xpath gf}^{+}){=}\downarrow\end{array}$$

(5.209) $$\text{XPextr}_{type} \quad \rightarrow \quad \begin{array}{c}\text{XP}_{type}\\ \uparrow{=}\downarrow\\ ((\textsc{udf} \in^{*} \uparrow)\ \textsc{xpath gf}^{+}){=}\downarrow\end{array}$$

This change makes it possible for f-structures of particular conjuncts within UDF to be structure-shared with relevant parts of the top-level f-structure (and potentially further embedded f-structures: $\textsc{xpath gf}^{+}$). In this way various grammatical (rather than discourse) functions are assigned properly.

Moreover, it is worth noting that another change was introduced in (5.209) with respect to (5.208): the head annotation (↑=↓) was added to XP_{type}. In this way the lexico-semantic conjunct passes its f-structure to the mother category, which later becomes an element of the UDF set, apart from being structure shared with the relevant part of the f-structure (outside the UDF set) using the constraint discussed above, $((\textsc{udf} \in^{*} \uparrow)\ \textsc{xpath gf}^{+}){=}\downarrow$.

Secondly, the lexico-semantic coordinate phrase needs to be assigned a discourse function so that (5.209), the rule handling grammatical function assignment to lexico-semantic conjuncts, based on an inside-out path, ($\textsc{udf} \in^{*} \uparrow$), can work properly:

(5.210) $$\text{lexsemP} \quad \rightarrow \quad \begin{array}{c}\text{XPlxm}_{type}\\ \downarrow{\in}(\uparrow \textsc{udf})\end{array}$$

The rule provided in (5.210) is preliminary – it will be replaced later by (5.229).

5.7.1 Monoclausal structures

The new rule handling monoclausal lexico-semantic coordination, a modified version of (5.138) repeated below in (5.211), is provided in (5.212). Let us discuss the introduced modifications.

(5.211) $$\text{XPlxm}_{type} \quad \rightarrow \quad \begin{array}{c}\text{XPextr}_{type}\\ \uparrow{=}\downarrow\end{array} \quad [, \begin{array}{c}\text{XPextr}_{type}]^{*}\\ \uparrow{=}\downarrow\end{array} \quad \begin{array}{c}\text{CONJ}\\ \uparrow{=}\downarrow\end{array} \quad \begin{array}{c}\text{XPextr}_{type}\\ \uparrow{=}\downarrow\end{array}$$

(5.212) $$\text{XPlxm}_{type} \quad \rightarrow \quad \begin{array}{c}\text{XPlxmC}_{type}\\ \downarrow{\in}\uparrow\end{array} \quad [, \begin{array}{c}\text{XPlxmC}_{type}]^{*}\\ \downarrow{\in}\uparrow\end{array} \quad \begin{array}{c}\text{CONJ}\\ \uparrow{=}\downarrow\end{array} \quad \begin{array}{c}\text{XPlxmC}_{type}\\ \downarrow{\in}\uparrow\end{array}$$

(5.213) $$\text{XPlxmC}_{type} \rightarrow \{\ \text{XPextr}_{type} \mid \text{XPlxm}_{type}\ \}$$

First, conjuncts are no longer treated as co-heads (↑=↓) – set membership annotation (↓∈↑) on conjuncts in (5.212) makes it possible to avoid issues caused by the flat representation obtained using the co-head annotation in (5.211) – see § 5.6.3 for discussion.

Secondly, particular conjuncts are now XPlxmC$_{type}$ (see the rule in (5.213)) – it rewrites to a disjunction of XPextr$_{type}$ and XPlxm$_{type}$, which makes it possible to account for further embedding of lexico-semantic coordination – see the discussion in § 5.6.4.

Let us now examine some f-structures produced by the modified rules, starting with the counterpart of an f-structure presented earlier for a sentence involving simple lexico-semantic coordination – the f-structure provided in (5.215) corresponds to (5.5) repeated as (5.214) below:

(5.214) Skąd i jakie otrzymujemy informacje?
whence and what.ACC receive information.ACC
'What information and from where do we receive?' (NKJP)

(5.215)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right] \\
\text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`INFORMATION'}\\ \text{CASE} & \text{ACC}\\ \text{ADJ} & \left\{\boxed{3}\right\}\end{array}\right]\\
\text{ADJ} & \left\{\boxed{4}\right\}\\
\text{UDF} & \left\{\left[\begin{array}{l}\left\{\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{array}\right], \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\\ \text{COORD-FORM} \quad \text{I}\end{array}\right]\right\}
\end{array}\right]$$

Let us first consider how the f-structure shown in (5.215) is produced by the modified rules. The partial f-structure in (5.216) corresponds to *Skąd i jakie* in (5.214) – the f-structures corresponding to relevant conjuncts are added to a set using (5.212), where conjuncts correspond to the first disjunct of (5.213): XPextr$_{type}$. The f-structure fragment in (5.217) is produced after the coordinate structure corresponding to *Skąd i jakie* represented in (5.216) is added to the UDF set as a result of the rule in (5.210) – once this constraint is introduced, the structure-sharing statements defined in (5.209) are effective. Finally, the partial f-structure corresponding to *otrzymujemy informacje* shown in (5.218) is unified with (5.217) – the result is the full f-structure in (5.215), which corresponds to (5.214).

(5.216)
$$\left[\begin{array}{l}\left\{\left[\begin{array}{ll}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\\ \text{COORD-FORM} \quad \text{I}\end{array}\right]$$

(5.217)
$$\left[\begin{array}{ll}
\textsc{obj} & \left[\begin{array}{ll}\textsc{adj} & \left\{\boxed{3}\right\}\end{array}\right] \\
\textsc{adj} & \left\{\boxed{4}\right\} \\
\textsc{udf} & \left\{\left[\begin{array}{l}
\left\{\boxed{4}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{whence}\text{'} \\ \textsc{type} & \textsc{int}\end{array}\right], \boxed{3}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{what}\text{'} \\ \textsc{case} & \textsc{acc} \\ \textsc{type} & \textsc{int}\end{array}\right]\right\} \\
\textsc{coord-form} \quad \textsc{i}
\end{array}\right]\right\}
\end{array}\right]$$

(5.218)
$$\left[\begin{array}{ll}
\textsc{pred} & \text{`}\textsc{receive}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{pro}\text{'}\end{array}\right] \\
\textsc{obj} & \boxed{2}\left[\begin{array}{ll}\textsc{pred} & \text{`}\textsc{information}\text{'} \\ \textsc{case} & \textsc{acc}\end{array}\right]
\end{array}\right]$$

Let us now compare (5.194), the earlier representation of (5.214), with (5.215). While both contain a representation of the conjunction, these f-structures differ considerably: in (5.194) the conjunction found in lexico-semantic coordination is represented in the same place where the conjunction used in verbal coordination would be expected to appear (see § 5.6.3), while lexico-semantic conjuncts are simply values of relevant grammatical function attributes. By contrast, in (5.215) the entire lexico-semantic coordinate phrase is placed inside the UDF attribute. Note that the lexico-semantic phrase is treated in the same way as plain instances of coordination: conjuncts are placed in a set, while the conjunction can be found in the COORD-FORM attribute next to the set containing relevant conjuncts. The f-structures of particular lexico-semantic conjuncts are structure-shared (this is indicated using variables) with the values of relevant grammatical function attributes: the first conjunct, *Skąd* 'whence', is an adjunct depending on the verb, *otrzymujemy*, while the second conjunct, *jakie* 'what (like)', is an adjunct which belongs to the direct object, *informacje*.

Let us now proceed to f-structures produced by the modified rules for structures which were problematic under the previous analysis, namely to structures which involve verbal coordination and embedding – for discussion of these issues, see § 5.6.3 and § 5.6.4, respectively.

5.7.1.1 With verbal coordination

Let us start with (5.220) corresponding to (5.197), repeated below as (5.219).

(5.219)	[Skąd	i	jakie]	[otrzymujemy	lub	kradniemy]	informacje?
	whence	and	what.ACC	receive	or	steal	information.ACC

'What information and where from do we receive or steal?'

(5.220)

$$\left[\begin{array}{l}\left\{\begin{array}{l}\left[\begin{array}{ll}\text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right]\\ \text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`INFORMATION'}\\ \text{CASE} & \text{ACC}\\ \text{ADJ} & \left\{\boxed{3}\right\}\end{array}\right]\\ \text{ADJ} & \left\{\boxed{4}\right\}\\ \text{UDF} & \boxed{5}\left\{\left[\begin{array}{l}\left\{\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`WHENCE'}\\ \text{TYPE} & \text{INT}\end{array}\right], \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`WHAT'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\\ \text{COORD-FORM} \quad \text{I}\end{array}\right]\right\}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`STEAL}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ} & \boxed{2}\\ \text{ADJ} & \left\{\boxed{4}\right\}\\ \text{UDF} & \boxed{5}\end{array}\right]\end{array}\right\}\\ \text{COORD-FORM} \quad \text{LUB}\end{array}\right]$$

The f-structure in (5.220) is obtained in a way similar to (5.215) described above: the f-structure fragments corresponding to *Skąd i jakie* are the same (see (5.216)–(5.217)), only the partial f-structure corresponding to *otrzymujemy lub kradniemy informacje* is different: instead of (5.218), (5.221) is unified with (5.217), yielding (5.220).

(5.221)

$$\left[\begin{array}{l}\left\{\left[\begin{array}{ll}\text{PRED} & \text{`RECEIVE}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\end{array}\right]\\ \text{OBJ} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`INFORMATION'}\\ \text{CASE} & \text{ACC}\end{array}\right]\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{`STEAL}\langle\boxed{1},\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{1}\\ \text{OBJ} & \boxed{2}\end{array}\right]\right\}\\ \text{COORD-FORM} \quad \text{LUB}\end{array}\right]$$

Let us now proceed to the discussion of the f-structure representation in (5.220): the conjunction coordinating verbal phrases (*lub* 'or' in (5.219)) is represented in the top-level f-structure, while the conjunction used in lexico-semantic coordination (*i* 'and') is represented inside the hybrid structure which is the value of UDF. The f-structures of relevant lexico-semantic conjuncts stored inside UDF are structure-shared with values of appropriate grammatical functions, as explained in the discussion of (5.215). It is crucial to note that the lexico-semantic phrase is a dependent of both coordinated verbs in (5.219) at the same time: as indicated by structure-sharing in (5.220), the UDF attribute containing the lexico-semantic coordinate phrase is shared by both verbs and the lexico-semantic conjuncts it contains are in turn shared with corresponding attributes inside the f-structure of the second verb (KRAŚĆ 'steal').

5.7.1.2 With embedding

Another feature structure, (5.223), represents sentence (5.206), repeated below as (5.222):

(5.222) [[Kto i kiedy] lub [kogo i gdzie]] spotkał?
who.NOM and when or who.ACC and where met
'Where did who meet whom and when?'

(5.223)
$$\begin{bmatrix} \text{PRED} & \text{‘MEET}\langle\boxed{1},\boxed{2}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{OBJ} & \boxed{2} \\ \text{ADJ} & \left\{\boxed{3},\boxed{4}\right\} \\ \text{UDF} & \left\{ \begin{bmatrix} \left\{ \begin{bmatrix} \left\{ \begin{matrix} \boxed{1} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{NOM} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \boxed{3} \begin{bmatrix} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix}, \begin{bmatrix} \left\{ \begin{matrix} \boxed{2} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \boxed{4} \begin{bmatrix} \text{PRED} & \text{‘WHERE’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \end{bmatrix} \right\} \end{bmatrix}$$

In order to build the f-structure in (5.223), both disjuncts of (5.213) defining the conjuncts in (5.212) must be used: its first disjunct is used for plain lexico-semantic coordination such as *Kto i kiedy* and *kogo i gdzie*, where particular conjuncts correspond to single, non-coordinate phrases – their partial f-structures are shown in (5.224) and (5.225), respectively.

(5.224)
$$\begin{bmatrix} \left\{ \begin{matrix} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{NOM} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \begin{bmatrix} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix}$$

(5.225)
$$\begin{bmatrix} \left\{ \begin{matrix} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \begin{bmatrix} \text{PRED} & \text{‘WHERE’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix}$$

Together with the rule in (5.212), the second disjunct of (5.213) is used for the embedding of lexico-semantic coordination such as in *Kto i kiedy lub kogo i gdzie*, where particular conjuncts are coordinate phrases constructed using its first disjunct.

(5.226)
$$\begin{bmatrix} \left\{ \begin{bmatrix} \left\{ \begin{matrix} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{NOM} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \begin{bmatrix} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix}, \begin{bmatrix} \left\{ \begin{matrix} \begin{bmatrix} \text{PRED} & \text{‘WHO’} \\ \text{CASE} & \text{ACC} \\ \text{TYPE} & \text{INT} \end{bmatrix}, \\ \begin{bmatrix} \text{PRED} & \text{‘WHERE’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \end{bmatrix}$$

The next step is to add the partial f-structure of the embedded lexico-semantic phrase in (5.226) to the UDF set (according to the rule in (5.210)), so that the structure-sharing statements defined in (5.209) have effect – the result is the f-structure fragment in (5.227). Finally, (5.227) is unified with the f-structure of the remaining part of the sentence, *spotkał*, shown in (5.228) – the full f-structure in (5.223) results.

(5.227)
$$\left[\begin{array}{ll}
\text{SUBJ} & \boxed{1}\\
\text{OBJ} & \boxed{2}\\
\text{ADJ} & \left\{\boxed{3},\boxed{4}\right\}\\
\text{UDF} & \left\{\left[\begin{array}{ll}
\multicolumn{2}{l}{\left\{\left[\begin{array}{ll}
\multicolumn{2}{l}{\left\{\begin{array}{l}
\boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right],\\
\boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{array}\right]
\end{array}\right\}}\\
\text{COORD-FORM} & \text{I}
\end{array}\right],
\left[\begin{array}{ll}
\multicolumn{2}{l}{\left\{\begin{array}{l}
\boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INT}\end{array}\right],\\
\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`WHERE'}\\ \text{TYPE} & \text{INT}\end{array}\right]
\end{array}\right\}}\\
\text{COORD-FORM} & \text{I}
\end{array}\right]\right\}}\\
\text{COORD-FORM} & \text{LUB}
\end{array}\right]\right\}
\end{array}\right]$$

(5.228)
$$\left[\begin{array}{ll}
\text{PRED} & \text{`MEET}\langle\boxed{1},\boxed{2}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\\
\text{OBJ} & \boxed{2}
\end{array}\right]$$

Let us now briefly discuss the representation obtained in (5.223): embedded lexico-semantic coordination is represented inside the UDF attribute – the top-level hybrid structure contains a conjunction (*lub* 'or') and a set containing two elements, each of which is a lexico-semantic coordinate structure containing a conjunction (*i* 'and') and a set containing relevant conjuncts (an argument and an adjunct). Again, the f-structures of all lexico-semantic conjuncts are structure-shared with appropriate attributes of the relevant predicate (subject, object and two modifiers of SPOTKAĆ 'meet').

5.7.2 Multiclausal structures

Changes introduced in the representation of monoclausal lexico-semantic coordination make it possible to change the representation of structures which require multiclausal representation – the two possible modes of representation are presented below in separate sections devoted to biclausal (always two clauses; § 5.7.2.2) as opposed to multiclausal (possibly more than two clauses; § 5.7.2.1) representation, starting with the latter as it requires fewer changes.

Before proceeding to the discussion of particular representations, let us introduce a shared top-level rule for lexico-semantic coordinate phrases:

(5.229)
$$\text{anyLEXSEM} \;\rightarrow\; \left\{\; \begin{array}{c}\text{XPlxm}_{type}\\ \downarrow\in(\uparrow\ \text{UDF})\end{array} \;\middle|\; \text{XPlxb}_{int} \;\right\}$$

In the first disjunct monoclausal lexico-semantic coordinate phrases are assigned the UDF discourse function – for this reason the rule defined in (5.210) is no longer needed. By contrast, no discourse function is assigned to multiclausal lexico-semantic phrases used in the second disjunct – this is done in the relevant rules for XPextrbicl$_{type}$ provided in the following subsections.

5.7.2.1 Multiclausal representation

Under the multiclausal representation, which can involve more than two clauses (as opposed to the biclausal analysis presented in § 5.7.2.2), there is no need to modify (5.176), the rule for handling such coordination, which is repeated in (5.230) for convenience. The difference is how the category to which XPextrbicl[41] rewrites is annotated: the co-head annotation (↑=↓) used in (5.231) is replaced in (5.232) by set membership annotation which adds its f-structure to the set representing the UDF discourse function (↓∈(↑ UDF)).

(5.230)
$$\begin{array}{llllll} \text{XPlxb}_{int} & \rightarrow & \text{PARTbicl}_{int} & [,\ \text{XPextrbicl}_{int}]^{*} & \text{CONJ} & \text{XPextrbicl}_{int} \\ & & \downarrow\in\uparrow & \downarrow\in\uparrow & \uparrow=\downarrow & \downarrow\in\uparrow \end{array}$$

(5.231)
$$\begin{array}{lll} \text{XPextrbicl}_{type} & \rightarrow & \text{XPextr}_{type} \\ & & \uparrow=\downarrow \\ & & \text{PRODROP} \end{array}$$

(5.232)
$$\begin{array}{lll} \text{XPextrbicl}_{type} & \rightarrow & \text{XPextr}_{type} \\ & & \downarrow\in(\uparrow\ \text{UDF}) \\ & & \text{PRODROP} \end{array}$$

To see the effect of these changes, let us consider (5.234), the f-structure corresponding to the embedded question (*czy i kiedy wróci*) in (5.233), repeated from (5.160):

(5.233) Nie wiadomo było, czy *(i) kiedy wróci.
NEG know was PART and when returns
'It was not clear whether and when he would return.' (NKJP)

(5.234)
$$\left[\begin{array}{l} \left\{ \left[\begin{array}{ll} \text{PRED} & \text{`RETURN}\langle\boxed{1}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\text{PRED}\quad\text{`PRO'}\right] \\ \text{CLAUSE-TYPE} & \text{INT} \end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{`RETURN}\langle\boxed{1}\rangle\text{'} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \left\{\boxed{2}\right\} \\ \text{UDF} & \left\{\boxed{2}\left[\begin{array}{ll} \text{PRED} & \text{`WHEN'} \\ \text{TYPE} & \text{INT} \end{array}\right]\right\} \end{array}\right] \right\} \\ \text{COORD-FORM}\quad \text{I} \end{array}\right]$$

(5.234) is constructed as a result of unifying the partial f-structures corresponding to *czy i kiedy* ((5.235) represents the effect of applying the rule in (5.230), which includes structure-sharing relevant elements of the UDF set with relevant grammatical functions) and *wróci* (see (5.164), repeated below as (5.236)). Since (5.235) contains a set, (5.236) distributes over it, yielding the structure in (5.234), which consists of two clauses.

(5.235)
$$\left[\begin{array}{l} \left\{ \left[\text{CLAUSE-TYPE}\quad\text{INT}\right], \left[\begin{array}{ll} \text{ADJ} & \left\{\boxed{3}\right\} \\ \text{UDF} & \left\{\boxed{3}\left[\begin{array}{ll} \text{PRED} & \text{`WHEN'} \\ \text{TYPE} & \text{INT} \end{array}\right]\right\} \end{array}\right] \right\} \\ \text{COORD-FORM}\quad \text{I} \end{array}\right]$$

(5.236)
$$\left[\begin{array}{ll} \text{PRED} & \text{`RETURN}\langle\boxed{1}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\left[\text{PRED}\quad\text{`PRO'}\right] \end{array}\right]$$

[41] *type* is *int* under multiclausal lexico-semantic coordination – only interrogative conjuncts may be used in this construction, as in (5.230).

The difference with respect to the previous representation of (5.233), given in (5.166), is the presence of the UDF attribute in the second clause in (5.234) – its only element is structure-shared with the relevant grammatical function (adjunct) of the relevant predicate, which makes it possible to identify it as one of the lexico-semantic conjuncts. Apart from this, the basic representation is unchanged – both involve two clauses: one containing the *yes/no* question particle CZY and another one which contains the *wh*-word conjunct. Though the representation of the conjunction was omitted in (5.166), it would be found in the same place as in (5.234).

For the sake of comparison with the representation presented in § 5.7.2.2, let us consider (5.239), the f-structure representation of (5.238), the embedded question from (5.237), which involves three conjuncts (*Czy*, *kiedy* and *kto*):

(5.237) Czy, kiedy i kto zajmie się drogami w Głębowicach na razie nie wiadomo.
PART when and who.NOM take care roads in Głębowice so far NEG known
‘It is not known yet, whether, who and when will take care of roads in Głębowice.’
(NKJP)

(5.238) Czy, kiedy i kto zajmie się drogami?
PART when and who.NOM take care roads

(5.239)

$$\left[\begin{array}{l}\left\{\begin{array}{l}\left[\begin{array}{ll}\text{PRED} & \text{‘TAKE\_CARE}\langle\boxed{1},\boxed{2}\rangle\text{’}\\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\\ \text{CASE} & \text{NOM}\end{array}\right]\\ \text{OBL} & \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{‘ROADS’}\\ \text{CASE} & \text{INST}\end{array}\right]\\ \text{CLAUSE-TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{‘TAKE\_CARE}\langle\boxed{3},\boxed{2}\rangle\text{’}\\ \text{SUBJ} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\\ \text{CASE} & \text{NOM}\end{array}\right]\\ \text{OBL} & \boxed{2}\\ \text{ADJ} & \left\{\boxed{4}\right\}\\ \text{UDF} & \left\{\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{‘WHEN’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right], \left[\begin{array}{ll}\text{PRED} & \text{‘TAKE\_CARE}\langle\boxed{5},\boxed{2}\rangle\text{’}\\ \text{SUBJ} & \boxed{5}\\ \text{OBL} & \boxed{2}\\ \text{UDF} & \left\{\boxed{5}\left[\begin{array}{ll}\text{PRED} & \text{‘WHO’}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]\end{array}\right\}\\ \text{COORD-FORM} \quad \text{I}\end{array}\right]$$

Let us see how (5.239) is constructed. Following the rule in (5.230), each conjunct is added to a set as a separate element – this effect is shown in (5.240), where the set contains 3 elements corresponding to particular conjucts. Moreover, the calls to PRODROP template placed in (5.232) result in adding implicit arguments in the f-structures of relevant conjuncts: an implicit subject is added in conjuncts which do not contain a lexical one (the last conjunct contains *kto*).

(5.240)

$$\left[\begin{array}{l}\left\{\begin{array}{l}\left[\begin{array}{ll}\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\\ \text{CASE} & \text{NOM}\end{array}\right]\\ \text{CLAUSE-TYPE} & \text{INT}\end{array}\right], \left[\begin{array}{ll}\text{SUBJ} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\\ \text{CASE} & \text{NOM}\end{array}\right]\\ \text{ADJ} & \left\{\boxed{4}\right\}\\ \text{UDF} & \left\{\boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{‘WHEN’}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right], \left[\begin{array}{ll}\text{SUBJ} & \boxed{5}\\ \text{UDF} & \left\{\boxed{5}\left[\begin{array}{ll}\text{PRED} & \text{‘WHO’}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{array}\right]\right\}\end{array}\right]\end{array}\right\}\\ \text{COORD-FORM} \quad \text{I}\end{array}\right]$$

(5.241)
$$\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{take\_care}\langle\boxed{1},\boxed{2}\rangle\text{’}\\ \textsc{subj} & \boxed{1}\\ \textsc{obl} & \boxed{2}\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{roads}\text{’}\\ \textsc{case} & \textsc{inst}\end{array}\right]\end{array}\right]$$

When the f-structure in (5.240) is unified with (5.241), the f-structure corresponding to the rest of (5.238) (*zajmie się drogami*), (5.241) distributes over the set in (5.240) and the f-structure in (5.239) is produced. The resulting f-structure consists of three coordinated clauses: the first one contains *czy*, the second one features *kiedy* and the last one hosts *kto*. While the OBL argument corresponding to *drogami* is shared by all three clauses, the remaining argument, the subject, must be implicit in clauses other than the last one, which contains the lexical subject (see § 5.5.3 for discussion of the mechanism handling argument saturation under the multiclausal analysis).

5.7.2.2 Biclausal representation

There is an alternative to the analysis presented in § 5.7.2.1 – the proposed representation always involves two clauses: one containing the *yes/no* question particle (*czy*) and the other which contains the remaining lexico-semantic conjuncts. It is motivated by economy of representation (fewer implicit arguments) and the intuition that the remaining conjuncts should be placed in one clause (rather than in distinct clauses, as it is the case in § 5.7.2.1).

To obtain such a representation, the rule for handling multiclausal lexico-semantic coordination presented in (5.230) is modified as in (5.242). This modification is accompanied by the change of the rewriting rule for XPextrbicl$_{type}$. While in (5.232), XPextrbicl$_{type}$ rewrites to XPextr$_{type}$ exclusively, in (5.243) it rewrites to a disjunction of two categories: XPextr$_{type}$ and XPlxm$_{type}$ (see (5.212) for the definition of the latter).

(5.242)
$$\begin{array}{lllll}\text{XPlxb}_{int} & \rightarrow & \text{PARTbicl}_{int} & \text{CONJ} & \text{XPextrbicl}_{int}\\ & & \downarrow\in\uparrow & \uparrow=\downarrow & \downarrow\in\uparrow\end{array}$$

(5.243)
$$\begin{array}{llllllll}\text{XPextrbicl}_{type} & \rightarrow & \{ & \text{XPextr}_{type} & | & \text{XPlxm}_{type} & \}\\ & & & \downarrow\in(\uparrow\ \textsc{udf}) & & \downarrow\in(\uparrow\ \textsc{udf}) & \\ & & & \text{PRODROP} & & \text{PRODROP} & \end{array}$$

While the first disjunct of (5.243) handles simple, non-coordinate lexico-semantic conjuncts, the second one makes it possible to coordinate monoclausal lexico-semantic coordinate structures (possibly embedded)[42] with the *yes/no* question particle under multiclausal coordination.

Though this change has no effect on representation of multiclausal lexico-semantic coordination featuring two conjuncts (it is the same as in § 5.7.2.1 – the f-structure in (5.234) corresponds to (5.233)), there is a stark difference when there are at least three conjuncts, as in (5.237) and its simplified version in (5.238).

By contrast with the representation proposed in § 5.7.2.1, which produces the f-structure in (5.239) with three clauses, the modified rule given in (5.242) produces[43] (5.244) with only two

[42] The rewriting rule for XPlxm$_{type}$ provided in (5.212) allows for embedded lexico-semantic coordination (see § 5.6.4 and § 5.7.1 for discussion and examples).

[43] More precisely, the rule (5.242) would only produce the f-structure in (5.244) only if the comma in (5.237) is treated as an instance of CONJ category. See the discussion of (5.247).

clauses: the first one contains CZY, while the other hosts the two remaining conjuncts (*kiedy* and *kto*). It is worth noting that the second clause in (5.244) is a monoclausal lexico-semantic structure described in § 5.7.1, produced by the rule defined in (5.212) (more specifically, using the first disjunct of (5.213)) to which the second disjunct of (5.243) rewrites.

(5.244)

$$\left\{ \begin{bmatrix} \text{PRED} & \text{`TAKE\_CARE}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{NOM}\end{bmatrix} \\ \text{OBL} & \boxed{2}\begin{bmatrix}\text{PRED} & \text{`ROADS'}\\ \text{CASE} & \text{INST}\end{bmatrix} \\ \text{CLAUSE-TYPE} & \text{INT}\end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{`TAKE\_CARE}\langle\boxed{3},\boxed{2}\rangle\text{'} \\ \text{SUBJ} & \boxed{3} \\ \text{OBL} & \boxed{2} \\ \text{ADJ} & \left\{\boxed{4}\right\} \\ \text{UDF} & \left\{ \begin{bmatrix} \left\{ \boxed{4}\begin{bmatrix}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{bmatrix}, \boxed{3}\begin{bmatrix}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\} \\ \text{COORD-FORM} \quad \text{I}\end{bmatrix}\right\}\end{bmatrix}\right\}$$

The f-structure in (5.244) is obtained for (5.238) as a result of unifying (5.245), the partial f-structure corresponding to lexico-semantic coordination, with (5.241), the f-structure corresponding to the rest of (5.238).

(5.245)

$$\left\{ \begin{bmatrix} \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{NOM}\end{bmatrix} \\ \text{CLAUSE-TYPE} & \text{INT}\end{bmatrix}, \begin{bmatrix} \text{SUBJ} & \boxed{3} \\ \text{ADJ} & \left\{\boxed{4}\right\} \\ \text{UDF} & \left\{ \begin{bmatrix} \left\{ \boxed{4}\begin{bmatrix}\text{PRED} & \text{`WHEN'}\\ \text{TYPE} & \text{INT}\end{bmatrix}, \boxed{3}\begin{bmatrix}\text{PRED} & \text{`WHO'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INT}\end{bmatrix}\right\} \\ \text{COORD-FORM} \quad \text{I}\end{bmatrix}\right\}\end{bmatrix}\right\}$$

Note that the set in (5.245), produced using (5.242), contains only two elements: one hosts *czy*, while the remaining lexico-semantic conjuncts are placed in the other set, inside the UDF attribute (using the rule in (5.243)).

Since there seems to be no (but see the discussion of (5.246) below) arguments in favour of putting conjuncts other than CZY in separate clauses (see § 5.4 for an extensive discussion of representation tests), the monoclausal analysis (see § 5.7.1) could be used for these conjuncts for reasons of simplicity and economy of representation, which seem to give an advantage over the representation proposed in § 5.7.2.1.

There exist, however, examples such as (5.246), where CZY is not an edge conjunct:

(5.246) kto, czy i kiedy zdobył jakiś szczyt?
who.NOM PART and when reach some summit
'Who, whether and when did reach some summit?' (Google)

Such examples pose a potential problem to the current analysis (though not to the one presented in § 5.7.2.1, provided that the rule in (5.230) is modified so as to allow a different word order), though it could be assumed that the second conjunct of multiclausal coordination (consisting of *kto* and *kiedy*) is discontinuous – it is split by *czy*.

Finally, though (5.244) is more economic than (5.239), there is a problem related to the representation of the conjunction: the conjunction used in lexico-semantic coordination joins clauses in (5.239), while in (5.244) it joins the two last lexico-semantic conjuncts – it is only represented in the UDF attribute, as in monoclausal lexico-semantic coordination.

This problem can be solved by changing the rule in (5.242) in the following way (C-F in (5.247) is an abbreviation for COORD-FORM):[44]

(5.247)
$$\mathrm{XPlxb}_{int} \rightarrow \left\{ \begin{array}{llll} & \mathrm{PARTbicl}_{int} & \mathrm{CONJ} & \mathrm{XPextrbicl}_{int} \\ & \downarrow\in\uparrow & \uparrow=\downarrow & \downarrow\in\uparrow \\ \mid & \mathrm{PARTbicl}_{int} & \mathrm{COMMA} & \mathrm{XPextrbicl}_{int} \\ & \downarrow\in\uparrow & (\uparrow \in \textsc{udf} \in \textsc{c-f})=(\uparrow \textsc{c-f}) & \downarrow\in\uparrow \end{array} \right\}$$

In this way, when a comma is used between CZY and the following lexico-semantic conjuncts, the conjunction stored in the UDF attribute in the second clause is structure-shared with the conjunction joining the two main clauses, as shown in (5.248):

(5.248)
$$\left[\begin{array}{l} \left\{ \left[\begin{array}{ll} \textsc{pred} & \text{`TAKE\_CARE}\langle\boxed{1},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{1}\left[\begin{array}{ll} \textsc{pred} & \text{`PRO'} \\ \textsc{case} & \textsc{nom} \end{array}\right] \\ \textsc{obl} & \boxed{2}\left[\begin{array}{ll} \textsc{pred} & \text{`ROADS'} \\ \textsc{case} & \textsc{inst} \end{array}\right] \\ \textsc{clause-type} & \textsc{int} \end{array}\right], \left[\begin{array}{ll} \textsc{pred} & \text{`TAKE\_CARE}\langle\boxed{3},\boxed{2}\rangle\text{'} \\ \textsc{subj} & \boxed{3} \\ \textsc{obl} & \boxed{2} \\ \textsc{adj} & \left\{\boxed{4}\right\} \\ \textsc{udf} & \left\{\left[\begin{array}{l} \left\{ \boxed{4}\left[\begin{array}{ll} \textsc{pred} & \text{`WHEN'} \\ \textsc{type} & \textsc{int} \end{array}\right], \boxed{3}\left[\begin{array}{ll} \textsc{pred} & \text{`WHO'} \\ \textsc{case} & \textsc{nom} \\ \textsc{type} & \textsc{int} \end{array}\right] \right\} \\ \textsc{coord-form} \quad \boxed{5}\ \textsc{i} \end{array}\right]\right\} \end{array}\right] \right\} \\ \textsc{coord-form} \quad \boxed{5} \end{array}\right]$$

5.7.3 Outstanding problematic issue

There is, however, another issue which remains unresolved, namely the combination of multiclausal lexico-semantic coordination (see § 5.7.2) and verbal coordination (see § 5.6.3). Consider the example provided below, a modified version of the embedded question from (5.233):

(5.249) [Czy *(i) kiedy] [wróci lub napisze]?
PART and when returns or writes
'Will he return or write and when will he return or write?'

As explained in § 5.4.2.16, lexico-semantic structures where the *yes/no* question particle CZY is one of the conjuncts are multiclausal. (5.250) is the overtly biclausal near counterpart of (5.249).

(5.250) [Czy [wróci lub napisze]] i [kiedy [wróci lub napisze]]?
PART returns or writes and when returns or writes
'Will he return or write and when will he return or write?'

[44] The change introduced in (5.247) does not affect sentences with only one lexico-semantic conjunct apart from CZY – in such situations the conjunction is represented in the top-level f-structure: see (5.234), which corresponds to (5.233).

(5.250) consists of two coordinated clauses, each of which contains two coordinated verbs: the partial f-structure corresponding to the first clause, *Czy wróci lub napisze*, is provided in (5.251), while the representation of *kiedy wróci lub napisze*, the fragment corresponding to the second clause, is given in (5.252).

(5.251)
$$\begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{‘PRO’}\end{bmatrix} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \\ \text{CLAUSE-TYPE} & \text{INT} \end{bmatrix}$$

(5.252)
$$\begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{‘PRO’}\end{bmatrix} \\ \text{ADJ} & \left\{ \boxed{2}\begin{bmatrix} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \right\} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \left\{\boxed{2}\right\} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \end{bmatrix}$$

Note that the subject in (5.251) and (5.252) is structure shared (the same variable is used). Also, the modifier *kiedy* is shared by both coordinated clauses in (5.252).

The full f-structure corresponding to (5.250) is provided in (5.253):

(5.253)
$$\begin{bmatrix} \left\{ \begin{matrix} \begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix}\text{PRED} & \text{‘PRO’}\end{bmatrix} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \\ \text{CLAUSE-TYPE} & \text{INT} \end{bmatrix}, \\ \begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \left\{ \boxed{2}\begin{bmatrix} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{bmatrix} \right\} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \left\{\boxed{2}\right\} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \end{bmatrix} \end{matrix} \right\} & \\ \text{COORD-FORM} & \text{I} \end{bmatrix}$$

Let us now discuss the process of constructing the f-structure corresponding to (5.249). The partial f-structure in (5.235) corresponds to the lexico-semantic coordination (*Czy i kiedy*), while (5.254) is the partial f-structure for the coordinate verbal phrase (*wróci lub napisze*). As previously, these partial f-structures are unified to yield the full f-structure – the desired resulting f-structure representation of (5.249) is provided in (5.255).

(5.254)
$$\begin{bmatrix} \left\{ \begin{bmatrix} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{bmatrix}, \begin{bmatrix} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{bmatrix} \right\} & \\ \text{COORD-FORM} & \text{LUB} \end{bmatrix}$$

(5.255)
$$\left[\begin{array}{l} \left\{\begin{array}{l} \left[\begin{array}{l} \left\{\left[\begin{array}{ll} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right] \end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{array}\right]\right\} \\ \begin{array}{ll} \text{COORD-FORM} & \text{LUB} \\ \text{CLAUSE-TYPE} & \text{INT} \end{array} \end{array}\right], \\ \left[\begin{array}{l} \left\{\left[\begin{array}{ll} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \boxed{2} \\ \text{UDF} & \boxed{3}\left\{\boxed{2}\left[\begin{array}{ll} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{array}\right]\right\} \end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \\ \text{ADJ} & \boxed{2} \\ \text{UDF} & \boxed{3} \end{array}\right]\right\} \\ \begin{array}{ll} \text{COORD-FORM} & \text{LUB} \end{array} \end{array}\right] \end{array}\right\} \\ \begin{array}{ll} \text{COORD-FORM} & \text{I} \end{array} \end{array}\right]$$

It differs from (5.253) in that *kiedy*, the lexico-semantic conjunct other than *czy*, is placed in the UDF attribute in respective two clauses and it is structure-shared as elements of respective adjunct sets.

There is, however, a fundamental problem with the f-structure representation of (5.249) provided in (5.255) – it is not going to be produced by the LFG theory in its current shape as the result of unifying (5.235) with (5.254). The problem is the unification of two partial f-structures which contain sets: the f-structures in (5.235) and (5.254) would be unified, yielding (5.256) – a coordinate structure which contains the COORD-FORM attribute with conflicting values and a set which is the union of the two sets from the relevant partial f-structures.

(5.256) *
$$\left[\begin{array}{l} \left\{\begin{array}{l} \left[\begin{array}{ll} \text{CLAUSE-TYPE} & \text{INT} \end{array}\right], \left[\begin{array}{ll} \text{ADJ} & \left\{\boxed{3}\right\} \\ \text{UDF} & \left\{\boxed{3}\left[\begin{array}{ll} \text{PRED} & \text{‘WHEN’} \\ \text{TYPE} & \text{INT} \end{array}\right]\right\} \end{array}\right], \\ \left[\begin{array}{ll} \text{PRED} & \text{‘RETURN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{array}\right], \left[\begin{array}{ll} \text{PRED} & \text{‘WRITE}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1} \end{array}\right] \end{array}\right\} \\ \begin{array}{ll} \text{COORD-FORM} & \text{I}\neq\text{LUB} \end{array} \end{array}\right]$$

By contrast, the desired result is (5.255) – in order to obtain such an f-structure, the partial f-structure in (5.254) would have to be distributed over the elements of the set contained in the f-structure fragment in (5.235). This is not possible, however, as LFG does not have special rules which would allow such a special unification of sets. This problem was first described on the occasion of discussing the LFG treatment of non-constituent coordination in Maxwell and Manning 1996[45] on the basis of (5.257), where the fragment *John flew to and Bill drove to* corresponds to one set and *Amsterdam on Monday and Brussels on Wednesday* to the other:

(5.257) John flew to and Bill drove to Amsterdam on Monday and Brussels on Wednesday.
(Maxwell and Manning 1996, ex. (42a))

[45] Thanks are due to John Maxwell and Ron Kaplan for discussing the problem described here.

To better understand this problem, let us return for a moment to how (5.234), the f-structure corresponding to (5.233), was produced. (5.233) differs from (5.249) in that it does not involve coordination of verbs. The partial f-structure corresponding to *Czy i kiedy* was provided in (5.235), while (5.164) corresponds to *wróci*. When these are unified, (5.164) distributes over the set in (5.235) and the well-formed f-structure in (5.234) results. This is not possible with (5.249) because the partial f-structures ((5.235) and (5.254)) contain two sets each and their unification results in the malformed f-structure in (5.256), namely a set containing four elements, instead of the desired (5.255).

This problem can[46] be solved by extending the LFG notation so that it makes it possible to specify explicitly the rules of unifying two structures containing sets, namely specifying which set should be distributed over the other. In the situation where one set contains f-structures with PRED as top-level attributes (as in (5.254)), while the other does not (compare (5.235)) – the desired effect would be to distribute the set containing PRED to the other set of f-structures, which would yield the f-structure in (5.255).

5.8 More types of conjuncts?

Previous sections presented data focusing on four types of conjuncts involved in lexico-semantic coordination, namely, as discussed in § 5.2, pronouns expressing free choice (shown in (5.258), repeated from (5.2)), universal quantifiers (as in (5.259)), *n*-words (see (5.260)) and *wh*-words (example given in (5.261)), which seem to have attracted most attention so far.

(5.258) czy komukolwiek, kiedykolwiek i do czegokolwiek przydał się poradnik
PART anybody.DAT anytime and for anything come in handy guide
'Has a(ny) guide ever come in handy to anybody for anything?' (NKJP)

(5.259) Zawsze i o wszystkim decyduje przypadek.
always and about everything decides chance
'Blind chance always decides about everything.' (Kallas 1993, p. 121, ex. (243))

(5.260) Nikt i nic, i nigdy go nie złamie.
nobody.NOM and nothing.NOM and never him NEG break
'Nobody and nothing will ever break him.' (Kallas 1993, p. 36, ex. (90))

(5.261) Co, komu i z czym się kojarzy, to jego prywatna sprawa.
what.NOM who.DAT and with what.INST REFL associate is his.GEN own business
'Who associates what with what is their own business.' (NKJP)

The LFG analysis presented in § 5.5 (first approach) and § 5.7 (improved) took only these classes of conjuncts into account. It seems, however, that the inventory of lexico-semantic types should be extended. This section discusses other items which may take part in lexico-semantic coordination on the basis of attested data taken from NKJP or found using the Google search engine.

[46] Thanks are due to John Maxwell for implementing a solution along these lines, providing a proof of concept.

5.8.1 Demonstratives

Kallas (1993) discusses the following example:

(5.262)

Jan	pamięta	tyle	i	takich	oskarżeń.
John	remembers	that many.ACC	and	such.GEN	accusations.GEN

'John remembers that many (of) such accusations.' (Kallas 1993, p. 53, ex. (39a))

There are similar attested examples:

(5.263)

my	nie	mogłybyśmy	zapewnić	naszym	podopiecznym	tylu	i	takich	materiałów	do	pracy
we	NEG	could	provide	our	charges	so many.GEN	and	such.GEN	resources.GEN	for	work

'We would not be able to provide our charges with so many (of) such work resources.' (NKJP)

(5.264)

Że	będzie	i	jest	tyle	i	takich	afer?
that	be.FUT	and	is	that many.ACC	and	such.GEN	scandals.GEN

'That there is and will be that many (of) such scandals?' (NKJP)

In (5.262) and (5.263) particular conjuncts correspond to the object (*tyle* and *tylu*, respectively) and the modifier of the object's object (*takich*). (5.264) shows lexico-semantic coordination of the subject and the modifier of the subject's object. It seems that the common feature of lexico-semantic conjuncts in the examples presented above is the fact that they belong to the class of demonstratives.

The f-structures provided in (5.265)–(5.266) correspond to (5.262) and (5.263), respectively.

(5.265)

$$\left[\begin{array}{ll}
\text{PRED} & \text{`REMEMBER}\langle\boxed{1},\boxed{2}\rangle\text{'} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`JAN'}\end{array}\right] \\
\text{OBJ} & \boxed{2} \\
\text{UDF} & \left\{\left[\begin{array}{l}
\left\{\boxed{2}\left[\begin{array}{ll}
\text{PRED} & \text{`THAT\_MUCH}\langle\boxed{3}\rangle\text{'} \\
\text{OBJ} & \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`ACCUSATION'} \\ \text{CASE} & \text{GEN} \\ \text{ADJ} & \left\{\boxed{4}\right\}\end{array}\right] \\
\text{ACM} & \text{REC} \\
\text{CASE} & \text{ACC} \\
\text{TYPE} & \text{DEM}
\end{array}\right], \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{`SUCH'} \\ \text{CASE} & \text{GEN} \\ \text{TYPE} & \text{DEM}\end{array}\right]\right\} \\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]\right\}
\end{array}\right]$$

(5.266)

$$
\left[\begin{array}{ll}
\text{PRED} & \text{‘COULD}\langle\boxed{2}\rangle\boxed{1}\text{’}\\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{‘PRO’}\end{array}\right]\\
\text{XCOMP} & \boxed{2}\left[\begin{array}{ll}
\text{PRED} & \text{‘PROVIDE}\langle\boxed{1},\boxed{3},\boxed{4}\rangle\text{’}\\
\text{SUBJ} & \boxed{1}\\
\text{OBJ} & \boxed{3}\\
\text{OBJ}_{\theta} & \boxed{4}\left[\begin{array}{ll}\text{PRED} & \text{‘CHARGES’}\\ \text{CASE} & \text{DAT}\end{array}\right]\\
\text{UDF} & \left\{\left[\begin{array}{l}
\left\{\boxed{3}\left[\begin{array}{ll}
\text{PRED} & \text{‘THAT\_MUCH}\langle\boxed{5}\rangle\text{’}\\
\text{OBJ} & \boxed{5}\left[\begin{array}{ll}\text{PRED} & \text{‘RESOURCE’}\\ \text{CASE} & \text{GEN}\\ \text{ADJ} & \left\{\boxed{6}\right\}\end{array}\right]\\
\text{ACM} & \text{CONGR}\\
\text{CASE} & \text{GEN}\\
\text{TYPE} & \text{DEM}
\end{array}\right], \boxed{6}\left[\begin{array}{ll}\text{PRED} & \text{‘SUCH’}\\ \text{CASE} & \text{GEN}\\ \text{TYPE} & \text{DEM}\end{array}\right]\right\}\\
\text{COORD-FORM} \quad \text{I}
\end{array}\right]\right\}
\end{array}\right]\\
\text{NEG} & +
\end{array}\right]
$$

The fact that lexico-semantic conjuncts contain demonstratives is signalled by the DEM value of the TYPE attribute in the f-structures of relevant items. It is worth noting that these examples show the interaction with structural case assignment to objects: both PAMIĘTAĆ ‘remember’ and ZAPEWNIĆ ‘provide’ mark their object for structural case. The object in (5.262) is marked for the accusative case (see (5.265)), while in (5.263) it is marked for the genitive case due to the fact that it is in scope of sentential negation (the higher predicate, *moglibyśmy*, is negated; see § 3.2.3.2 for a discussion of structural case assignment to objects in Polish) as shown in (5.266).

5.8.2 Indefinites

The example in (5.267) features conjuncts which are indefinite pronouns[47] – their TYPE is INDEF in the f-structure representation provided in (5.268):[48]

(5.267)	Ktoś,	gdzieś	i	coś	mocno	pokiełbasił.
	someone.NOM	somewhere	and	something.ACC	really	messed up

‘Someone really messed something up somewhere.’ (NKJP)

[47] Indefinite pronouns could be analysed as instances of existential quantifiers.

[48] The f-structure in (5.268) shows why it is the element of the ADJ set that is structure shared with the relevant dependent rather than the entire adjunct set (as mentioned in fn. 39) – in (5.267) there is another adjunct, *mocno* ‘really’, which is represented in (5.268) as another member of the ADJ set.

(5.268)

$$\left[\begin{array}{ll}
\text{PRED} & \text{`MESS\_UP}\langle\boxed{1},\boxed{2}\rangle\text{'}\\
\text{SUBJ} & \boxed{1}\\
\text{OBJ} & \boxed{2}\\
\text{ADJ} & \left\{\boxed{3},\left[\begin{array}{ll}\text{PRED} & \text{`REALLY'}\end{array}\right]\right\}\\
\text{UDF} & \left\{\left[\begin{array}{l}
\left\{\boxed{1}\left[\begin{array}{ll}\text{PRED} & \text{`SOMEBODY'}\\ \text{CASE} & \text{NOM}\\ \text{TYPE} & \text{INDEF}\end{array}\right],\ \boxed{3}\left[\begin{array}{ll}\text{PRED} & \text{`SOMEWHERE'}\\ \text{TYPE} & \text{INDEF}\end{array}\right],\ \boxed{2}\left[\begin{array}{ll}\text{PRED} & \text{`SOMETHING'}\\ \text{CASE} & \text{ACC}\\ \text{TYPE} & \text{INDEF}\end{array}\right]\right\}\\
\text{COORD-FORM}\quad \text{I}
\end{array}\right]\right\}
\end{array}\right]$$

5.8.3 Free relatives

Another class is constituted by free relatives. Citko and Gračanin-Yüksek (2012) discussed such coordination on the basis of data from Polish, English and Croatian. They provide the following Polish examples:

(5.269) Jan je cokolwiek i kiedykolwiek Maria gotuje.
John eats whatever and whenever Mary cooks
(Citko and Gračanin-Yüksek 2012, ex. (1b))

(5.270) *Jan je cokolwiek kiedykolwiek Maria gotuje.
John eats whatever whenever Mary cooks
(Citko and Gračanin-Yüksek 2012, ex. (1a))

In (5.269) two free choice (-*kolwiek* type) pronouns are coordinated: an object (*cokolwiek*) and an adjunct (*kiedykolwiek*). In their abstract, Citko and Gračanin-Yüksek (2012) claim that using multiple free relatives without coordination is ungrammatical in Polish, providing (5.270) to substantiate this claim. Though (5.270) is perfectly grammatical for some speakers, it may be made more acceptable for speakers who find it controversial by using different aspect (perfective), as in (5.271), or different word order, as in (5.272):

(5.271) Jan (z)je cokolwiek kiedykolwiek Maria (u)gotuje.
John will eat whatever whenever Mary will cook

(5.272) Cokolwiek kiedykolwiek Maria (u)gotuje, Jan (z)je.
whatever whenever Mary will cook John will eat

It seems therefore that it is not the case that multiple free relatives are prohibited by Polish syntax, which is important for the issue of representation of such lexico-semantic coordination. Since examples with multiple free relatives prove to be grammatical upon closer scrutiny (contra the claims of Citko and Gračanin-Yüksek (2012)), they could be analysed as monoclausal lexico-semantic coordination and handled by the relevant rules (see § 5.7.1) – the f-structure corresponding to (5.269) is provided in (5.273).

(5.273)
$$\left[\begin{array}{ll}
\textsc{pred} & \text{‘}\textsc{eat}\langle\boxed{1},\boxed{2}\rangle\text{’}\\
\textsc{subj} & \boxed{1}\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{Jan}\text{’}\end{array}\right]\\
\textsc{obj} & \boxed{2}\\
\textsc{adj} & \left\{\boxed{3}\right\}\\
\textsc{udf} & \left\{\left[\begin{array}{l}
\left\{\boxed{2}\left[\begin{array}{ll}
\textsc{pred} & \text{‘}\textsc{whatever}\text{’}\\
\textsc{case} & \textsc{acc}\\
\textsc{type} & \textsc{any}\\
\textsc{adj} & \left\{\boxed{5}\left[\begin{array}{ll}
\textsc{pred} & \text{‘}\textsc{cook}\langle\boxed{4},\boxed{2}\rangle\text{’}\\
\textsc{subj} & \boxed{4}\left[\begin{array}{ll}\textsc{pred} & \text{‘}\textsc{Maria}\text{’}\end{array}\right]\\
\textsc{obj} & \boxed{2}\\
\textsc{adj} & \left\{\boxed{3}\right\}
\end{array}\right]\right\}
\end{array}\right],\ \boxed{3}\left[\begin{array}{ll}
\textsc{pred} & \text{‘}\textsc{whenever}\text{’}\\
\textsc{type} & \textsc{any}\\
\textsc{adj} & \left\{\boxed{5}\right\}
\end{array}\right]\right\}\\
\begin{array}{ll}\textsc{coord-form} & \textsc{i}\end{array}
\end{array}\right]\right\}
\end{array}\right]$$

5.8.4 Relatives

Citko and Gračanin-Yüksek (2012) claim that there is "a more general constraint that rules out two relative pronouns in a relative clause modifying a single head, regardless of whether the relative pronouns are coordinated or not" and provide example (5.274) in support of this claim:

(5.274) *student którego (i) któremu Maria przedstawiła
student which.ACC and which.DAT Mary introduced
(Citko and Gračanin-Yüksek 2012, ex. (11a))

While (5.274) is indeed ungrammatical, it does not prove that coordinating relatives always results in ungrammaticality. Counterexamples exist, see constructed (5.275) and attested (5.276):

(5.275) człowiek, z którym *(i) o którym lubię mówić
man with which and about which like talk
'the man with whom and about whom I like to talk'

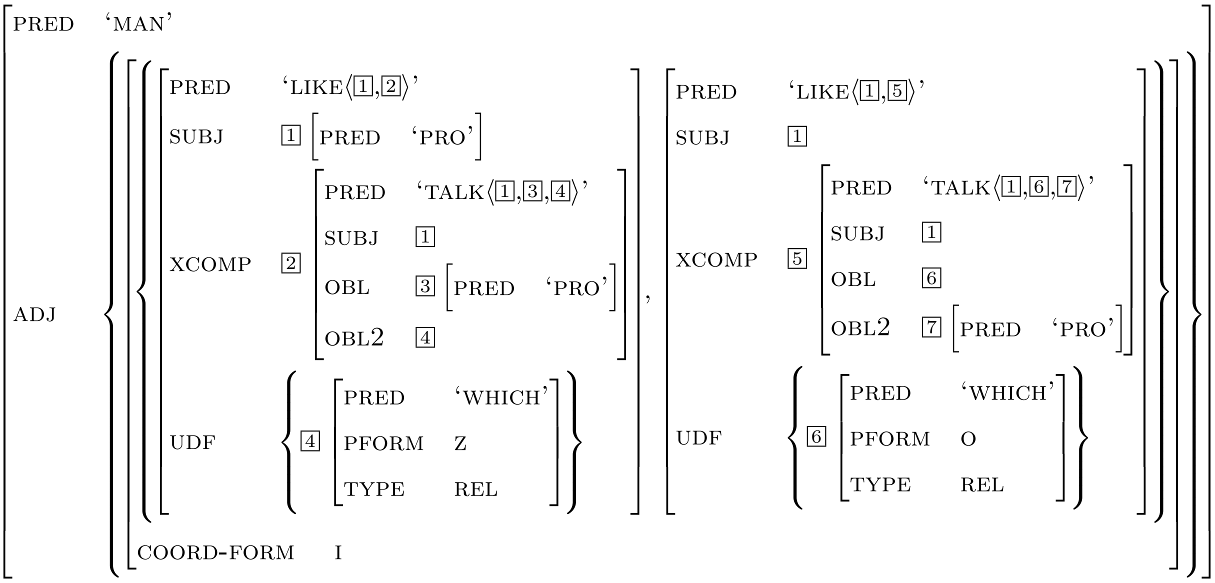
(5.276) SŁOWA tej księgi pozwalają budować człowieka któremu i z którym jest dobrze żyć.
words this book let build man which.DAT and with which is good live
'Words of this book let one build a man for whom it is good to live and with whom it is good to live.' (NKJP)

As indicated in (5.275), lexico-semantic coordination of relative items seems possible, but multiple use of relatives without coordination is ungrammatical. This would suggest that such examples should be handled in a way similar to sentences featuring lexico-semantic coordination with CZY, the *yes/no* question particle (see the discussion in § 5.4.2.16), leading to the adoption of the multiclausal analysis (see § 5.7.2) – the rule provided in (5.277) is a simple modification of (5.242);[49] it produces the f-structure shown in (5.278) for (5.275).

[49] If there are instances of lexico-semantic coordination of relatives featuring more than two conjuncts (consider (i) below), the representation in § 5.7.2.1 should be used instead of the one in § 5.7.2.2 as the basis for introducing

(5.277) $\text{XPlxb}_{rel} \rightarrow \underset{\downarrow\in\uparrow}{\text{XPextrbicl}_{rel}} \quad \underset{\uparrow=\downarrow}{\text{CONJ}} \quad \underset{\downarrow\in\uparrow}{\text{XPextrbicl}_{rel}}$

(5.278)



Such an analysis seems to be consistent with the fact that, while it is impossible to use multiple relatives without coordination (as shown in (5.275)), one may coordinate relative clauses:

(5.279) człowiek, [z którym lubię mówić] i [o którym lubię mówić]
man with whom like talk and about whom like talk
'the man with whom I like to talk and about whom I like to talk'

5.8.5 Modification

This section discusses the issue of how selected pronouns taking part in lexico-semantic coordination can be modified. This section is organised into subsections which show modification patterns possible with certain types of conjuncts taking part in lexico-semantic coordination.

5.8.5.1 ***Wszyscy/każdy*-based**

Pronouns which correspond to the universal quantifier can be modified using the negative particle *nie* or by the word *prawie* 'almost' (note that original examples include the words in brackets):

(5.280) Nie wszyscy i *(nie) każdemu mogą sprzedać broń.
NEG all.NOM and NEG everybody.DAT can sell arms
'It is not the case that everyone can sell arms to everybody.' (Google)

(5.281) Prawie wszyscy i (prawie) zawsze są uśmiechnięci.
almost all.NOM and almost always are smiling
'Almost everyone and almost always is smiling.' (Google)

modifications. This is because (5.230) in § 5.7.2.1 can produce f-structures containing more than two clauses, so that each relative pronoun would be placed in a different clause.

(i) człowiek, z którym, o którym i przy którym lubię mówić
man with which about which and near which like talk
'the man with whom, about whom and near whom I like to talk'

These modifiers change the semantics of conjuncts taking part in lexico-semantic coordination: conjuncts in (5.280) have the semantics of the universal quantifier under the scope of negation (not every: there is an element which does not satisfy the relevant condition). The effect of the modifier in (5.281) is similar, though it is not as simple to formalise as the effect of negation in (5.280): *prawie* does not have well-defined semantics, it can refer in different contexts to different proportions of some amount (e.g. 99%, 90%, 80%).

On a side note, it is perhaps worth mentioning that while negation must be used with each conjunct in (5.280), *prawie* in (5.281) can scope over the entire coordinate phrase – *prawie* in the second conjunct is not necessary (though it was present in the original example). This contrast suggests that particular conjuncts in (5.281) are not multiword lexical items (*prawie wszyscy*, *prawie zawsze*), but they are phrases constructed from single words, one being *prawie*, the other being a pronoun expressing universal quantification.

Though negation is obligatory with each conjunct in (5.280), this does not necessarily mean that *nie wszyscy* and *nie każdemu* are multiword items. It seems that the negative particle has narrow scope, as in other contexts – see the examples with verbal negation below:

(5.282) przez kilka dni nic nie jadł i nie pił.
for some days nothing.GEN NEG ate and NEG drunk
'For a few days he did not eat anything and did not drink.' (NKJP)

(5.283) przez kilka dni nic nie jadł i pił.
for some days nothing.GEN NEG ate and drunk
'For a few days he did not eat anything and he was drinking.'

Under coordination both verbs must be negated, as in (5.282), since a single negation marker cannot scope over the coordinate verbal phrase, as shown in (5.283), a modified version of (5.282).

5.8.5.2 Emphatic interrogatives

The conjuncts under lexico-semantic coordination in (5.284) are *wh*-words (KTO, GDZIE) to which the intensifying particle -*ż* is attached:

(5.284) A któż i gdzież postawił taką tezę…
but who.NOM and where proposed such thesis
'Who and where did propose such a thesis?' (Google)

5.8.5.3 Based on existentials homophonous with *wh*-words

Interrogative words such as KTO, CO, etc. are homophonous with pronouns whose meaning is existential. There are many possible ways of modifying such existentials, as shown below:

(5.285) W pracy mało kto i mało kogo tak naprawdę lubi.
at work few someone.NOM and few someone.ACC really likes
'Hardly anybody really likes hardly anyone at work.' (Google)

(5.286) Nie pije się byle czego, byle gdzie i z byle kim.
NEG drink REFL any something.GEN any somewhere and with any someone
'You don't drink whatever, wherever and with whoever.' (NKJP)

(5.287) Dziś w Wytwórni zagra nie byle kto i nie byle jak.
today in Wytwórnia plays NEG any someone.NOM and NEG any somehow
(Google)

(5.288) Ci bandyci... drwią z kogo bądź i gdzie bądź.
those thugs jeer from someone any and somewhere any
'Those thugs jeer at anyone and anywhere.' (Google)

In (5.285) *mało* 'few' is used as the modifier. As shown in the glosses, the English counterpart of *mało* is 'hardly' – again, such quantification is not precise.

(5.286) features the modifier *byle*, which was translated in glosses as 'any' – it changes the semantics of the modified existential pronoun into a free choice pronoun, but its meaning is marked in comparison to pronouns of *-kolwiek* type which are neutral. Unlike *-kolwiek* type pronouns, the meaning of pronouns with *byle* is marked negatively: the item picked in this way is any item, but the use of *byle* suggests that it was not an informed choice: this item was chosen without attention or due care.

(5.287) shows that negation can be used with such a pronoun, its meaning is changed accordingly: it becomes a pronoun which is restricted semantically so that the item picked is not just any element of the set, but a good one.[50]

In (5.288) the existential pronoun is modified using *bądź*, which was also translated in glosses as 'any', but, unlike *byle*, it does not have negative associations – the existential pronoun accompanied by *bądź* functions as a free choice pronoun similar in meaning to *-kolwiek* type pronouns.[51]

The following examples suggest that as long as the requirement of similar semantics of conjuncts is satisfied, pronouns can be coordinated using lexico-semantic coordination in spite of the fact that they are accompanied by different modifiers, or themselves belong to different classes:

(5.289) może dlatego mają takie opory przed oddaniem się komu bądź
maybe therefore have.3.PL such inhibition from giving REFL someone.DAT any
i byle kiedy?
and any sometime
'Perhaps that's why they have such inhibition against giving themselves to anyone and anytime?' (Google)

(5.290) Nie zapraszam też kogo bądź i jak leci.
NEG invite too someone.GEN any and how comes
'I don't invite anybody and whatever comes.' (Google)

[50] Selected definitions of *byle* from Słownik PWN (`http://sjp.pwn.pl/szukaj/byle.html`): "partykuła nadająca komuś lub czemuś cechę dowolności [...] lub wyrażająca obojętność wyboru" ("particle expressing freedom of choice when choosing someone or something, or expressing indifference when making the choice"), "partykuła wyrażająca negatywną ocenę czegoś lub kogoś [...] lub – w połączeniu z poprzedzającym *nie* – wyróżniająca kogoś lub coś jako zasługującego na uwagę" ("particle expressing a negative opinion on something or someone, or – in connection with preceding *nie* – distinguishing someone or something as deserving attention").

[51] One of the interpretations from Słownik PWN (`http://sjp.pwn.pl/haslo.php?id=2443328`): "partykuła [...] nadająca im znaczenie nieokreśloności, wyrażająca obojętność wyboru" ("particle [...] adding the meaning of indefiniteness, expressing that choice does not matter").

(5.291) Nie wydaje mi się, by udostępnianie boiska komu bądź i na każde
NEG seems me REFL that making available pitch someone.DAT any and for every
życzenie było dobrym pomysłem…
demand was good idea
'It doesn't seem to me to be a good idea to make available the pitch to whoever and at any demand.' (Google)

The first conjunct in (5.289), *komu*, is modified by *bądź* (as in (5.288)), while the modifier of the second conjunct, *kiedy*, is *byle* (see (5.286)). However, lexico-semantic coordination is possible because both conjuncts are free choice pronouns (though *byle* is more marked than *bądź*).

(5.290) is another instance of lexico-semantic coordination based on semantics rather than pronoun type: the first conjunct is a free choice item resulting from the combination of an existential pronoun *kogo* and *bądź*; the second conjunct, *jak leci*, is a phrase which can be roughly translated as 'whatever comes', so its meaning is also that of a free choice item.

The last example, (5.291), is even more interesting as it features the coordination of a free choice pronoun (*komu bądź*) and a phrase containing a universal quantifier (*na każde życzenie*), though the semantics of the latter is that of a free choice pronoun.

5.8.5.4 *Ktoś*-based

Indefinite pronouns expressing existential quantification such as *ktoś* 'someone' and *coś* 'something' can be modified by *tam* 'there':

(5.292) Jakbyś to opisywała bezosobowo ktoś tam i coś tam
as this describe impersonally someone.NOM there and something.ACC there
zrobił.
did
'As you would describe it impersonally: someone did something.' (Google)

Such modification does not, however, change the semantics of the resulting items: they still express existential quantifiers. The only potential contribution of the modifier *tam* is that it suggests that it is not important what the relevant pronoun refers to or that the speaker does not know or does not care what it refers to.

5.8.5.5 *Niektóry*-based

In (5.293) lexico-semantic coordination is based on the adjective NIEKTÓRY 'some':

(5.293) Podobno mężczyźni wiele obiecują, ale tylko niektórzy (i tylko niektóre
supposedly men much promise but only some.NOM and only some.ACC
obietnice) spełniają.
promises keep
'Supposedly men promise a lot, but only some keep some promises.' (Google)

Each conjunct in (5.293) is accompanied by the word *tylko* 'only' – it does not change the semantics of the items involved, though when it is used, the meaning of the entire phrase is not neutral – it adds emphasis, suggesting that only few elements of the set satisfy the relevant condition.

Interestingly, it seems that the modifier TYLKO must be used for lexico-semantic coordination with NIEKTÓRY to be possible.

5.8.5.6 Comparatives

Another class of lexico-semantic conjuncts which may be modified are comparatives:

(5.294) Współpracują częściej i z większą liczbą instytucji.
cooperate more often and with greater number institutions
'They cooperate more often with a greater number of institutions.' (Google)

(5.294) is a plain case of lexico-semantic coordination, where particular conjuncts contain items which are in comparative degree: an adverb (*częściej* 'more frequently') and an adjective (*większą* 'greater') which modifies the oblique object (*liczbą instytucji*). The remaining examples show modification patterns with comparatives:[52]

(5.295) W Sieci kupujemy coraz więcej i coraz częściej.
in net buy increasingly more.ACC and increasingly more often
'We buy more and more and more and more often on the Internet.' (Google)

(5.296) Wspominam coraz mniej i coraz rzadziej.
reminisce increasingly less.ACC and increasingly less often
'I reminisce less and less and less and less often.' (Google)

(5.297) będzie powoli jeść coraz więcej i coraz rzadziej
AUX slowly eat increasingly more.ACC and increasingly less often
'She will eat more and more and less and less often.' (Google)

(5.295) and (5.296) show that comparatives can be modified using *coraz* 'increasingly', which stresses the increasing character of changes taking place. While there is a tendency for conjuncts to express the fact of increasing or decreasing consistently (as in (5.295) and (5.296), respectively), it is not necessary, as evidenced in (5.297).

Furthermore, there are modifiers whose purpose is to add emphasis. These are used with conjuncts in positive degree, as in the examples provided below:[53]

(5.298) Potrzebują od rodziców tak wiele i tak często!
need from parents so much.GEN and so often
'They so often need so much from their parents!' (Google)

[52] It seems that it is possible to analyse Polish numeral forms such as *więcej* and *mniej* (see (5.295)–(5.297)) as specified for comparative degree. Similarly, there seems to be a preposition which can be specified for degree, namely BLISKO: *blisko (domu)* 'close (to house)', *bliżej* 'closer', *najbliżej* 'closest'; moreover, it is possible to use analytic degree modifiers with BLISKO: *bardziej blisko (domu)* (comparative), *najbardziej blisko (domu)* (superlative).

[53] Though all sentences in (5.298)–(5.301) contain the word *wiele*, it is glossed as specified for different case values. In (5.298) it is the lexical genitive case, while in the remaining examples it is a realisation of structural case assigned to the object (see § 3.2.3–§ 3.2.3.2 for discussion): in (5.299)–(5.300) it is marked for the genitive case due to being in scope of sentential negation, while in (5.301) the object bears the accusative case as it is not in scope of negation.

(5.299) Nie piszę zbyt wiele i zbyt często.
NEG write too much.GEN and too often
‘I don’t write too much too often.’ (Google)

(5.300) Staram się za wiele i za często nie kupować.
try REFL too much.GEN and too often NEG buy.INF
‘I try not to buy too much too often.’ (Google)

(5.301) Fakt, że pije za wiele i zbyt często, nie jest zależny od jego woli.
fact that drinks too much.ACC and too often NEG is dependent from his will
‘The fact that he drinks too much and too often does not depend on him.’ (Google)

In (5.298) the first conjunct is a numeral (*wiele*), the second one is an adverb, both are specified for positive degree, both are modified using *tak* ‘so’, which adds emphasis, but does not change the semantics of relevant phrases.

The modifiers in (5.299) and (5.300), *zbyt* and *za* ‘too’, respectively, express the fact that some limit was exceeded: *wiele* ‘much’ is neutral, while *zbyt/za wiele* ‘too much’ indicates excess. Though in these sentences the same modifiers are used consistently in both conjuncts, (5.301) shows that it is possible to use mixed modifiers (*za* in the first conjunct, *zbyt* in the second), probably due to the fact that their semantics is the same.

5.8.5.7 *N*-word-based

It is possible to modify *n*-words under lexico-semantic coordination:

(5.302) Przykładowo – taka miernota jak jakiś Tadas – o którym nikt więcej
for example such mediocrity as some Tadas about whom nobody.NOM more
i nigdy więcej się nie dowie!!
and never more REFL NEG finds out
‘For example, such a mediocre person as some Tadas about whom no more person will ever find out more!’ (Google)

The use of *więcej* with both conjuncts in (5.302) restricts the meaning of *n*-words involved: their meaning is no longer absolute, the assumption is that somebody managed to get to know about Tadas at some point, but from now on nobody will ever find out about him.

5.8.5.8 Demonstratives

The example provided below shows that it is possible to use modifiers with demonstratives involved in lexico-semantic coordination:

(5.303) Każde dziecko musi dostać tyle samo i takich samych cukierków
every child must get that much.ACC same and such.GEN same.GEN candies.GEN
w tym samym czasie.
at such same time
‘Every child must get the same amount of identical candy at the same time.’
(Google)

In (5.303) both conjuncts, which are demonstrative elements (*tyle* 'that many' and *takich* 'such'), are modified by elements which express identity in comparison to some point of reference (the same amount of candy, the same type of candy for each child), *samo* and *samych*, respectively. Note that (5.303) is an example where one lexico-semantic conjunct is a dependent of the other: *takich samych cukierków*, the second conjunct, the nominal dependent (object) of *tyle samo*, the numeral in the first conjunct (see (5.6) for an example with a similar construction; see § 3.1.2.1 for discussion of the analysis of Polish numeral phrases).

5.8.6 Summary

On the basis of a range of attested examples, this section presented the diversity of conjuncts taking part in lexico-semantic coordination in Polish, investigating less frequent pronoun types. Moreover, it explored modification patterns observed in lexico-semantic coordination, showing the effects of various types of modification: a change of semantics of the items involved may result (as when negation is used), at other times the effect may be adding emphasis.

It must be noted, however, that examples such as the ones presented in this section are not as numerous, varied and productive as examples presented in previous sections, which concentrated on lexico-semantic coordination of *wh*-words, *n*-words, pronouns expressing universal quantifiers and -*kolwiek*-type free choice pronouns.

5.9 Conclusion

This chapter presented a variety of coordination which may be difficult conceptually and which may be challenging to various grammatical theories. It demonstrated that coordinated elements may not only correspond to different grammatical functions, but they may also depend on entirely different heads, unifying at various levels of f-structure representation.

It seems, however, that such coordination displays surprisingly consistent behaviour across various semantic classes, patterning into two well-defined classes according to the required representation: monoclausal and multiclausal. A critical review of arguments raised in the discussion of which representation should be adopted for lexico-semantic coordination was provided – its conclusion is that there is no convincing evidence for adopting the multiclausal representation except when the *yes/no* question particle CZY is one of the conjuncts or when relative pronouns are coordinated. In other contexts the monoclausal representation is adopted as the more economic one.

A detailed LFG analysis was offered – it is capable of producing both monoclausal and multiclausal representation, but the latter is only restricted to lexico-semantic coordination with CZY.[54] The proposed analysis takes interactions with various phenomena into account; these include: verbal coordination and embedding of lexico-semantic coordination. It was demonstrated that currently available LFG mechanisms are well-suited for modelling such phenomena.[55]

Finally, a description of less known items taking part in lexico-semantic coordination was offered in § 5.8, together with information on allowed modification patterns. The analysis of such instances requires further work, in the area of both syntax and semantics.

[54] Relevant rules can be extended to cover the coordination of relative pronouns – see the discussion in § 5.8.4.

[55] With the exception of the complex and subtle issue described in § 5.7.3.

Part III

Implementation

Chapter 6

About the implemented grammar

6.1 Introduction[1]

This chapter provides some background information about the implemented LFG grammar of Polish which covers (among others) the phenomena described in the two previous chapters, namely coordination of unlike categories (ch. 4) and lexico-semantic coordination (ch. 5).

The grammar was developed by adopting a parasitic approach to grammar development, whereby POLFIE, an LFG grammar of Polish, is created on the basis of a variety of resources, including a DCG[2]-like grammar for Polish, a treebank developed using this grammar and Walenty, a valence dictionary of Polish. The new grammar extends the original grammar "vertically", by adding the level of f-structure to the c-structure offered by the DCG grammar, and "horizontally", by attempting to cover a wider range of phenomena. The coverage of the LFG grammar as well as the quality of the analyses it offers is evaluated by building an LFG structure bank.

This chapter is structured as follows: § 6.2 lists and briefly describes the resources that the present LFG grammar builds upon. § 6.3 presents the process of grammar development in more detail, while § 6.4 outlines the adopted method of ensuring a reasonable quality of the grammar during its development.

6.2 Resources for grammar development

The effort of creating an LFG grammar implemented in the XLE platform (`http://www2.parc.com/isl/groups/nltt/xle/`; Crouch *et al.* 2011) consists of two major tasks: creating annotated rules and building the lexicon. Since manual development of large-scale grammars is a rather costly and time-consuming task, the adopted strategy is to reuse as many available resources as possible instead of developing another grammar from scratch. As there is a wide range of language resources for Polish at hand, it is possible to draw on the results of many projects, completed and ongoing, and minimise the workload, concentrating on further improvements.

[1] This chapter is based on the following papers: Patejuk and Przepiórkowski 2012c, 2015b, 2014e.

[2] *Definite Clause Grammar* (Warren and Pereira 1980).

6.2.1 Previous grammars

C-structure rules of the current LFG grammar are based on context-free grammar rules of GFJP2 (Gramatyka formalna języka polskiego 2; Formal grammar of Polish 2), which were first implemented for use by another parser for Polish, Świgra (`http://zil.ipipan.waw.pl/Świgra/`; Woliński 2004), on the basis of Świdziński's 1992 grammar (the original GFJP). These rules were annotated with instructions on how to build an additional level of structure on top of trees, namely the f-structure (see § 7.3 for an example). This provides a representation employing grammatical functions, which is considered more universal across languages than the constituent structure, which is subject to much variation (such as word order, for instance). The f-structure annotation was inspired by two resources: the original metamorphosis grammar used by Świgra and a small-scale but linguistically sophisticated HPSG grammar of Polish (Przepiórkowski *et al.* 2002).

6.2.2 Morfeusz

While most large-scale grammars implemented in XLE use XFST morphology (Beesley and Karttunen 2003) combined with an additional set of rules, namely sublexical rules, the current grammar relies on Morfeusz, the state-of-the-art morphological analyser for Polish (`http://sgjp.pl/morfeusz/`; Woliński 2006, Saloni *et al.* 2012, Woliński 2014). Therefore, rather than trying to build FST (finite state transducer) morphology for Polish from scratch – a very demanding task in itself – the output provided by Morfeusz is converted into ready-made XLE lexical entries which correspond to full, inflected forms.[3]

6.2.3 The National Corpus of Polish

The National Corpus of Polish, the largest currently available corpus of Polish, which contains around 1.8 billion segments (around 1.5 million words) out of which 1.2 million were manually annotated (NKJP1M subcorpus), is used in a twofold way. First, it may be used as one of alternative sources of information about morphosyntax and segmentation, which is necessary to create a lexicon. Morphosyntactic information is specified according to the NKJP tagset (Przepiórkowski and Woliński 2003, Przepiórkowski 2009), which additionally provided the names of many attributes and values in the f-structures created by the Polish LFG grammar, especially the non-standard ones. It is worth mentioning that morphosyntactic intepretations available for every segment in NKJP were disambiguated (automatically or, in case of the manual subcorpus, by human annotators), which results in far fewer possibilities than provided for the same segment by Morfeusz. Secondly, NKJP provides a rich body of interesting examples, which makes it possible to ensure that further extensions of the grammar have firm empirical grounding.

[3] At the moment of writing a new version of the grammar is under development: it uses a grammar library transducer (`http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#clm3.5`) written by Katarzyna Krasnowska, which simulates the output of XFST morphology on the basis of information from Morfeusz. The grammar was modified so as to use sublexical rules.

6.2.4 Składnica

The next resource actively used in the development of Polish LFG is Składnica (Woliński 2010, Świdziński and Woliński 2010, Woliński *et al.* 2011; `http://zil.ipipan.waw.pl/Składnica/`), a treebank containing parse trees selected by human annotators from the rich output generated by Świgra for selected sentences from NKJP1M, the manually annotated subcorpus of NKJP. Składnica serves as the main testbed for the current grammar, ensuring backwards compatibility with the original grammar and checking grammar coverage on authentic texts. The information about morphosyntax and segmentation from manually disambiguated trees is converted into XLE lexical entries, which considerably reduces the amount of interpretations in comparison to Morfeusz.

6.2.5 Walenty

The last resource which is used here is Walenty (`http://zil.ipipan.waw.pl/Walenty/`; Przepiórkowski *et al.* 2014b,c), a new valence dictionary for Polish. Unlike its predecessor, the Syntactic Dictionary of Polish Verbs (SDPV, Świdziński 1994, 1998) used by Świgra, Walenty provides information about coordination possibilities for each argument. Moreover, it marks the subject grammatical function explicitly, so it is clear when a predicate takes a non-canonical subject or when it takes no subject at all. Walenty also distinguishes which objects can passivise (becoming the subject under passive voice) and it provides an account of control relations, namely which argument acts as the controller and which arguments are controlled, both for infinitival complements and predicative complements (adjectives and nominals, possibly embedded inside a prepositional phrase). Finally, it takes into account structural case assignment to relevant arguments. For more discussion, see ch. 8.

6.3 Towards an LFG grammar for Polish

As already mentioned, grammar development in XLE can be roughly divided into the creation of rules and the lexicon – this section presents an overview of this process, while the following chapters provide more detail.

6.3.1 Annotating c-structure with f-descriptions

The original c-structure rules provided by GFJP2, the grammar currently used by Świgra (and constantly developed as new trees are added to Składnica), were manually rewritten so as to comply with XLE notational conventions (see § 7.3 for an example). Even though this conversion could probably have been done automatically, there are some gains stemming from adopting the manual approach. Some linguistic generalisations expressed in rules, at the level of syntax, were transferred to the lexicon or gathered from various places in the grammar and stored in new syntactic templates (see § 7.5 on how subject-verb agreement is handled). The grammar writer (the author of this dissertation) also had the chance to better understand the mechanisms employed by the original grammar, which led in some cases to a decision to adopt a different analysis, either better motivated linguistically or more suitable from the perspective of the LFG formalism (see § 7.4.2 for the discussion of structural case assignment to objects; coordination is another example).

Adding f-structure annotation to the obtained c-structure required in the first place the identification of grammatical functions appropriate for Polish (see § 4.2.1 and § 8.3.1). This choice was made on the basis of the LFG literature, as well as the solutions adopted within the ParGram project (see § 6.4.5), taking the specifics of Polish into account.

Analyses of many linguistic phenomena offered by the original DCG grammar could often be translated into the f-structure representation almost unchanged. There are, however, some significant differences, especially in the area of agreement, case assignment and negation, where the LFG analysis draws broadly from existing HPSG analyses of these phenomena (Przepiórkowski 1999, Przepiórkowski *et al.* 2002).

Furthermore, the LFG grammar has undergone major c-structure changes. These changes have framework-independent motivation and are aimed at providing a better model of the interaction of some phenomena (such as coordination, negation and case assignment) and obtaining better f-structures. Therefore, rather than using original flat structure analyses[4] in the relevant areas, the LFG grammar adopts a hierarchical structure, which better accommodates relevant phenomena. It is perhaps worth noting that the choice of these analyses over the original ones was largely based on data from NKJP.

Last but not least, in contrast to GFJP2, which treats punctuation as a syntactic issue and consequently models it in its rules, the LFG grammar leaves the phenomenon of punctuation haplology at the discretion of the tokenizer,[5] following the practice adopted in ParGram grammars.

6.3.2 Lexicon creation

The morphosyntactic information necessary for the construction of a lexicon may be provided by Morfeusz, but it may also be extracted from manually disambiguated parse trees taken from the treebank (Składnica) or from NKJP (from the manually annotated subcorpus, NKJP1M, for instance). Data obtained from any of these sources is passed on to part-of-speech or lexicalised templates which are bundles of calls to simple templates, which set the values of appropriate features, etc.

There is a wide range of lexicalised information, mainly in the form of valence information, which accompanies morphosyntactic data in the lexicon. Valence information is taken from Walenty (see § 6.2.5). Since LFG states valence requirements in terms of grammatical functions rather than c-structure categories, the valence dictionary was automatically converted to an appropriate format – the process of converting Walenty (discussed in ch. 8) yielded better, more accurate results than when dictionaries distributed with Świgra were used – this is attributable to the fact that Walenty is considerably larger and it provides more information (for instance about structural case, passivisation, control, etc.) than the latter.

[4] It is worth noting that some GFJP2 rules were changed following the introduction of changes in the LFG grammar, improving the analysis of coordination.

[5] The XFST script used for compiling a tokenizer for use with XLE was written by Ron Kaplan.

6.4 Quality control

The evaluation of the LFG grammar of Polish is performed against three independent measures: constructed testsuites, authentic sentences from the treebank (Składnica), and authentic sentences from the corpus (NKJP1M, a manually annotated subcorpus of NKJP containing 1.2 million segments). The aim of testing using constructed examples is to ensure that the grammar correctly models particular linguistic phenomena. Testing based on sentences from the treebank checks the level of compatibility with the grammar which provided the original c-structure (GFJP2). Finally, testing on sentences from the corpus checks the grammar for robustness and real-life coverage.

6.4.1 Constructed testsuites

There are approximately 1200 constructed testsuite sentences. More than 700 were designed specifically for the purposes of the present implementation while the remainder was provided by testsuites which were used in the development of earlier grammars. These include around 200 constructed sentences extracted from the source code of GFJP2 and around 300 elicited sentences (Marciniak *et al.* 2003) which were used for testing the HPSG grammar of Polish described in Przepiórkowski *et al.* 2002.

It is worth noting that, by contrast with treebank sentences, constructed testsuites are not limited to positive examples – almost half of these sentences are negative examples, which are not supposed to be accepted by the grammar. While treebank testing is the main method of ensuring a reasonable overall coverage of the grammar, constructed testsuites provide an indispensable measure of ensuring high quality of the linguistic analysis, making it possible to detect minute changes and identify potential problems as early and precisely as possible.

6.4.2 Treebank testing

The other method of evaluation is treebank testing, which takes the form of reparsing sentences from Składnica for which human annotators identified a correct parse among the trees provided by Świgra (sentences belonging to the FULL type). The most recent results amount to 77.6% out of 9011 such sentences (the version of Składnica of April 2014; Składnica is under development, it currently contains over 10000 sentences classified as FULL).

The remaining 22.4% are mainly sentences which were not parsed due to the fact that the limit of available resources, time in most cases (18.1%), or memory, was exceeded. Only 3.4% sentences failed because the grammar could not produce a structure for them.

A sample of such problematic sentences was subsequently parsed manually in fragments and the obtained c- and f-structures were inspected carefully. Fragments were chosen so as to constitute a representative subset of the original sentence. The results of this experiment suggest that the grammar would accept such sentences if it was not for issues related to resources described above.

It seems that introducing changes in the c-structure and limiting the reliance on f-structure constraints at the same time (through the use of more parameterised c-structure rules,[6] for

[6] Parameterised rules is a different name for complex categories – this formal device was used in ch. 5 when

instance) could be a viable solution to this problem.

6.4.3 NKJP1M testing

As mentioned in § 6.2.3, NKJP1M, the manually annotated subcorpus of NKJP, contains 1.2 million segments (approximately), which corresponds to 85663 sentences (exactly). Składnica (see § 6.2.4) is based on a subset of NKJP1M: the version of April 2014 contains 19959 sentences taken from NKJP1M, 9011 out of which have a good parse chosen by human annotators (FULL type).

Since the entire Składnica (the version of April 2014 mentioned above) is around 23.3% of NKJP1M and sentences which have a good parse in Składnica amount to 10.5% of NKJP1M, the grammar is also tested by parsing NKJP1M itself. The reason for this is to assess how well the grammar copes with parsing sentences which were not included in Składnica.

The results of parsing NKJP1M are not as good as the results of reparsing Składnica: 33.2% of sentences from NKJP1M were parsed (this includes sentences which are present in Składnica), while 77.6% of FULL type sentences from Składnica can be parsed (sentences which have a good parse identified by human annotators).

6.4.4 Building LFG structure bank for Polish[7]

An LFG structure bank is currently being created with the help of the INESS infrastructure for building structure banks (Rosén *et al.* 2007). It is based on Składnica (see § 6.2.4) in the sense that sentences from Składnica are reparsed using the LFG grammar with a lexicon created from disambiguated morphosyntactic interpretations taken from Składnica (which in turn were taken from NKJP1M, the manually annotated subcorpus of NKJP; see § 6.2.3) and valence information provided by schemata from Walenty (converted into LFG; see § 6.2.5). It contains almost 6500 sentences, which have been manually disambiguated.

Sentences parsed with XLE are uploaded to INESS, where they are disambiguated by human annotators: each sentence is disambiguated independently by two people who cannot see each other's solution nor comments. Annotators can, however, communicate with each other and the grammar writer using the dedicated mailing list to discuss issues related to disambiguation.

Annotators perform the disambiguation by choosing discriminants describing the differences between structures. Discriminants may apply to different parts of structures: there are lexical, c-structure and f-structure discriminants. When disambiguating, annotators of Polish LFG structure bank are asked to choose f-structure discriminants whenever possible as these are considered less likely to change across versions. It is also worth noting that it is possible to choose a complement of a discriminant, indicating that some discriminant is not applicable in the given sentence and therefore choosing the (partial) solution which does not use this particular discriminant.

discussing rules handling lexico-semantic coordination of items of the same type (see § 5.5); see also the relevant part of XLE documentation: `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N3.4`. The use of this formal device makes it possible to shift some constraints from the f-structure to the c-structure, which is said to improve performance.

[7] An alternative method of building an LFG structure bank, namely through automatic partial disambiguation on the basis of the information extracted from trees in Składnica (see § 6.2.4), is described in Krasnowska and Kieraś 2013.

When there is no good analysis or the analysis offered could be improved, annotators write comments pointing to the problematic fragment and sometimes also identify the problem at hand. Some comments report problems related to the grammar (e.g. some coordinate structures are not accounted for), many comments report problems related to the valence schemata used in relevant sentences – a missing argument (a problem with Walenty), an argument not classified properly (usually an issue with Walenty, sometimes with the conversion), etc. Finally, some comments point out problems with morphosyntactic annotation taken from Składnica (originally from NKJP) such as wrong case marking, wrong choice of part of speech, etc., making it possible to correct these problems at the source. Currently, there are 2938 comments.

After a round of annotation is completed (2–3 weeks, around 1000 sentences), comments created by annotators are inspected by the grammar writer, who responds to each of them (after they have been anonymised) using the mailing list – is the comment right (sometimes explaining what is happening in the relevant case) or is it wrong (explaining why it is wrong). The purpose of this review is to give feedback to annotators (improving their skills by making them aware of certain linguistic issues, encouraging them to write comments) and to improve the grammar (by introducing relevant changes) as well as the valence dictionary.

Subsequently relevant comments containing confirmed issues are passed together with responses (and additional comments, if needed) to the developers of relevant resources. Developers of Walenty are asked to inspect relevant entries and introduce appropriate changes if need be, if the suggestion is right. Issues related to the conversion are handled by the grammar writer. Finally, comments related to problems in the grammar are collected and passed to the grammar writer to introduce appropriate modifications to improve the treatment of relevant phenomena.

After relevant changes have been introduced in Walenty and the grammar, a new lexicon is created, sentences are reparsed and a new version of parses is added to INESS so that discriminants can be reapplied from the previous disambiguated version of the structure bank. Discriminant choices are reapplied only if the relevant discriminant choice is still available in the new version. It is a very useful feature of INESS since it makes it possible to maximally reuse previous disambiguation work rather than start from scratch. After discriminants have been reapplied, annotators are asked to return to sentences from their annotation set which did not have a complete good solution in the previous version, return to their comments and check if the relevant problem has been solved in the current version.

6.4.5 ParGram

Finally, Polish LFG structures created using the present grammar have been taking part in the biannual ParGram structure comparison, an initiative to ensure cross-linguistic compatibility of a number of LFG grammars. The goal of the ParGram project (Butt *et al.* 2002; `http://pargram.b.uib.no/`) is to develop parallel grammars by means such as sharing a common set of features which are used in the f-structures and attempting to use similar analyses of particular linguistic phenomena across various languages.

A recent initiative of ParGram community was the creation of ParGramBank (Sulger *et al.* 2013), an LFG structure bank built with the help of INESS infrastructure, which makes it possible to manually disambiguate the parses in a convenient way and to align f-structures created for a range of typologically diverse languages (these include: English, Georgian, German, Hungarian, Indonesian, Norwegian, Polish, Turkish, Urdu and Wolof). ParGramBank consists of sentences prepared for structure comparison carried out during ParGram meetings – currently it contains 50–100 sentences for each language, which corresponds to sentences from one or two meetings, respectively. So far alignment (both at the level of c-structure and f-structure) was performed between language pairs, with one element of every pair being English.[8]

6.5 Summary

This chapter briefly discussed the general approach adopted to developing an LFG grammar of Polish, where the goal is to maximise the use of existing resources. These include previous implemented formal grammars of Polish (DCG and HPSG, § 6.2.1), which have greatly influenced the shape of the current grammar, a morphological analyser (Morfeusz, § 6.2.2), a large linguistically annotated corpus (NKJP, § 6.2.3) and a treebank (Składnica, § 6.2.4) as potential sources of morphosyntactic information used for the creation of the lexicon, and finally a valence dictionary (Walenty, § 6.2.5) as the source of valence information to be used in the lexicon.

The grammar is evaluated using a variety of methods (§ 6.4), which most notably include building a manually disambiguated LFG structure bank (§ 6.4.4).

[8] For a detailed discussion of alignment in INESS, see Dyvik *et al.* 2009.

Chapter 7

Implementation basics

7.1 Introduction

This chapter introduces the notation used in XLE (Xerox Linguistic Environment; `http://www2.parc.com/isl/groups/nltt/xle/`; Crouch *et al.* 2011), discussing the use of templates, local variables and comparing XLE constraint notation with notation used in theoretical LFG (§ 7.2), shows an example of how DCG rules may be rewritten to LFG rules in the XLE notation (§ 7.3) and finally presents how phenomena introduced in ch. 3 such as case assignment (§ 7.4) and verbal agreement (§ 7.5) are implemented in XLE.

7.2 Notation

This section briefly introduces the XLE notation used in this chapter.

7.2.1 Comments

There are two ways of adding comments, illustrated below:

(7.1)

```
"this is a comment
inside quotes"
```

(7.2)

```
<comment>
this is a comment
delimited by tags
</comment>
```

(7.1) contains a comment placed between quotes – such comments are usually short, though they can span more than one line. (7.2) is an example of a comment delimited by tags: `<comment>` starts a comment, while `</comment>` ends it – everything between these tags is considered to be a comment, it may span multiple lines.

7.2.2 Templates

The `@` character marks template calls (see § 2.2.3):

(7.3) `@ZERO-PARAMETER-TEMPLATE`

(7.4) `@(ZERO-PARAMETER-TEMPLATE)`

(7.5)
```
@(TEMPLATE-TAKING-PARAMETERS P1 P2 P3)
```

(7.3) and (7.4) are alternative ways of calling a template which takes no parameters,[1] while (7.5) is a call to a template which takes three parameters (`P1`, `P2` and `P3`).

Reading the templates presented in this chapter, one may be surprised that the definition of relevant templates as well as calls to these templates take much more space than if the relevant constraint was formalised without the use of any templates. For example, (7.7), the call to template `CASE-SUBC-PATH` defined in (7.6), introduces the constraint in (7.8), which is much shorter and can be considered more readable.

(7.6)
```
CASE-SUBC-PATH(PATH C) = "checks that case in PATH is equal to C"
(PATH CASE)=c C.
```

(7.7)
```
@(CASE-SUBC-PATH (^ SUBJ) acc)
```

(7.8)
```
(^ SUBJ CASE)=c acc
```

However, it does make sense to use templates when developing a grammar, especially a large grammar – one of the reasons is making grammar maintenance easy. If the grammar writer decides to make changes, for instance to use a different representation of case (complex case discussed in § 3.2.3.3 instead of case with atomic values), it is much easier if templates are used everywhere in the grammar when imposing constraints on case – once the definitions of relevant templates are changed accordingly, there is no need to make changes to template calls and all statements related to case are changed consistently. By contrast, if constraints related to case were formalised without the use of templates, the grammar writer would need to rewrite each and every such statement, trying hard to be consistent and not to make any mistakes.

7.2.3 Local variables

It is possible to define local variable assignment in XLE, as in (7.9), where `%V` is the variable to which the value of the path `(^ SUBJ)` is assigned.

(7.9)
```
%V=(^ SUBJ)
```

(7.10)
```
%V=(^ XCOMP* {SUBJ|OBJ|OBL})
```

Local variable assignment is particularly useful when the variable is used in more than one constraint – see the discussion of subject-verb agreement in § 7.5, where the `%L` variable is used to host the agreement controller: the entire subject for full agreement, § 7.5.1, or its closest conjunct for single conjunct agreement, § 7.5.2.1.[2] (7.9) assigns the value of a simple path to `%V`, so substituting its value instead of the variable in multiple constraints would have the same effect as using the variable. However, the effect would be different in the case of (7.10). If the variable assignment such as in (7.10) is used, it is evaluated once and substituted for

[1] Though XLE documentation uses the terms “argument” and “parameter” interchangeably (`http://www2.parc.com/isl/groups/nltt/xle/doc/walkthrough.html#W.templates`), only the latter is used in this work in order to avoid potential confusion caused by interference with the notion of argument used in Walenty to refer to a syntactic position.

[2] Note that default agreement (see § 7.5.2) may result with both agreement strategies (full and SCA) if the selected agreement controller triggers default agreement (i.e. it is a non-agreeing numeral, a clause, etc.).

the variable in all constraints which use it. By contrast, if relevant equations did not use the variable defined in (7.10) but placed `(^ XCOMP* {SUBJ|OBJ|OBL})`, the functional uncertainty corresponding to (7.10), instead of the variable, the result would be different. The functional uncertainty `(^ XCOMP* {SUBJ|OBJ|OBL})` would be resolved independently in each constraint – as a result, each constraint could use a different path, unlike when the local variable defined in (7.10) is used.

7.2.4 LFG notation vs XLE notation

The table in (7.11) provides a mapping between the LFG notation used in earlier chapters (see especially ch. 2) and the XLE notation used here and in the following chapters.[3]

(7.11)

LFG notation	XLE notation	comment
$\rightarrow$	`-->`	rewrite operator
$\uparrow$	`^`	metavariable (mother category)
$\downarrow$	`!`	metavariable (current category)
$\rightarrow$	`->`	off-path metavariable (structure inside this attribute)
$\leftarrow$	`<-`	off-path metavariable (structure containing this attribute)
$=$	`=`	defining equality
$=_c$	`=c`	constraining equality
$\neq$	`~=`	inequality
$\in$	`$`	set membership assignment
$\in_c$	`$c`	checking set membership
$\wedge$	*whitespace*	logical conjunction (implicit in XLE)
$\vee$	`|`	logical disjunction
$\neg$	`~`	negation
$*$	`*`	Kleene star (zero or more)
$+$	`+`	Kleene plus (one or more)

7.3 Rewriting rules

Let us see how two DCG rules from GFJP2 (see § 6.2.1) are translated into XLE code:

(7.12)

```
fpm(Pm, P, Kl, Zap, Neg, Dest, I, Pk, Sub) -->
        s(pm1),
        przyimek(Pm, P, na),
        fno(P, R/L, O, wym([],WymN), Kl, Zap, pre, Neg, Dest, I, Pk2, po),
        { oblpk(Pk, [(0|+), Pk2]) }.
```

(7.13)

```
fpm(Pm, P, Kl, Zap, Neg, Dest, I, Pk, Sub) -->
        s(pm2),
        fno(P, R/L, O, wym([],_), Kl, Zap, pre, Neg, Dest, I, Pk1, po),
        przyimek(Pm, P, na),
```

[3] The complete notation mapping used by XLE is available at this address: http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#NOA.

```
                { równe(Pm, naprzeciwko.temu),
                  oblpk(Pk, [Pk1, (+|0)]) }.
```

(7.12) and (7.13) are rules whose purpose is to handle prepositional phrases. Their left-hand side (before `-->`) is the mother category (`fpm`), their right-hand side (after `-->`) contains elements to which it rewrites (`s`, `przyimek` and `fno`). One right-hand side element is ignored when creating LFG rules: it is `s`, whose parameter is the unique identifier of the rule in the DCG grammar (`pm1` and `pm2` in (7.12) and (7.13), respectively). While the DCG grammar uses separate rewriting rules for the same category (`fpm` is the left-hand side element of more than one rule, see (7.12)–(7.13)), the LFG grammar uses one disjunctive rule, see (7.14), whose disjuncts correspond to the right-hand side of relevant DCG rules. Finally, the categories used in DCG rules (terminals and non-terminals) have parameters which store the values of various features relevant to particular categories. To pass the values of relevant features from one category to the other, from the head of the phrase to the entire phrase, for instance, variables are used: variables are upper-case values of parameters in (7.12) and (7.13). In both rules the form of `przyimek`, the preposition, bound to the variable `Pm`, is shared with `fpm`, the prepositional phrase. The same applies to the variable `P` which contains the case required by the preposition: it is shared with the mother category (`fpm`), but it is additionally shared with `fno`, the noun phrase. As a result, the case required by the preposition (`przyimek`) must match the case of the nominal (`fno`) – agreement is ensured using variables in rules. Two other important pieces of information are passed to the mother category (`fpm`) from the noun phrase (`fno`): the `Kl` variable hosts the class of the nominal (it may be interrogative, for example) and the `Neg` variable signals whether the nominal belongs to items which need negative concord licensing[4] (the availability of negation is marked as an appropriate value of `Neg` variable on the verb). The remaining parameters used in (7.12) and (7.13) will not be discussed here (but see Woliński 2004).

The XLE rule which corresponds to `fpm` defined in (7.12) and (7.13) is provided in (7.14):

(7.14)
```
PP -->
{
P
NP: {
    "non-semantic"
    ^=!
    |
    "semantic"
    @OBJ
    }
|
"head-final prepositions (only semantic, only with NAPRZECIWKO and TEMU)"
NP: @OBJ;
P: ^=!
   (^ PRED FN) $c {naprzeciwko temu}
}.
```

[4] Patejuk and Przepiórkowski 2014b briefly discusses how this is handled by the LFG grammar of Polish.

(7.15)
```
OBJ = @(GF ^ OBJ).
```

(7.16)
```
GF(PATH GF) = (PATH GF)=!.
```

While the DCG grammar uses category labels based on their full Polish names (*fraza przyimkowa* ‘prepositional phrase’: `fpm` in (7.12)–(7.13)), the XLE implementation of Polish LFG uses English names as these are more widely known (`PP` stands for ‘prepositional phrase’ in (7.14)).

In XLE every right-hand side category can bear functional annotation: it is attached to the name of the category following the “`:`” symbol. In the second disjunct of (7.14), `NP` bears the `@OBJ` annotation – it is a call to the template `OBJ` (as explained in § 7.2, calls to templates are marked by `@` before the name of the template) defined in (7.15), this template in turn makes a call to the template `GF` defined in (7.16). While `OBJ` is a zero-parameter template, `GF` takes two parameters: `PATH` and `GF`. When the `OBJ` template is called, it calls the `GF` template passing the following values to its relevant parameters: `^` is passed as `PATH` and `OBJ` is passed as `GF`. The following functional annotation results: `(^ OBJ)=!`; it is an XLE counterpart of (↑ OBJ)=↓ in the LFG notation used in previous chapters – it assigns the grammatical function `OBJ` to the `NP`. On the other hand, there are categories to which no functional annotation is attached (see `P` in the first disjunct of (7.14)) – as in theoretical LFG, such categories bear the default co-head annotation (`^=!` in the XLE notation, ↑=↓ in the LFG notation), which unifies the f-structure of the category to which it is attached with the f-structure of the mother category. The XLE constraint `(^ PRED FN) $c {naprzeciwko temu}` is the counterpart of (↑ PRED FN) $\in_c$ {NAPRZECIWKO TEMU} in the theoretical LFG notation: it states that the form of the predicate inside the value of the `PRED` attribute[5] of the preposition must be a member of the set which contains `naprzeciwko` and `temu` – it must be one of these two forms. This constraint is used only with the forms listed in the second disjunct of (7.14), the one where the nominal precedes the preposition, i.e. in case of NAPRZECIWKO and TEMU – prepositions which are syntactic postpositions. When the ordering is reversed, as in the first disjunct of (7.14), where the preposition precedes the nominal, no such constraint is used.

Finally, there is one important difference between the two disjuncts of (7.14): in the first disjunct, the preposition can be semantic (taking an object) or non-semantic (treated as a co-head), while in the second disjunct the preposition can only be semantic (for discussion, see § 2.3). To understand the difference between semantic and non-semantic prepositions, let us take a look at the template which is used to create the lexical entries of prepositions in XLE, i.e. `PREP` defined in (7.17):

(7.17)
```
PREP(P C) = "preposition (semantic or non-semantic)"
{
"semantic: own PRED, CAT"
@(PTYPE sem)
```

[5] Note that the effect of using a path with PRED FN is different than using just PRED since the latter points to the entire value of PRED attribute (‘RUN<$\boxed{1}$>’ in (i), which is the f-structure representation of the sentence sentence *John runs*), rather than just to the form of the predicate, as in the case of the former (‘RUN’).

(i) $\begin{bmatrix} \text{PRED} & \text{‘RUN}\langle\boxed{1}\rangle\text{’} \\ \text{SUBJ} & \boxed{1}\begin{bmatrix} \text{PRED} & \text{‘JOHN’} \end{bmatrix} \end{bmatrix}$

```
        (^ OBJ)=%PATH
        @(KAT prep)
           {
           "adjunct or semantic OBL"
           (^ PRED)='P<(^ OBJ)>'
           |
           "predicative (locative, has a SUBJ)"
           (^ PRED)='P<(^ SUBJ)(^ OBJ)>'
           @(CHECK-PATH-SET ^ _PREDICATIVE +)
           }
        |
        "non-semantic: co-head (no CAT, noun's PRED)"
        @(PTYPE nosem)
        ^=%PATH
        @(PFORM P)
        }
        "imposes CASE constraints in %PATH"
        @(PREP-CORE P C %PATH).
```

As indicated in comments in (7.17), its first disjunct defines semantic prepositions – such prepositions have their own PRED function (as in (2.43) in § 2.3), which corresponds to the lemma of the preposition (assigned to the P parameter in the template). The nominal accompanying the preposition is treated as its object (OBJ in the value of PRED). When the preposition is used as a predicative complement (as in copular locative constructions), it additionally takes a subject. Semantic prepositions which only take an object are used as adjuncts or semantic obliques. The fact that a preposition is semantic is signalled by the sem value of the PTYPE attribute – its value is set using the PTYPE template defined in (7.18).

```
(7.18)  PTYPE(PT) = "sets PTYPE"
        (^ PTYPE)= PT.
```

By contrast, non-semantic prepositions do not have a PRED feature of their own (see (2.47) in § 2.3), as indicated in the second disjunct of (7.17). Instead, they store the form of the preposition as the value of the PFORM attribute, which is set using PFORM template defined in (7.19).

```
(7.19)  PFORM(PF) = "sets PFORM"
        (^ PFORM)= PF.
```

Additionally, as indicated in (7.17), the value of the PTYPE attribute is set to nosem using an appropriate call to the PTYPE template defined in (7.18).

There is a common part of (7.17) which is used with both disjuncts – the template PREP-CORE which imposes case constraints in relevant paths, namely in the path assigned to the %PATH variable. The PREP-CORE template defined in (7.20) calls another template, CASE-SUBC-PATH defined in (7.21), which checks that the value of the CASE attribute is equal to a given value.

```
(7.20)  PREP-CORE(P C PATH) = "preposition base (no PRED, no CAT)"
        @(CASE-SUBC-PATH PATH C).
```

(7.21)
```
CASE-SUBC-PATH(PATH C) = "checks that case in PATH is equal to C"
(PATH CASE)=c C.
```

Called by `PREP` in (7.17), `PREP-CORE` calls the template `CASE-SUBC-PATH` passing `%PATH` as the value of `PATH` and the relevant case value as `C`. The value of the `%PATH` variable is defined for each disjunct in (7.17): for semantic prepositions defined in the first disjunct, `%PATH` is `(^ OBJ)`, while for non-semantic prepositions `%PATH` is `^`. As a result, with semantic prepositions the relevant case is required from the nominal object of the preposition, while with non-semantic prepositions it is required from the head (the nominal). In the first disjunct of (7.14) the preposition is not annotated (so it is treated as a (co-)head, it passes all its f-structure information to the mother), while the nominal can be a co-head (first disjunct) or an object of the preposition (second disjunct). Under the co-head annotation of the nominal, it can only combine with a non-semantic preposition, which does not have its own `PRED` feature because nominals always have a `PRED` value and unifying two f-structures with `PRED` attributes would lead to a clash (inconsistency, see § 2.2.2) – `PRED` is an instantiated feature, which means that it can only be set once, so even identical values of `PRED` would conflict. By contrast, non-semantic prepositions have no `PRED`, so there is no clash – `PRED` is taken from the nominal, f-structures of the preposition and the nominal are successfully unified, the case requirement of the non-semantic preposition is checked against the value of case introduced by the nominal.

As a result of calling the template `PREP` defined in (7.17), three partial f-structures are created: one for a non-semantic preposition (`nosem` value of the `PTYPE` attribute and `PFORM` attribute hosting the form of the preposition instead of `PRED`; see the f-structure in (2.46)) and two for the semantic preposition variant (`sem` value of `PTYPE`, `PRED` attribute instead of `PFORM`): one with only one argument in `PRED` (`OBJ`; see the f-structure in (2.42)), the other additionally taking a subject (for predicative uses of prepositions).

7.4 Structural case assignment

Structural case assignment is handled by calling an appropriate template in the lexicon, namely in the lexical entry of the verb assigning case.

7.4.1 Subject

As explained in detail in § 3.2.2, depending on the particular verb assigning case, the verb may assign structural case to its subject (typically nominative, but accusative non-agreeing numerals are also possible) or assign no case at all, as in the situation when the subject has a non-canonical realisation, for instance sentential.

Since structural case assignment to subjects is closely related to agreement, it will be discussed at length in the section focusing on the implementation of verbal agreement (see § 7.5).

7.4.2 Object

This section discusses the issue of how structural case is assigned to nominal objects (see § 3.2.3). It does not take unlike category coordination into account (see § 4.5.2) – this issue is discussed in § 8.3.2.3 when describing the process of converting Walenty to LFG constraints.

As explained in § 3.2.3, structural case assignment to objects depends on factors such as part of speech of the predicate assigning case and on the availability of sentential negation in the relevant domain.

The template defined below handles structural case assignment for objects:

(7.22)
```
STRCO-LEX(GF) = "structural case assignment for objects of verbs
and gerunds"
{
"gerunds always require genitive as structural case"
@(CAT-SUBC ger)
@(CASE-SUBC-PATH (^ GF) gen)
|
"heads other than gerunds"
@(CAT-NEQ-PATH ^ ger)
@(STRCASE GF)
}.
```

The template STRCO-LEX defined in (7.22) has only one parameter – it is GF, which is the variable corresponding to the relevant object grammatical function (OBJ or OBL-STR – see the discussion in § 4.2.1). The template ensures that the grammatical function provided when calling this template bears a value of case which is appropriate in the relevant syntactic environment.

As indicated in comments, the first disjunct of STRCO-LEX in (7.22) handles dependents of gerunds. It requires GF to be specified for the genitive case using the template CASE-SUBC-PATH defined in (7.21) when the predicate imposing case requirements is a gerund, which is checked using the CAT-SUBC template defined in (7.23). CAT-SUBC in turn makes a call to templates CAT-SUBC-PATH and CHECK-PATH-SUBC,[6] see (7.24) and (7.25) for their respective definitions.

(7.23)
```
CAT-SUBC(C) = "checks CAT"
@(CAT-SUBC-PATH ^ C).
```

(7.24)
```
CAT-SUBC-PATH(PATH C) = "checks CAT in PATH"
@(CHECK-PATH-SUBC PATH _CAT C).
```

(7.25)
```
CHECK-PATH-SUBC(PATH ATTR VAL) = "checks that the value of some
CHECK attribute is equal to VAL"
(PATH CHECK ATTR)=c VAL.
```

The second disjunct of STRCO-LEX contains calls to two templates: CAT-NEQ-PATH and STRCASE. The former, defined in (7.26), ensures that the category of the head assigning case is not a gerund by calling the template CHECK-PATH-NEQ (see (7.27) for its definition) with appropriate values of parameters.

[6] The CHECK attribute is used in ParGram grammars for storing attributes which are technical rather than linguistically well-motivated – this is the place where non-standard attributes particular to some specific grammar can be placed (a relevant fragment from XLE documentation (http://www2.parc.com/isl/groups/nltt/xle/doc/PargramStarterGrammar/starternotes.html): "CHECK feature is a feature that each grammar can use for grammar internal features that are largely used as well-formedness checks"). The convention is to prefix the names of attributes stored in CHECK with an underscore: _CAT.

```
(7.26)  CAT-NEQ-PATH(PATH C) = "checks that CAT in PATH is not C"
        @(CHECK-PATH-NEQ PATH _CAT C).
```

```
(7.27)  CHECK-PATH-NEQ(PATH ATTR VAL) = "checks that the value of some CHECK
        attribute is not equal to VAL"
        (PATH CHECK ATTR)~= VAL.
```

The template `STRCASE` defined in (7.28), imposes appropriate case requirements on objects of heads which are not gerunds – a disjunction of calls to templates `AFFIRMATIVE` and `NEGATIVE` is used for this purpose: they require appropriate values of structural case according to the availability of negation.

```
(7.28)  STRCASE(GF) = "structural case assignment for objects of verbs"
        {
        @(AFFIRMATIVE GF)
        |
        @(NEGATIVE GF)
        }.
```

Note that these constraints take the phenomenon of non-local genitive of negation into account (see § 3.2.3.2 for discussion) – they are parameterised (they take `GF` as the parameter corresponding to the relevant object grammatical function) near[7] counterparts of case assignment statements defined in (3.68) and (3.69), respectively.

The first disjunct of `STRCASE` in (7.28) is a call to the template `AFFIRMATIVE` defined in (7.29):

```
(7.29)  AFFIRMATIVE(GF) = "no negation at all: accusative"
        ~(({XCOMP|XCOMP-PRED}* ^) NEG)
        @(CASE-SUBC-PATH (^ GF) acc).
```

It handles cases where sentential negation is not available at all – it uses functional uncertainty coupled with an inside-out path, `~(({XCOMP|XCOMP-PRED}* ^) NEG)`, to check that there is no negation in this path, and requires the accusative case using the template `CASE-SUBC-PATH` defined in (7.21).

The second disjunct of `STRCASE` in (7.28) is a call to the template `NEGATIVE` defined in (7.30):

```
(7.30)  NEGATIVE(GF) =
        @(ANYNEG GF) @(NEGTYPE GF).
```

It is dedicated to cases where sentential negation is available in the relevant domain. `NEGATIVE` is defined in (7.30) as a conjunction of two template calls: `ANYNEG` and `NEGTYPE` defined in (7.31) and (7.32), respectively.

```
(7.31)  ANYNEG(GF) = "negation present at some level"
        (({XCOMP|XCOMP-PRED}* ^) NEG)=c +.
```

[7] The difference is that statements called by the template `STRCASE` defined in (7.28) use the disjunctive path `{XCOMP|XCOMP-PRED}` instead of XCOMP.

(7.32)
```
NEGTYPE(GF) =
{ @(LOCNEG GF) | @(NONLOCNEG GF) }.
```

ANYNEG ensures that negation exists in the relevant path using a constraint similar to the one used in AFFIRMATIVE – the difference is that while (7.31) requires negation, (7.29) prohibits it.

NEGTYPE, the second template call in (7.30), requires an appropriate value of structural case from the object depending on whether negation is local to the predicate assigning case or non-local, transferred syntactically from some higher predicate.

The former case, where local negation is available, is handled by the template LOCNEG defined in (7.33). It calls the template NEG-SUBC-PATH (see (7.34) for its definition) to make sure that negation is local to the head assigning case (the ^ path used as the parameter of NEG-SUBC-PATH points to the f-structure of the verb assigning case) and requires the genitive case by calling the template CASE-SUBC-PATH (defined in (7.21)).

(7.33)
```
LOCNEG(GF) = "local negation: obligatory GoN"
@(NEG-SUBC-PATH ^)
@(CASE-SUBC-PATH (^ GF) gen).
```

(7.34)
```
NEG-SUBC-PATH(PATH) = "checks that NEG in PATH is equal to +
(morphological or semantic)"
(PATH NEG)=c +.
```

The template NONLOCNEG defined in (7.35) is dedicated to handling the latter case, where there is no local negation. It calls the template NEG-NOTEXISTS-PATH defined in (7.36), which ensures that there is no local negation, so it must be transferred due to the fact that it is known that sentential negation is available somewhere in the verb chain – this is ensured by the the template ANYNEG defined in (7.31). When negation is non-local to the predicate assigning case, the object, GF, is required to bear the accusative or genitive case ((^ GF CASE) $c {acc gen}).

(7.35)
```
NONLOCNEG(GF) = "no local negation, only transferred: optional GoN"
@(NEG-NOTEXISTS-PATH ^)
(^ GF CASE) $c {acc gen}.
```

(7.36)
```
NEG-NOTEXISTS-PATH(PATH) = "checks that NEG does not exist in PATH
(morphological or semantic)"
~(PATH NEG).
```

The fully expanded version of (7.22) is shown in (7.37):

(7.37)
```
STRCO-LEX(GF) = "structural case assignment for objects of verbs
and gerunds"
{
"gerunds always require genitive as structural case"
(^ CHECK _CAT)=c ger
(^ GF CASE)=c gen
|
"heads other than gerunds"
```

```
        (^ CHECK _CAT)~= ger
            {
            "no negation at all: accusative"
            ~(({XCOMP|XCOMP-PRED}* ^) NEG)
            (^ GF CASE)=c acc
            |
            "negation present at some level"
            (({XCOMP|XCOMP-PRED}* ^) NEG)=c +
               {
               "local negation: obligatory GoN"
               (^ NEG)=c +
               (^ GF CASE)=c gen
               |
               "no local negation, only transferred: optional GoN"
               ~(^ NEG)
               (^ GF CASE) $c {acc gen}
               }
            }
        }.
```

7.5 Verbal agreement

This section shows how verbal agreement is handled in the implemented grammar. It must be noted that there is some variation in agreement as different verb forms display agreement in different categories – past tense verbs agree with their subject in number, person and gender, while present tense forms do not display gender agreement (as explained in § 3.1).

This section demonstrates verbal agreement mechanisms used by `praet` verb forms – since they may be used as past tense forms (*szedł* ‘he walked’) or as a part of analytic future forms (when accompanied by a future form of BYĆ: *będzie szedł* ‘he will walk’), the common template used by `praet` forms is provided in (7.38).

(7.38)
```
PRAET-AUX(N G A) = "past/analytic future verb form (no CAT)"
@(PRAET-NUMGEND N G)
@(PRAET-PERSVOC)
@(PRAET-TNSASP A).
```

Only the first template called by the template `PRAET-AUX` defined in (7.38), `PRAET-NUMGEND`, will be discussed here – the two remaining templates handle person agreement and features such as vocalicity, tense and aspect, which are of relatively little interest here.

The template `PRAET-NUMGEND` defined in (7.39) ensures number and gender agreement between the subject (or its part, see § 7.5.3) and the verb:

(7.39)
```
PRAET-NUMGEND(NUM G)= "number and gender for past/analytic future
verb form"
```

```
{
"default"
NUM=sg
G=n
@(SV-DEFAULT)
|
"agreement"
@(PRAET-AGR NUM G)
}.
```

The first disjunct of (7.39) handles default agreement, see § 7.5.2 for discussion, while the second one is dedicated to fully agreeing verb forms discussed in § 7.5.1. Finally, single conjunct agreement, possible with both agreement types (default/full), is explained in § 7.5.3.

7.5.1 Full agreement

For the sake of simplicity, let us start with the second disjunct of (7.39). It is devoted to cases where the subject triggers full agreement with the verb (see § 3.1.1 for discussion). Unlike in the first disjunct of (7.39), which handles default agreement, there are no constraints restricting the application of this disjunct – it may be used with any values of number (`NUM`) and gender (`G`).

When this disjunct is used, the `PRAET-AGR` template defined in (7.40) is called:

(7.40)
```
PRAET-AGR(N G)= "full number and gender agreement for past/analytic
future verb form (no CAT)"
@(AGR-ALL-PATH (^ SUBJ) ^)
@(SV-AGR-N-CASE-PATH N %L)
@(PATH-VAL (%L GEND) G).
```

(7.40) calls three other templates: `AGR-ALL-PATH` defined in (7.41), `SV-AGR-N-CASE-PATH` shown in (7.43) and `PATH-VAL` (see (7.46)).[8] It is worth noting that `%L`, assigned a value in `AGR-ALL-PATH`, is passed in `PRAET-AGR` as a parameter (or its part) to two different templates: `SV-AGR-N-CASE-PATH` and `PATH-VAL`. Let us now see how all these template calls interact.

When `AGR-ALL-PATH` is called, its first disjunct is used for full agreement (the second one is dedicated to single conjunct agreement, see § 7.5.3 for discussion): it assigns its first parameter, `CONTROLLER`, to the variable `%L` – the result is the specification in (7.42).

(7.41)
```
AGR-ALL-PATH(CONTROLLER CONTROLLEE)= "agreement by resolution and CCA"
{
CONTROLLER=%L
|
@(AGR-CCA-PATH CONTROLLER CONTROLLEE)
}.
```

[8] One may wonder why one template is used to set number and case (`SV-AGR-N-CASE-PATH` defined in (7.43)), while another template is used to set gender (`PATH-VAL`, see (7.46)) – this is because most finite forms (with the exception of the *-no/to* impersonal) impose constraints on the number and case of the verb's subject, while only one form imposes a constraint on its gender (the past tense form).

```
(7.42)   (^ SUBJ)=%L
```

(7.42) is crucial to understand the effect of calling `SV-AGR-N-CASE-PATH` in (7.40): the first parameter passed to `SV-AGR-N-CASE-PATH` defined in (7.43) is `N`, which hosts the value of number, the second parameter is `%L` defined in (7.42). Resulting template calls are shown in (7.44):

```
(7.43)   SV-AGR-N-CASE-PATH(N PATH)= "nominative case and appropriate number"
         @(CASE-SUBC-PATH PATH nom)
         @(PATH-VAL (PATH NUM) N).

(7.44)   @(CASE-SUBC-PATH %L nom)
         @(PATH-VAL (%L NUM) N)
```

Template `CASE-SUBC-PATH`, defined in (7.21), is repeated below as (7.45), while the definition of template `PATH-VAL` is provided in (7.46). According to these definitions and the variable assignment in (7.42), (7.47) is the result of respective template calls in (7.44):

```
(7.45)   CASE-SUBC-PATH(PATH C) = "checks that case in PATH is equal to C"
         (PATH CASE)=c C.

(7.46)   PATH-VAL(PATH VAL) = "setting a value for a particular path"
         PATH=VAL.

(7.47)   (^ SUBJ CASE)=c nom
         (^ SUBJ NUM)= N
```

As a result, under full agreement the template `SV-AGR-N-CASE-PATH` requires the subject of the verb to be marked for the nominative case and sets the value of its number to `N` (as specified in the call to the `PRAET-AGR` template).

Finally, `PRAET-AGR` makes a call to the template `PATH-VAL` (see (7.46) for its definition) to ensure proper gender agreement (again, according to (7.42) `%L` is replaced with `(^ SUBJ)`):

```
(7.48)   @(PATH-VAL (^ SUBJ GEND) G)
```

(7.49) is the result of the template call in (7.48):

```
(7.49)   (^ SUBJ GEND)= G
```

(7.49) sets the value of gender of the subject to `G` (used when making a call to the `PRAET-AGR` template defined in (7.40)).

7.5.2 Default agreement

The first disjunct of `PRAET-NUMGEND` template defined in (7.39) contains a call to the template `SV-DEFAULT`, devoted to handling default agreement (see § 3.1.2 for discussion), defined in (7.50).

Since only certain verb forms may be used with default agreement, this template may only be called when the verb is specified for singular number and neuter gender (as specified by relevant parameter constraints in (7.39): `NUM=sg` and `G=n`).[9]

[9] A constraint on the value of person (`PERS`) is not necessary since `praet` forms do not bear this attribute.

Though the template SV-DEFAULT defined in (7.50) consists of three disjuncts which handle particular cases of default agreement, only the first one (containing four embedded disjuncts) will be discussed in this section.

(7.50)
```
SV-DEFAULT= "default agreement"
{
@(AGR-ALL-PATH (^ SUBJ) ^)
   {
   "non-agreeing numerals"
   @(ACM-SUBC-PATH %L rec)
   @(CASE-SUBC-PATH %L acc)
   |
   "clausal subject (with a complementiser)"
   @(COMP-FORM-EXISTS-PATH %L)
   |
   "clausal subject (interrogative)"
   @(CL-TYPE-SUBC-PATH %L int)
   |
   "infinitival subject"
   @(CAT-SUBC-PATH %L inf)
   }
|
"SIĘ impersonals"
@(IMPERSONAL-SUBC)
|
"for subjectless predicates"
~(^ SUBJ)
}.
```

In the first disjunct of (7.50) the template AGR-ALL-PATH is called with the same parameters as in PRAET-AGR defined in (7.40) – assuming that the single conjunct agreement disjunct of (7.41) is not taken into consideration for the time being (but see § 7.5.3), the effect of this template call is assigning the path (^ SUBJ) to %L variable, as in (7.42). The following four embedded disjuncts impose further constraints on the path assigned to %L. The first one is discussed in § 7.5.2.1, the second and the third in § 7.5.2.2, the fourth in § 7.5.2.3.

7.5.2.1 Non-agreeing numerals

The first embedded disjunct of SV-DEFAULT in (7.50) handles non-agreeing numerals, which trigger default agreement when they serve as the subject (see § 3.1.2.1 for discussion). It uses two templates: CASE-SUBC-PATH defined in (7.21) (repeated as (7.45)) and ACM-SUBC-PATH defined in (7.51) below:

(7.51)
```
ACM-SUBC-PATH(PATH A) = "checks that ACM in PATH is equal to A"
(PATH ACM)=c A.
```

When, according to (7.42), `%L` is substituted with `(^ SUBJ)`, the following constraints are introduced as a result of `@(ACM-SUBC-PATH %L rec)` and `@(CASE-SUBC-PATH %L acc)` template calls, respectively:

(7.52) `(^ SUBJ ACM)=c rec`

(7.53) `(^ SUBJ CASE)=c acc`

(7.53) ensures that the subject is marked for the accusative case, while (7.52) ensures that the subject is a non-agreeing numeral by requiring the `rec` value of its `ACM` attribute, which corresponds to accommodability (see § 3.1.2.1 for discussion).

7.5.2.2 Clausal subjects

The second embedded disjunct of `SV-DEFAULT` in (7.50) is dedicated to sentential subjects, which also trigger default agreement in Polish (see § 3.1.2.2 for discussion). In this case, the template `COMP-FORM-EXISTS-PATH` is used to ensure that the subject is a complementiser phrase:

(7.54)
```
COMP-FORM-EXISTS-PATH(PATH) = "checks that COMP-FORM exists in PATH"
(PATH COMP-FORM).
```

(7.54) is called in (7.50) with `%L` as the value of its only parameter: `@(COMP-FORM-EXISTS-PATH %L)`. When the variable `%L` is substituted with `(^ SUBJ)`, as in (7.42), the following constraint is introduced:

(7.55) `(^ SUBJ COMP-FORM)`

As a result, the subject is required to have the `COMP-FORM` attribute in its f-structure ((7.55) is an existential constraint) – this attribute is used to indicate the possible complementiser forms.

Similarly, the third embedded disjunct of `SV-DEFAULT` is used to allow for subjects realised as interrogative clauses. It makes a call to the template `CL-TYPE-SUBC-PATH`:

(7.56)
```
CL-TYPE-SUBC-PATH(PATH CL) = "checks that CLAUSE-TYPE in PATH is equal
to CL"
(PATH CLAUSE-TYPE)=c CL.
```

As a result, the following constraint is introduced (as mentioned above, the variable `%L` is substituted with `(^ SUBJ)`):

(7.57) `(^ SUBJ CLAUSE-TYPE)=c int.`

This constraint ensures that the subject contains the attribute `CLAUSE-TYPE` whose value is `int` – it must be an interrogative clause.

7.5.2.3 Infinitival subjects

The fourth embedded disjunct of `SV-DEFAULT` in (7.50) is devoted to infinitival subjects, which are another element which triggers default agreement in Polish – an example is provided below:

(7.58) Mówić prawdę było twoim obowiązkiem.
speak.INF truth.ACC was your.INST duty.INST
'To speak the truth was your duty.' (Dziwirek 1990, p. 154, ex. (17a))

Using the `CAT-SUBC-PATH` template defined in (7.24) (which in turn calls the template `CHECK-PATH-SUBC` defined in (7.25)), it checks that the subject is infinitive by checking the value of the `_CAT` attribute, which corresponds to its morphosyntactic category. As a result, the following constraint is introduced (as explained above, `%L` used in the template call was substituted with `(^ SUBJ)`):

(7.59)
```
(^ SUBJ CHECK _CAT)=c inf
```

7.5.3 Single conjunct agreement

After simple agreement mechanisms have been introduced, let us proceed to single conjunct agreement (see § 3.1.3 for discussion and examples). While mechanisms discussed in § 7.5.1–§ 7.5.2 handled non-coordinate structures or coordinate structures with resolved agreement features, single conjunct agreement is used only when the subject is a coordinate structure. More importantly, it does not use resolved agreement features of the coordinate subject, but instead it chooses one of the edge conjuncts and agrees with it – this is the agreement controller.

Under single conjunct agreement the agreement controller is typically the conjunct which is closest to the verb: the rightmost conjunct if the subject precedes the verb (as in (3.32) repeated as (7.60)) or the leftmost conjunct if the verb precedes the subject (see (3.31) repeated as (7.61)).

(7.60) Pan Mirosław i czternastu ludzi pracowało
Mr Mirosław.NOM.SG.M1 and fourteen.ACC.PL.M1 man.GEN.PL.M1 worked.3.SG.N
dzień i noc
day and night
'Mr Mirosław and fourteen men worked night and day.' (NKJP)

(7.61) panowała harmonia i spokój
prevail.3.SG harmony.NOM.SG.F and peace.NOM.SG.M3
'Harmony and peace prevailed.' (NKJP)

The template `AGR-CCA-PATH`, defined in (7.62), uses a mechanism which implements the choice of the closest conjunct as the agreement controller along the lines of the basic analysis presented in § 3.1.3:

(7.62)
```
AGR-CCA-PATH(CONTROLLER CONTROLLEE)= "closest conjunct agreement"
(CONTROLLER $)=%L
   {
   CONTROLLEE <h CONTROLLER
   "CCA: leftmost"
   ~[(CONTROLLER $) <h %L]
   |
   CONTROLLER <h CONTROLLEE
```

```
        "CCA: rightmost"
        ~[%L <h (CONTROLLER $)]
        }.
```

According to its definition provided in (7.40), the template PRAET-AGR handling full agreement calls the template AGR-ALL-PATH, passing the following values to its relevant parameters (see (7.41) for its definition): (^ SUBJ), the first parameter, as the CONTROLLER and ^, the second parameter, as the CONTROLLEE.

The template AGR-ALL-PATH has two disjuncts – the second one is used for single conjunct agreement. Both parameters of AGR-ALL-PATH are passed to respective parameters of AGR-CCA-PATH defined in (7.62) above.

After parameters used in the definition of AGR-CCA-PATH are replaced with relevant values used in the macro call ((^ SUBJ) as the CONTROLLER and ^ as the CONTROLLEE, as explained above), the following constraint results:

```
(7.63)  (^ SUBJ $)=%L
        {
        ^ <h (^ SUBJ)
        "CCA: leftmost"
        ~[(^ SUBJ $) <h %L]
        |
        (^ SUBJ) <h ^
        "CCA: rightmost"
        ~[%L <h (^ SUBJ $)]
        }.
```

The notation (^ SUBJ $) has the effect of choosing an element of the set corresponding to the f-structure of the subject – the assumption is that the subject is a coordinate structure and one of its conjuncts is picked in this way. After an element of the set was selected in this way, it is assigned to %L variable, which corresponds to the agreement controller.

Finally, there are two disjuncts in (7.63) – they ensure the proper choice of the agreement controller depending on the word order, namely depending on how the verb and the subject are placed with respect to one another.

The first disjunct of (7.63) handles situations where the verb precedes the subject (see (7.61)). The constraint ^ <h (^ SUBJ) ensures the ordering in which the subject follows the verb – it uses head precedence operator <h, which ensures that the element to its left precedes the element to its right at c-structure.[10] Using the same formal device, namely head precedence, the constraint ~[(^ SUBJ $) <h %L] ensures that there is no conjunct that would precede the conjunct assigned to the variable %L – this constraint ensures that the conjunct assigned to %L variable is the left edge conjunct. Together, these constraints pick the leftmost conjunct when the verb precedes the subject – this is the conjunct which is closest to the verb (as in (7.61)).

[10] The head precedence operator <h is defined as follows in XLE documentation (http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N4.2.9): "f1 <h f2 is true if and only if f1 and f2 have heads and the head of f1 precedes the head of f2 in the c-structure".

The second disjunct of (7.63) handles the opposite case where the subject precedes the verb (`(^ SUBJ) <h ^`) – in this situation the conjunct closest to the verb is the rightmost conjunct of the subject (see (7.60)). This conjunct is picked using the statement `~[%L <h (^ SUBJ $)]`, which ensures that there is no conjunct right to `%L` variable, which hosts the conjunct acting as the agreement controller.

As a result, instead of using the variable assignment in (7.42), repeated as (7.64) below, where `%L` refers to `(^ SUBJ)`, the entire f-structure of the subject (non-coordinate or coordinate with resolved agreement features), the variable `%L` points to the relevant conjunct of the f-structure of the subject chosen by the template `AGR-CCA-PATH`, as shown in (7.65):

(7.64) `(^ SUBJ)=%L`

(7.65) `(^ SUBJ $)=%L`

After the appropriate conjunct (assigned to the `%L` variable) is chosen using the template `AGR-CCA-PATH` defined in (7.62), it is used by the relevant agreement templates called by `PRAET-AGR` defined in (7.40). The effect of calling templates `@(SV-AGR-N-CASE-PATH N %L)` and `@(PATH-VAL (%L GEND) G)` was discussed in previous sections (see § 7.5.1 and § 7.5.2 for full and default agreement, respectively); the only difference is that the assignment of the `%L` variable shown in (7.64) (the entire subject) is used there instead of (7.65) used here, where it corresponds to the relevant conjunct of the subject. When the assignment in (7.65) is used with the relevant agreement templates (full/default agreement), the constraints defined in these templates must be satisfied by the closest conjunct.

While this section focused on showing how the template `AGR-CCA-PATH` (see (7.62)) interacts with the template `PRAET-AGR` defined in (7.40), which is dedicated to handling full agreement, single conjunct agreement may also be used with default agreement – the template `SV-DEFAULT` defined in (7.50) makes a call to the template `AGR-ALL-PATH`, which in turn, according to its definition provided in (7.41), calls the template `AGR-CCA-PATH`.

7.6 Summary

This chapter introduced the mapping between theoretical LFG notation and the notation used in grammars implemented in XLE (§ 7.2). It also presented an example of how DCG rules were rewritten into XLE/LFG (§ 7.3). More importantly, this chapter provided a detailed discussion of how two key mechanisms used by the grammar were implemented: structural case assignment to the object, see § 7.4.2, as well as to the subject. The latter was discussed in § 7.5 as a part of verbal agreement mechanisms, which include full agreement, § 7.5.1, default agreement, § 7.5.2, and closest conjunct agreement, § 7.5.3.

Chapter 8

Valence information: converting Walenty to LFG

8.1 Introduction[1,2]

This chapter is concerned with Walenty (Przepiórkowski *et al.* 2014b,c,a), a new valence dictionary of Polish, which is freely available on an open source licence from `http://zil.ipipan.waw.pl/Walenty`, and with the way its valence information can be converted to XLE/LFG constraints. It is presented here due to the fact that the information about the occurrence of unlike category coordination used elsewhere in this work is taken from Walenty.

Walenty has been developed since 2012, spanning 3 projects, and it is currently the largest valence dictionary of Polish – the version of September 2014 described in this chapter contains 50442 schemata (48095 verbs and 2347 other, which include nouns and adjectives) for a total of 11360 lemmata (10967 verbs and 393 other). It must be highlighted that Walenty is under development, so it is constantly growing.

Walenty provides an explicit account of a range of phenomena. These include coordination of unlike categories, structural case assignment, control, passivisation and lexicalised requirements (including multiword expressions, together with their modification patterns).

Entries in Walenty are created on the basis of attested examples taken from NKJP or found on the Internet (if relevant examples could not be found in NKJP). Walenty uses its own, independent formalism, which can be converted so as to be used with different grammar formalisms.

Some information about the formalism used in Walenty is provided in § 8.2, while § 8.3 describes the process of converting entries to constraints used by the Polish LFG grammar. Similarly to previous chapters devoted to implementation, this chapter takes a more technical approach than theoretical chapters in previous parts of this work.

[1] This chapter is based on the following papers, which provide a description of the formalism used in Walenty: Przepiórkowski *et al.* 2014b in Polish, describing the early version of the formalism (of September 2013), Przepiórkowski *et al.* 2014c in English, describing the version of March 2014, and Przepiórkowski *et al.* 2014a in English, describing the extended formalism used to account for lexicalised arguments.

[2] The author of this work has taken part in discussions since the inception of Walenty, as well as in its development and quality control (especially when performing conversion). While the development of Walenty is joint work, conversion of Walenty is the author's own work associated with the development of Polish LFG.

8.2 Formalism used in Walenty

Valence requirements of verbs are described in Walenty using its own, dedicated formalism (see Przepiórkowski *et al.* 2014b,c for a detailed description). The aim of this section is to provide a brief description of its main characteristics.

Let us start with two simple subentries from Walenty as an example:

(8.1) `adaptować się: _: : _: subj{np(str)} + {prepnp(do,gen)}`

(8.2) `dedykować: _: : _: subj{np(str)} + obj{np(str)} + {np(dat); prepnp(dla,gen)}`

The schemata provided in (8.1) and (8.2) are selected from the entries for the verbs ADAPTOWAĆ SIĘ 'adapt' and DEDYKOWAĆ 'dedicate', respectively.

Each subentry in Walenty consists of 5 parts (fields), separated by colons (`:`):

- lemma (containing `się` when inherent, rather than reflexive[3] SIĘ is involved, as in (8.1));
- negativity:
  - `aff` when the schema is valid only without negation,
  - `neg` when the schema is valid only when negation is present,
  - `_` (underscore) when there are no constraints with respect to this feature;
- predicativeness:
  - `pred` if the item must be in a predicative position (or contained in it);
  - empty if not predicative or not applicable (as with verbs, see (8.1)–(8.2));
- aspect:
  - `imperf` when the schema is valid only with imperfective aspect,
  - `perf` when the schema is valid only with perfective aspect,
  - `_` (underscore) when the schema is valid with any aspect (for biaspectual schemata);
- valence schema for the relevant verb (see § 8.2.2 for a detailed description).

8.2.1 Grammatical function labels

Let us proceed to the discussion of the last field listed above, namely the valence schema. Some syntactic positions are labelled in Walenty – some sets are preceded by one or more labels, as in (8.3), the schema taken from the entry of DEDYKOWAĆ 'dedicate' in (8.2):

(8.3) `subj{np(str)} + obj{np(str)} + {np(dat); prepnp(dla,gen)}`

Sets are separated by `+` (see § 8.2.2). The first set in (8.3) is labelled as `subj` – it is the subject; one of the criteria for determining subjecthood, perhaps the most decisive one, is agreement (see

[3] Reflexive SIĘ is represented as `refl` in the relevant argument set.

§ 3.1 for discussion of subject-verb agreement in Polish). The second set is labelled as `obj` – it corresponds to the argument which becomes the subject under passive voice.[4]

Note that there are no constraints on which categories may be marked as `obj`, it is also not influenced by case marking – the verb MANIPULOWAĆ 'manipulate' is an example of a predicate taking a nominal object marked for the instrumental case, as shown in (8.5), while (8.6) demonstrates that this argument can become the subject under passive voice; its schema is provided in (8.4):

(8.4) `subj{np(str)} + obj{np(inst)}`

(8.5) Manipulowała mną i swoim późniejszym mężczyzną.
manipulated I.INST and SELF.INST later.INST man.INST
'She manipulated me and her later man.' (NKJP)

(8.6) Młodzi ludzie byli manipulowani przez starsze osoby.
young.NOM people.NOM were manipulated.NOM by elder persons
'Young people are manipulated by elder people.' (NKJP)

8.2.2 Syntactic positions as sets

As mentioned in § 8.2.1, syntactic positions are modelled in Walenty as sets, which is one of its key features. The following schema in (8.7) was used in the entry of the verb ADAPTOWAĆ SIĘ 'adapt' in (8.1), while (8.8) is the schema repeated from (8.3), which was taken from the entry of DEDYKOWAĆ 'dedicate' in (8.2):

(8.7) `subj{np(str)} + {prepnp(do,gen)}`

(8.8) `subj{np(str)} + obj{np(str)} + {np(dat); prepnp(dla,gen)}`

Walenty describes the valence of verbs using the notion of a syntactic position (Szupryczyńska 1996) – each argument of the relevant verb corresponds to one syntactic position. Syntactic positions are separated by `+` in Walenty – there are 2 positions in (8.7) and 3 positions in (8.8).

Since each syntactic position can have a number of categorial realisations, syntactic positions are modelled as sets – realisations are elements of the set corresponding to a given syntactic position. The following notational convention was adopted in Walenty: sets are represented in curly brackets (`{...}`), their elements are separated by semi-colons (`;`).

Modelling syntactic positions as sets makes it possible to provide an explicit account of coordination possibilities for particular arguments. If a given syntactic position has more than one realisation, this means that it can be realised by any of these realisations on its own or by a coordination of these realisations. This is the case in the schema in (8.8), where the last argument is a two-element set (`{np(dat); prepnp(dla,gen)}`), which contains a nominal phrase marked for the dative case (`np(dat)`) and a prepositional phrase featuring the preposition `dla`, which requires a nominal phrase in the genitive case (`prepnp(dla,gen)`). The example in (8.9) shows that these realisations can be coordinated, while the remaining examples, (8.10) and (8.11), demonstrate that particular realisations can also be used on their own:

[4] See § 4.2.3 for discussion of problems with such a definition.

(8.9) Gola dedykuję [dla rodziców] i [sympatii Iwonie].
goal dedicate to parents.GEN and girlfriend.DAT Iwona.DAT
'I dedicate this goal to my parents and my girlfriend Iwona.' (NKJP)

(8.10) Gola dedykuję dla rodziców.

(8.11) Gola dedykuję sympatii Iwonie.

On the other hand, if some realisations cannot be coordinated, it means that they are not realisations of the same argument and, as a result, they are placed in separate schemata. If a given position can be realised in only one way, a singleton set is used to represent this position (`{prepnp(do,gen)}` in (8.7)). Sets containing more elements are used only when more than one realisation is possible (as in (8.9)), according to the coordination test described above.

8.2.3 Unlike category coordination

First, since Walenty treats syntactic positions as sets and therefore accounts for coordination possibilities for each argument, it provides an explicit account of unlike category coordination. (8.12) contains one of the schemata of the verb CIEKAWIĆ 'make curious':

(8.12) `subj{np(str); cp(int); ncp(str,int); ncp(str,że)} + {np(str)}`

The schema in (8.12) consists of 2 arguments: the second one, `{np(str)}`, is a set containing only one realisation – a nominal complement marked for structural case, which cannot passivise (if it could, this position would be labelled as `obj`, as explained in § 8.2.1). The first argument is the subject, which is a four-element set – this argument can be realised by any combination of categories contained in this set: a nominal phrase marked for structural case (`np(str)`), an interrogative clause (`cp(int)`), an interrogative clause with a correlative pronoun (`ncp(str,int)`) or a clause featuring a `że`-type complementiser with a correlative pronoun (`ncp(str,że`). The example provided in (8.13) demonstrates the coordination of an interrogative clause (*ile zarabiają inkasenci*) with an interrogative clause with a correlative pronoun (*to, kto ich powołuje*) in the subject position:

(8.13) Ciekawi mnie także, [ile zarabiają inkasenci] oraz [to, kto ich
interests me also how much earn payment collectors and that.ACC who them
powołuje].
appoints
'I am also interested in how much payment collectors earn and who appoints them.'
(NKJP)

8.2.4 Arguments marked for structural case

Walenty also provides an explicit account of structural case. Unlike lexical case, which is stable in the sense that it is independent of the syntactic context, structural case is understood here as a case which may take different values depending on the syntactic environment (see § 3.2 for discussion) – such arguments have the `str` value of case. The information supplied by the valence dictionary is to be processed by the grammar so as to assign an appropriate case in the given context.

When a subject is marked for structural case, its case marking may be realised in three ways.[5] The first possibility is the nominative case, the most prototypical value – it is appropriate for subjects of finite verb forms which are not non-agreeing numerals. The second value is the accusative case – it is possible when the subject of a finite verb form is a non-agreeing numeral. Finally, the third possible value is the genitive case – this is the case with the subject of gerunds. See § 3.2.2 for more discussion and examples.

When an object (passivisable or not) bears structural case, there are two possible values: accusative or genitive, depending on the availability of sentential negation and part of speech of the head assigning case. Gerunds require the genitive case from their structural objects regardless of negation. With other verbal forms, genitive is required when the verb assigning structural case is in the scope of sentential negation; this phenomenon is known as genitive of negation (GoN). If negation is local, GoN is obligatory, while with non-local negation (present higher in the verb chain), GoN is optional. As a result, accusative is required as structural case when negation is not present at all and it is possible when it is non-local to the predicate assigning structural case. See § 3.2.3 for detailed discussion and examples.

8.2.5 Control relations

It is also possible to describe control relations in Walenty. Let us start with establishing control between one of the arguments of the control verb and the subject of the subcategorised infinitive: (8.14) is one of the schemata for the verb KAZAĆ ‘order’, while the schema in (8.15) is taken from one of the entries for the verb OBIECAĆ ‘promise’.

(8.14) `subj{np(str)} + controller{np(dat)} + controllee{cp(żeby); infp(_)}`

(8.15) `subj,controller{np(str)} + controllee{infp(_)} + {np(dat)}`

In (8.14), the dative argument is labelled as `controller`, while the argument containing the infinitival phrase is labelled as `controllee`[6] – this means that the dative argument of KAZAĆ is at the same time the subject of the infinitival complement of KAZAĆ; this is an instance of object control, see (8.16):

(8.16) Dowódca kazał nam wszystkim uciekać.
commander.NOM ordered us.DAT all.DAT escape.INF
‘The commanding officer ordered us all to run away’. (NKJP)

By constrast, in (8.15) it is the subject of OBIECAĆ which is the controller of the infinitival complement marked as the controllee – subject control is involved here, as illustrated in (8.17):

(8.17) obiecał nam zbadać tę sprawę
promised we.DAT investigate.INF this.ACC matter.ACC

[5] Note that this does not apply to the subject of adjectival participles (determined by agreement) and infinitives (determined by control).

[6] In the version of September 2014 control relations are assigned to entire positions rather than individual realisations. This does not mean, however, that `cp(żeby)`, the sentential clause with ŻEBY-type complementiser, is controlled in (8.14).

‘He promised us to investigate this issue.’ (NKJP)

It is perhaps worth mentioning that the controller does not have to be lexical to take part in control – the controller may be implicit. Let us see the schema taken from the entry of the object control verb ZAPROPONOWAĆ ‘propose’:

(8.18) `subj{np(str)} + controller{np(dat)} + controllee{infp(_)}`

The dative argument of ZAPROPONOWAĆ is identified as the controller of its infinitival complement. The example in (8.19) uses an implicit controller:

(8.19) Potem zaproponowano urządzić koniowi pogrzeb
later offer.IMPERS organise.INF horse.DAT funeral.ACC
‘Later they were offered to organise a funeral for the horse.’ (NKJP)

Finally, extending somewhat the usual understanding of the term *control*, `controller–controllee` pairs are used for identifying the controller of predicative complements (such as adjectives and nominals, possibly embedded in a prepositional phrase). Control is used here not only for the purposes of semantics, but also for the sake of agreement. See (8.20)–(8.21) for two schemata taken from among the entries of the verb UWAŻAĆ ‘consider’:

(8.20) `subj{np(str)} + obj,controller{np(str); ncp(str,int); ncp(str,że)}`
`+ controllee{prepadjp(za,acc)}`

(8.21) `subj{np(str)} + obj,controller{np(str)} + controllee{prepnp(za,acc)}`

When the control relation is established, the element labelled as `controllee` may inherit agreement features such as number and gender from the element identified as the `controller` – whether the controlled item inherits agreement features from the controller depends on the category of the controllee.

Agreement is obligatory when the controlled element (marked as `controllee`) is an adjectival phrase, even when it is embedded in a prepositional phrase (`{prepadjp(za,acc)}` in (8.20)) – see the examples in (8.22)–(8.24). Note, however, that such agreement does not apply to case: this is shown in (8.24), where the controller is marked for the structural genitive case (because negation is present) but the predicative adjective bears the accusative case (assigned by the preposition).

(8.22) Też uważam nazwę za dobrą
too consider name.ACC.F.SG for good.ACC.F.SG
‘I also consider this name (to be) good.’ (NKJP)

(8.23) Uważam ją za godną zaufania.
consider she.ACC.F.SG for worth.ACC.F.SG trust.GEN
‘I consider her to be trustworthy.’

(8.24) Nie uważam jej za godną zaufania.
NEG consider she.GEN.F.SG for worth.ACC.F.SG trust.GEN
‘I do not consider her to be trustworthy.’

However, such agreement is not obligatory (though possible) when the controllee is a noun phrase (also possibly embedded in a prepositional phrase: `{prepnp(za,acc)}` with the `controllee` label in (8.21)); see (8.25), where the controller (*okresie*) is marked for the masculine gender while the controlled noun phrase (*miłość*, embedded in a prepositional phrase) is feminine.

(8.25) Bawić lubiła się bardzo, nie na tyle jednak, aby w pierwszym okresie swego
play liked REFL very NEG enough though that in first period SELF
stosunku.M3 do męża, uważanym przez nią za miłość.F, nie pragnęła dziecka.
attitude to husband considered by her for love NEG want baby
'She liked a lot to have fun, though not so much as to not want a baby in the first period of her attitude to her husband, considered by her to be love.' (NKJP)

8.2.6 Adverbial-like complements classified according to semantic type

There is a class of arguments whose description takes semantics into consideration – this is the `xp` category. The valence dictionary used by Świgra (SDPV, Świdziński 1994, 1998) employs the category `advp`, whose prototypical realisation is an adverbial phrase, though it may also be realised as a prepositional phrase, a nominal phrase or a clause – such constraints are not imposed there, however, which means that any realisation of `advp` is possible, regardless of the lexicalised requirements of the predicate subcategorising for an argument of this category.

Walenty takes a step forward by making it possible to specify the semantics of such arguments – the `xp` category takes a parameter which corresponds to its semantic class – it may take one of the following 7 values:[7]

- `locat`: place, e.g. *być* ***w sklepie*** 'to be in a shop', *znajdować się* ***przy drzwiach*** 'to be at the door',
- `abl`: starting point (source), e.g. *wyprowadzić* ***ze strefy wojny*** 'to take (somebody) out of the area of war',
- `adl`: ending point (destination), e.g. *przywieźć* ***do Zabrza*** 'to take (somebody) to Zabrze',
- `perl`: route, e.g. *biec* ***przez wieś*** 'to run through the village',
- `temp`: point in time, e.g. *mieć miejsce* ***wczoraj*** 'to take place yesterday',
- `dur`: duration, e.g. *trwać* ***dwie godziny*** 'to last two hours',
- `mod`: manner, e.g. *traktować* ***źle*** 'to treat (somebody) badly', *zachowywać się* ***jak dziecko*** 'to behave like a child'.

The semantic class of `xp` restricts the range of possible realisations of this category – each semantic type has a defined list (intended to be near-exhaustive) of its possible categorial realisations. Realisations represent various phrases: nominal, prepositional, clausal, adverbial, lexicalised or not. Let us see the realisation lists for `xp(abl)` and `advp(abl)`, one of its realisations.

[7] The list with examples provided below is taken (in a translated version) from Przepiórkowski *et al.* 2014b.

Relevant lists of realisations are provided in the form of rewrite rules: in (8.26) the category `xp(abl)` is the left-hand side of the rule and it has 14 rewrite possibilities on the right-hand side. Each right-hand side element consists of two parts: the realisation and its assessment. The assessment is put in square brackets and it takes one of the following values:

- `pewna` 'certain',
- `potoczna` 'colloquial',
- `archaiczna` 'archaic',
- `wątpliwa` 'dubious'.

One of the categories to which `xp(abl)` may rewrite is `advp(abl)`, whose list of realisations is provided in (8.27) – it rewrites to 11 different adverbs, listed by their lemmata, together with the assessment of each realisation as described above. The remaining realisations of `xp(abl)` defined in (8.26) include a clause featuring `skąd` 'where from' (`cp(skąd)`),[8] a prepositional-adjectival phrase (`prepadjp(z,gen)`) and 11 various prepositional phrases.

(8.26)

```
xp(abl)-->
    advp(abl) [pewna]
    cp(skąd) [pewna]
    prepadjp(z,gen) [pewna]
    prepnp(od,gen) [pewna]
    prepnp(spod,gen) [pewna]
    prepnp(spomiędzy,gen) [pewna]
    prepnp(sponad,gen) [pewna]
    prepnp(spopod,gen) [pewna]
    prepnp(spośród,gen) [pewna]
    prepnp(spoza,gen) [pewna]
    prepnp(sprzed,gen) [pewna]
    prepnp(z,gen) [pewna]
    prepnp(znad,gen) [pewna]
    prepnp(zza,gen) [pewna]
```

(8.27)

```
advp(abl)-->
    skąd [pewna]
    skądkolwiek [pewna]
    skądś [pewna]
    skądże [pewna]
    stamtąd [pewna]
    stąd [pewna]
    zewsząd [pewna]
    znikąd [pewna]
    skądinąd [archaiczna]
    skądciś [potoczna]
    skądsiś [potoczna]
```

Together with the realisations of `advp(abl)` defined in (8.27), `xp(abl)` has 24 realisations (14 realisations in (8.26) minus `advp(abl)`, plus 11 realisations of `advp(abl)` defined in (8.27)). Note that each realisation of `xp(abl)` discussed above has the appropriate semantics – ablative.

8.2.7 Phraseological units

Walenty takes phraseological units into account, making it possible to specify explicitly that a given argument must be realised by some particular word, possibly requiring a certain value of number and some particular modification pattern. In this way, it is possible to provide an account of multiword expressions which takes their flexibility into account, treating them as units which are not excluded from syntax.

[8] An example featuring `cp(skąd)`:

(i) Pochodzi, skąd pochodzą wielcy poeci.
comes from where come from great poets
'(S)he comes from (the place) where great poets come from.'

This section describes the treatment of lexicalised units in Walenty as of September 2014. At the time of writing serious changes have been introduced to the formalism (see Przepiórkowski *et al.* 2014a) – the aim of these changes is to make the formalism more powerful, so that it can better describe such units and their requirements.[9] This section does not cover these changes.

There are three major types of multiword expressions (MWEs) in Walenty: fixed expressions, lexicalised phrases and complex prepositions. The following subsections discuss the first two types[10] and their properties, starting with the former – fixed expressions.

8.2.7.1 Fixed expressions

There exist phraseological items which cannot be modified in any way:

(8.28) Bije ich na (*bardzo) kwaśne jabłko/*jabłka.
beats them for very sour apple.SG/PL
'He beats them to a pulp.' (literally: 'He beats them into a sour apple.')

As shown in (8.28), *na kwaśne jabłko* cannot be modified using intensifiers, nor is it possible to change the grammatical number of the nominal used in this expression.

In order to describe such items, Walenty uses the category `fixed`, which takes a single argument – the exact string which is expected in a given position, as in (8.29):

(8.29) `subj{np(str)} + obj{np(str)} + {fixed('na kwaśne jabłko')}`

In this schema for the verb BIĆ 'beat', the third argument is realised by a fixed phrase – the set contains the category `fixed`, which takes the string `na kwaśne jabłko` as a value (strings are enclosed in single quotes). As mentioned above, such an argument cannot be modified in any way – it must appear exactly as it is; it must not undergo any internal or external changes.

8.2.7.2 Lexicalised phrases

The second type of phraseological items are phrases which, unlike fixed expressions discussed in § 8.2.7.1, may undergo changes – the range of possible changes is specified explicitly in Walenty.

In the September 2014 version of Walenty, there are two kinds of lexicalised phrases: nominal and prepositional. Plain, non-lexicalised nominal phrases (`np`) are only specified for case, as schematically shown in (8.30), while plain prepositional phrases (`prepnp`) additionally specify the required form of the preposition involved, as in (8.31).

(8.30) `np(case)`

(8.31) `prepnp(preposition,case)`

[9] Przepiórkowski *et al.* 2014a provides a description of the extended formalism for handling lexicalised arguments in Walenty. It makes it possible to precisely describe lexicalised arguments – instead of having predefined modification patterns, it is possible to explicitly specify the dependents of a given argument. Furthermore, such specifications may be embedded: one may specify modifiers of a modifier, and so on. Finally, the extended formalism allows for lexicalised phrases of any type used in Walenty – `lex` metacategory takes any base phrase type as one of its parameters and relevant constraints as remaining parameters.

[10] Complex prepositions are not discussed since they are treated as an instance of a lexicalised prepositional phrase in Walenty.

With lexicalised varieties of nominal and prepositional phrases, `lexnp` and `preplexnp` (see (8.32) and (8.33), respectively), it is possible to additionally restrict the following features of the nominal element, which is the semantic centre of the relevant lexicalised phrase: number, lemma and modification pattern.

(8.32) `lexnp(case,number,lemma,modification)`

(8.33) `preplexnp(preposition,case,number,lemma,modification)`

There are three values of `number`:

- `sg` when the singular form is required,
- `pl` when the plural form is required,
- `_` (underscore) when any value of number is possible.

The `lemma` parameter may take any value – it hosts the lemma required in the given position by the predicate.

Finally, the `modification` parameter determines which modification pattern is possible with the given nominal. Let us proceed to discussing modification patterns – there are four defined values in the September 2014 version of Walenty:

(8.34) `atr`: modification allowed (though not necessary),

(8.35) `ratr`: modification required (often possessive, NP or adjective),

(8.36) `batr`: specific modification required (possessive: SWÓJ 'self's' or WŁASNY 'own'),

(8.37) `natr`: modification not allowed.

Let us consider some examples to illustrate how lexicalised constraints work in Walenty:

(8.38) (Gorąca) krew/*krwie płynie/*płyną w *(jej/Marysi/tych) żyłach/*żyle.
hot blood.SG/PL flow.SG/PL in her/Marysia's/those vein.PL/SG
'(Hot) blood flows in her/Marysia's/those veins.'

Two lexicalised arguments are involved in (8.38), one nominal and one prepositional, whose nominal centres are restricted to the lemmata KREW and ŻYŁA, respectively. As demonstrated in the example, the nominal in the nominal phrase (KREW) must be specified for the singular number, while the nominal in the prepositional phrase (ŻYŁA) must be plural. Furthermore, the first nominal in (8.38) *may* be modified, while the nominal from the prepositional phrase *must* be modified – it is ungrammatical without a possessive modifier.

This is how (8.39), the schema taken from one of the entries of the verb PŁYNĄĆ 'flow', captures the relevant lexicalised constraints discussed above for (8.38):

(8.39) `subj{lexnp(str,sg,'krew',atr)} + {preplexnp(w,loc,pl,'żyła',ratr)}`

Let us consider another example with a lexicalised argument:

(8.40) Doręczyli to jej do rąk *(własnych).
delivered it.ACC she.DAT to hands.GEN own.GEN
'They delivered it to her as hand delivery.'
(literally: 'They delivered it to her to (her) own hands.')

In (8.40) the lexicalised prepositional phrase restricts the lemma of its nominal centre to RĘKA. Furthermore, it must be plural and it requires a specific possessive modifier, namely a form of WŁASNY. (8.41), one of the entries of the verb DORĘCZYĆ 'deliver', shows how these constraints are formalised:

```
(8.41)    subj{np(str)} + obj{np(str)} + {np(dat)}
          + {preplexnp(do,gen,pl,'ręka',batr)}
```

However, there is a shortcoming of such a formalisation of the modification restriction: as described in (8.36), the `batr` modification pattern accepts two kinds of possessive modifiers, namely forms of SWÓJ and WŁASNY, even though only the latter is grammatical in (8.40).[11]

The following example illustrates the last modification pattern, `natr`:

(8.42) Daję (*swoją/*mądrą) głowę/*głowy, że przyjdą.
give own/wise.ACC.SG head.ACC.SG/PL that come.FUT
'I'm sure that they will come.' (literally: 'I give (my) head that they will come.')

(8.42) shows that the nominal, whose lemma is restricted to GŁOWA, must be specified for singular number and that it does not accept any modifiers. It must be noted, however, that forbidding modification (see (8.37)) is not equivalent to being a fixed expression (`fixed` category, see § 8.2.7.1). Consider a slightly modified version of the example provided in (8.42):

(8.43) Nie daję (*swojej/*mądrej) głowy, że przyjdą.
NEG give own/wise.GEN.SG head.GEN.SG that come.FUT
'I'm not sure that they will come.'
(literally: 'I don't give (my) head that they will come.')

(8.43) demonstrates that even though the nominal GŁOWA does not accept modification, it can (or rather has to) appear in a different form, namely the genitive case, as a result of structural case assignment in the scope of sentential negation (see § 3.2.3 for discussion). Such a change would not be allowed with the category `fixed`.[12] The schema for the verb DAĆ 'give', which accounts for (8.42) and (8.43), is provided in (8.44):

```
(8.44)    subj{np(str)} + {cp(że)} + {lexnp(str,sg,'głowa',natr)}
```

8.3 Conversion to LFG constraints

The aim of this section is to show how entries taken from Walenty are converted to constraints used by the Polish LFG grammar implemented in XLE. It provides information about choosing

[11] As mentioned in fn. 9, the extended formalism for handling lexicalised arguments described in Przepiórkowski *et al.* 2014a is capable of imposing appropriate constraints in such examples.

[12] Though it would be possible to use two unrelated schemata with `fixed` arguments, one requiring the accusative case (`fixed('głowę')`) and forbidding negation, the other requiring the genitive case (`fixed('głowy')`) and requiring negation. Such an account would, however, miss a generalisation.

the grammatical function for each argument (§ 8.3.1) and imposing relevant constraints (§ 8.3.2). It also discusses the formation of passive voice (§ 8.3.3) and reducing schemata (§ 8.3.4).

8.3.1 Choosing the grammatical function

When converting constraints stated in Walenty to XLE/LFG constraints, the first step is to choose an appropriate grammatical function for each argument.

Since this task is considerably easier when coordination is not involved (see § 4.2.1), let us start with this simple case.

8.3.1.1 Without coordination

When a grammatical function is chosen for an argument which corresponds to a singleton set, unlike coordination is not involved – the particular syntactic position has only one realisation.[13]

In this situation it is possible to use a simple mapping from morphosyntactic categories used in Walenty to grammatical functions used in LFG. Note that grammatical functions are only assigned to positions which are not explicitly assigned a grammatical function in Walenty (these are: `subj`, the subject, and `obj`, the passivisable object; see § 8.2.1).

The mapping from phrasal categories to LFG grammatical functions for positions which are not assigned a grammatical function in Walenty is summarised below:

(8.45) [`np` ∨ `ncp` ∨ `lexnp` ∨ `adjp`] ∧

a. `controllee` → XCOMP-PRED

b. `case == dat` → OBJ-TH

c. [`case == str` ∨ `case == part`] → OBL-STR

d. `case == gen` → OBL-GEN

e. `case == inst` → OBL-INST

(8.46) [`prepnp` ∨ `prepncp` ∨ `preplexnp` ∨ `prepadjp` ∨ `comprepnp`] →

a. `controllee` → XCOMP-PRED

b. OBL (numerical index is appended when there is more than one argument of this type: OBL2, OBL3, etc.)

(8.47) `cp` → COMP

(8.48) `infp` → XCOMP

(8.49) `nonch` → OBL-STR

(8.50) `advp` → OBL-ADV

(8.51) `xp(sem)` → OBL-SEM (e.g. `xp(abl)` → OBL-ABL)

[13] There is an exception: as discussed in § 8.2.6, `xp` phrases are container phrases, which rewrite to a range of categorial realisations with common semantics. Since in this case the assignment of grammatical function is based on the semantic type of a particular `xp` category (see (8.51)), it is not problematic.

(8.52) `fixed` → OBL-ADV (a guess: not enough information[14] – no category)

(8.53) `or` → COMP

(8.54) `refl` → marker (co-head, not a GF)

Let us discuss (8.45), the rule which defines how nominal phrases, plain (`np`) and lexicalised (`lexnp`), which do not have any grammatical function label in Walenty, are assigned a grammatical function. The first rule, (8.45a), assigns the XCOMP-PRED grammatical function to nominals labelled as `controllee` in Walenty. This is the case with the second argument in (8.55), the schema taken from the entry of the verb CZUĆ SIĘ 'feel' (see the example in (8.56)):

(8.55) `subj,controller{np(str)} + controllee{np(inst)}`

(8.56) czują się tam intruzami
feel REFL there intruders.INST
'They feel there like intruders.' (NKJP)

The remaining mapping rules for nominals defined in (8.45b–e) are applicable when the nominal is not labelled as `controllee` – in such situations the grammatical function is assigned on the basis of the nominal's case: without the `controllee` label, a nominal argument marked for the instrumental case (e.g. `np(inst)`) would be assigned the OBL-INST grammatical function.

As explained at the beginning of this section, the first argument in (8.55) is assigned the subject grammatical function since the grammatical function of this position is specified in Walenty. For this reason, the mapping rule for `np` in (8.45c) is not applicable in this context.

Let us see how the mapping defined in (8.45)–(8.54) above works on the basis of some schemata from Walenty. The sentence in (8.58) is handled by the schema for the verb GŁOWIĆ SIĘ 'ponder, think hard' in (8.57):

(8.57) `subj{np(str)} + {cp(int)}`

(8.58) fachowcy głowią się, jak zasiedlić Gródek.
expert ponder REFL how settle Gródek
'Experts ponder how to settle Gródek.' (NKJP)

The schema in (8.57) contains two positions: one is labelled as `subj`, so it is assigned a grammatical function already in Walenty. The second set is unlabelled, so the grammatical function appropriate for it has to be determined in the process of conversion. According to the mapping rule provided in (8.47), the COMP grammatical function is appropriate for `cp` phrases regardless of their type (there are phrases featuring a complementiser and interrogative phrases).

Let us consider another, slightly more complex example. The schema for the verb DRZEĆ SIĘ 'cry, yell' provided in (8.59) is appropriate for handling the sentence in (8.60):

(8.59) `subj{np(str)} + {prepnp(do,gen)} + {prepnp(o,acc)}`

[14] Note that this issue was resolved in subsequent versions of Walenty – now `fixed` has a parameter which specifies its category (together with parameters appropriate for this category).

(8.60) Wasza podświadomość drze się do mnie o pomoc
your subconsciousness cries REFL to I.GEN about help.ACC
'Your subconsciousness cries to me for help.' (NKJP)

The schema in (8.59) has three arguments, one of which, the subject, is assigned a grammatical function in Walenty. The two remaining positions are prepositional phrases. The mapping rule for prepositional arguments is provided in (8.46) – such phrases are assigned the OBL grammatical function, but there are two such phrases in (8.59). The rule takes such cases into account – the first prepositional phrase (`prepnp(do,gen)`) is assigned the basic OBL grammatical function, while the second one (`prepnp(o,acc)`) is assigned the same grammatical function with a numerical index: OBL2. If there were some more prepositional phrases in this schema, they would be assigned the OBL grammatical function with subsequent indices: OBL3, etc. This step ensures that grammatical functions are unique, making it possible to avoid violations of the uniqueness (consistency) principle (see § 2.2.2).

8.3.1.2 Under coordination

While the choice of grammatical function is straightforward when the argument in question is a non-coordinate item, it is more complicated when unlike coordination is involved (as discussed in § 4.2.2): since unlike categories prototypically correspond to distinct grammatical functions (when coordination is not taken into account), the problem of choosing a common grammatical function arises.

For the purposes of Polish LFG, in such a situation candidate grammatical functions are collected and the common grammatical function is chosen on the basis of a ranking: the highest ranked function is chosen from the set of candidate grammatical functions. The following ranking is used to choose the common grammatical function for the relevant coordinate phrase:

(8.61)

#	GF
5	OBL-ABL, OBL-ADL, OBL-DUR, OBL-LOCAT, OBL-MOD, OBL-PERL, OBL-TEMP, OBL-ADV
4	OBL
3	OBL-GEN, OBL-INST, OBL-STR, OBJ-TH
2	COMP, XCOMP

In (8.61) semantic grammatical functions are highest-ranked (#5), together with OBL-ADV, the grammatical function corresponding to adverbial arguments and the `fixed` category. They are immediately followed by non-semantic obliques (#4). The next position (#3) is occupied by grammatical functions corresponding to nominal arguments marked for a certain case. Finally, clausal complements are lowest-ranked (#2).

If the argument is not assigned a grammatical function in Walenty (`subj` or `obj` – see § 8.2.1), the following procedure is used for coordinate arguments:

(8.62) each realisation of a given argument is assigned a grammatical function according to the mapping presented in § 8.3.1.1 in (8.45)–(8.54); a list of candidate grammatical functions results, (8.63) the resulting list is turned into a set (to remove repeated items),

(8.64) each grammatical function from the resulting set is assigned a ranking according to (8.61); each ranking position has a corresponding list of grammatical functions which were assigned such ranking,

(8.65) the highest ranking position is chosen,

(8.66) if the list corresponding to the chosen ranking position contains more than one element, candidate grammatical functions are inspected manually; else the only element of the list is chosen as the common grammatical function under coordination.

In summary: every realisation is assigned a grammatical function as if it was the only realisation of the given position (as if coordination was not involved), then each assigned grammatical function is ranked, finally the highest ranked grammatical function is chosen as the common grammatical function for the entire syntactic position.

It is perhaps worth underlining that the mechanism of choosing the common grammatical function under coordination is based on the ranking of grammatical functions rather than their relative frequency – even if the lower-ranked grammatical function were appropriate for 5 conjuncts, while the higher-ranked grammatical function were appropriate for only one conjunct, the higher-ranked grammatical function would be chosen. This is a ranking mechanism, not voting – this is underlined by turning the list of candidate grammatical functions into a set.

Let us consider (8.67), a sentence where one of the arguments is an instance of unlike category coordination – one conjunct is an interrogative clause, while the other is a prepositional phrase with a correlative pronoun taking a sentential complement (also an interrogative clause):

(8.67) Pytali, [jakie będą pieniądze] oraz [o to, czy zmienią się polskie szkoły].
asked what will be money and about this PART change REFL Polish schools
'They asked what money will be there and whether Polish schools will change.'
(NKJP)

The schema which handles (8.67) is provided in (8.68). Let us go through the process of assigning grammatical functions in this schema taken from the verb PYTAĆ 'ask'.

(8.68)
```
subj{np(str)} + obj{np(gen)}
+ {prepnp(o,acc); cp(int); prepncp(o,acc,int)}
```

The first two arguments have grammatical functions assigned in Walenty – they are labelled as `subj` and `obj`, respectively. The last, third argument does not have any grammatical function label, so the common grammatical function is assigned in the process of conversion, as described in (8.62)–(8.66) above. Let us follow this procedure step-by-step.

As described in (8.62), each realisation of the relevant argument is assigned a grammatical function individually. The table provided in (8.69) explains this procedure: in the first row each column contains one realisation; the second row contains candidate grammatical functions assigned according to § 8.3.1.1; the last row contains references to mapping rules used in the process of assigning the relevant grammatical function.

(8.69)

realisation	`prepnp(o,acc)`	`cp(int)`	`prepncp(o,acc,int)`
candidate GF	`OBL`	`COMP`	`OBL`
mapping rule	(8.46)	(8.47)	(8.46)

The following list is produced as a result of (8.62):[15]

(8.70) `[OBL, COMP, OBL]`

The next step is (8.63), it turns the list from (8.70) into the following set:

(8.71) `{OBL, COMP}`

The third step is (8.64), whereby each grammatical function in the relevant set is ranked according to (8.61); the following ranking is created:

(8.72) `{4: [OBL], 2: [COMP]}`

Next, according to (8.65), the highest value is chosen:

(8.73) `4: [OBL]`

Finally, following (8.66), the list which corresponds to the ranking position in (8.73) is inspected. It is a one-element list, so there is only one candidate – the common grammatical function of the coordinate phrase in (8.68) is OBL.

8.3.2 Imposing constraints

Once the grammatical function corresponding to a given syntactic position has been chosen, appropriate constraints are imposed for each realisation of the relevant syntactic position defined in the lexical entry.

8.3.2.1 Plain constraints

When coordination is not involved, plain constraints are used. Constraints to be imposed depend on the category of the relevant argument and values of parameters that it takes. Let us consider a few examples.

When a plain nominal phrase is used, the only constraint to be imposed is the value of case – the template defined in (8.74) (repeated from (7.21)) is used for this purpose:

(8.74)
```
CASE-SUBC-PATH(PATH C) = "checks that case in PATH is equal to C"
(PATH CASE)=c C.
```

This template takes two parameters, `PATH` and `C`, and it requires that the value of `CASE` in `PATH` is equal to `C`. Let us see how (8.74) can be used in the conversion of schemata from Walenty.

In (8.68), repeated below as (8.75), the second position is `obj{np(gen)}`:

(8.75)
```
subj{np(str)} + obj{np(gen)}
+ {prepnp(o,acc); cp(int); prepncp(o,acc,int)}
```

[15] The script used for converting Walenty to LFG constraints is written in the Python programming language, so its notational conventions for representing data structures are adopted here.

The grammatical function of the second argument of (8.75), `obj{np(gen)}`, is specified in Walenty: the label `obj` corresponds to the passivisable object, it therefore bears the `OBJ` grammatical function. It is now possible to impose relevant constraints:[16] as mentioned above, plain nominal phrases are only specified for case. According to the information provided in (8.75), the object must be specified for the lexical genitive case – (8.76) shows how this constraint can be imposed using a call to the template defined in (8.74):

(8.76) `@(CASE-SUBC-PATH (^ OBJ) gen)`

According to the definition of the template `CASE-SUBC-PATH` provided in (8.74), it takes two parameters: `PATH`, which is the path in which the value of the `CASE` attribute is set, and `C`, which stands for the particular value of case. In the template call shown in (8.76) the first parameter of `CASE-SUBC-PATH` is `(^ OBJ)`, while the second one is `gen`. The following constraint is introduced in the lexical entry of PYTAĆ 'ask' as a result of (8.76):[17]

(8.77) `(^ OBJ CASE)=c gen`

Let us now see how plain prepositional phrases are converted. Consider the schema for the verb DRZEĆ SIĘ 'cry, yell' in (8.59), repeated as (8.78) below:

(8.78) `subj{np(str)} + {prepnp(do,gen)} + {prepnp(o,acc)}`

As discussed in § 8.3.1.1 (see mapping rule (8.46) and the discussion following (8.59)), the first prepositional argument is assigned the `OBL` grammatical function, while the second one is an `OBL2`.[18] Plain prepositional phrases impose two constraints: the form of the preposition and the case of the nominal. Let us start with the first prepositional phrase in (8.78), namely `prepnp(do,gen)`. Let us start with the constraint imposing the appropriate form of the preposition – the template `PFORM-SUBC-PATH` defined in (8.79) is used for this purpose.

(8.79)
```
PFORM-SUBC-PATH(PATH PF) = "checks PFORM in PATH is equal to PF"
(PATH PFORM)=c PF.
```

(8.79) is similar to (8.74) since it takes two parameters: `PATH`, like (8.74), and `PF`, which hosts the required preposition form.

To ensure, as specified in `prepnp(do,gen)`, that the form of the preposition is `do`, (8.79) would be called in the following way:

(8.80) `@(PFORM-SUBC-PATH (^ OBL) do)`

The following constraint is produced as a result of calling (8.80):

(8.81) `(^ OBL PFORM)=c do`

[16] This section discusses constraints holding for active voice, see § 8.3.3 for discussion of constraints under passive voice.

[17] See § 7.2.2 for discussion of why templates should be used even if their definition is considerably longer than the obtained constraint.

[18] Prepositional phrases in (8.78) are assumed to be non-semantic (see § 2.3) and appropriate constraints are imposed. Note that constraints imposed for semantic prepositions are different – see § 7.3.

Secondly, the constraint on the value of case of the nominal is introduced using the template (8.74) discussed above – its first parameter is the relevant path (`(^ OBL)`), the second parameter is the required case (`gen`):

(8.82) `@(CASE-SUBC-PATH (^ OBL) gen)`

The result of (8.82) is provided below:

(8.83) `(^ OBL CASE)=c gen`

The second prepositional phrase used in (8.78), namely `prepnp(o,acc)`, would use the following constraints (as the second prepositional phrase, it was assigned the `OBL2` grammatical function):

(8.84) `@(PFORM-SUBC-PATH (^ OBL2) o)`

(8.85) `@(CASE-SUBC-PATH (^ OBL2) acc)`

As a result, the following constraints would be introduced:

(8.86) `(^ OBL2 PFORM)=c o`

(8.87) `(^ OBL2 CASE)=c acc`

Let us now consider how clauses are converted to LFG constraints. (8.57), the schema for the verb GŁOWIĆ SIĘ ‘ponder, puzzle over’, is repeated in (8.88) below:

(8.88) `subj{np(str)} + {cp(int)}`

According to the mapping rule (8.47), the argument `cp(int)` is assigned the `COMP` grammatical function. Constraints are assigned according to the type of the clause – there are phrases featuring a complementiser such as `cp(że)`, while `cp(int)` is a phrase containing an interrogative element which may correspond to any grammatical function or a marker (as with the *yes/no* question particle CZY). The relevant constraint is introduced using the template defined in (8.89):

(8.89)
```
CL-TYPE-SUBC-PATH(PATH CL) = "checks that CLAUSE-TYPE in PATH is equal
to CL"
(PATH CLAUSE-TYPE)=c CL.
```

The template `CL-TYPE-SUBC-PATH` takes two parameters: `PATH` and `CL`. The result of this template is that it checks that the value of the attribute `CLAUSE-TYPE` is equal to `CL` in the `PATH`.

(8.90) is the call of the template (8.89), which introduces constraints for `cp(int)`:

(8.90) `@(CL-TYPE-SUBC-PATH (^ COMP) int)`

(8.90) passes `(^ COMP)` as `PATH` and `int` as `CL`, the value of the attribute `CLAUSE-TYPE`. The result of this call is shown in (8.91):

(8.91) `(^ COMP CLAUSE-TYPE)=c int`

8.3.2.2 Constraints under coordination

Since entire statements are distributive in XLE, constraints handling unlike category coordination must be formalised differently than plain constraints – as described in § 4.3 (see the discussion of (4.19), repeated as (8.92) below), if a plain disjunctive constraint is used, it is evaluated once (one disjunct is chosen) and applied to all conjuncts, as formalised in (8.92b).

By constrast, the interpretation which is needed to handle unlike category coordination is the one formalised in (8.92a) – it evaluates the relevant statement for each conjunct separately, so it is possible that different conjuncts satisfy different constraints.

(8.92) a. $\forall x \in (\uparrow \textsc{gf})[A(x) \vee B(x)]$ (intended)

b. $\forall x \in (\uparrow \textsc{gf})A(x) \vee \forall x \in (\uparrow \textsc{gf})B(x)$ (actual)

The solution to this problem was discussed in § 4.4 – it relies on the use of off-path constraints in order to obtain the effect shown in (8.92a); see § 4.4.2 for a detailed explanation. In short: constraints to be imposed on a given argument are converted to their off-path equivalent and they are attached to the PRED attribute of the relevant argument – this attribute is distributive by definition, which ensures that the off-path constraint will be distributed to each conjunct and evaluated separately.

Let us see the off-path constraint which corresponds to the third argument of the verb PYTAĆ 'ask', `{prepnp(o,acc); cp(int); prepncp(o,acc,int)}` – its schema provided in (8.68) is repeated as (8.93):

(8.93)
```
subj{np(str)} + obj{np(gen)}
+ {prepnp(o,acc); cp(int); prepncp(o,acc,int)}
```

(8.94)
```
(^ OBL PRED:
       {(<- PFORM)=c o (<- CASE)=c acc (<- CORRELATIVE)~= +
       | (<- CLAUSE-TYPE)=c int | (<- PFORM)=c o (<- CASE)=c acc
       (<- CORRELATIVE)=c + (<- COMP CLAUSE-TYPE)=c int})
```

The common grammatical function of the coordinate phrase is `OBL` (as explained in (8.70)–(8.73)), so the off-path constraints are attached to the `PRED` attribute of this grammatical function. There are three off-path disjuncts in (8.94), one for each realisation of the corresponding argument in Walenty – these constraints apply to the f-structure of `OBL`. The first disjunct corresponds to `prepnp(o,acc)` – it requires that the value of the `PFORM` attribute of `OBL` (or one of its conjuncts) is equal to `o` and that `CASE` is equal to `acc`. The second disjunct corresponds to `cp(int)` – it requires an interrogative clause. The third disjunct corresponds to `prepncp(o,acc,int)` – it is satisfied by a prepositional phrase where the nominal takes a clausal complement which is interrogative. The constraint provided in (8.94) is satisfied if each element of the `OBL` f-structure satisfies one of its disjuncts. (8.95) given below is a more economic, packed version of (8.94) – it shares relevant parts of constraints for individual realisations (such as `(<- CASE)=c acc` and `(<- PFORM)=c`) and, while being equivalent to (8.94), makes it possible to avoid redundancy.

```
(8.95)   (^ OBL PRED:
                {(<- CLAUSE-TYPE)=c int | (<- CASE)=c acc (<- PFORM)=c o
                {(<- CORRELATIVE)~= + | (<- CORRELATIVE)=c +
                (<- CLAUSE-TYPE)=c int}})
```

There is one more important thing to note: plain constraints presented in this chapter often use templates as a convenient means of abbreviating the constraints. It is not possible, however, to use this formal device with statements which include off-path constraints, as in (8.94). The reason for this is that while it is possible to assign entire statements such as (8.94) to a template, it is not possible to assign fragments of off-path constraints such as `(<- CLAUSE-TYPE)=c int` to a template.[19]

8.3.2.3 Structural case assignment

After having introduced off-path constraints, it is possible to proceed to how structural case assignment is formalised in the conversion of Walenty.

Let us start with the subject. Since the formalisation of structural case assignment using plain constraints was discussed in § 7.5 (concentrating on past tense forms), let us therefore proceed to the discussion of the subject grammatical function being an instance of unlike category coordination. Such a situation was discussed from the theoretical perspective in § 4.5.1, where (4.42) was offered as the statement handling structural case assignment to nominal non-unlike subjects (noun or numeral); its XLE counterpart is provided in (8.96) below:

```
(8.96)   (^ SUBJ PRED:
                 {(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec})
         (^ SUBJ CORRELATIVE)~= +
```

The only difference with respect to (4.42) is the constraint `(^ SUBJ CORRELATIVE)~= +`, which ensures that the subject is not correlative – it corresponds to the `subj{np(str)}` argument in Walenty, which allows nominals (nouns and numerals) which do not subcategorise for a clause, unlike the `subj{ncp(str,że)}` argument, which requires the subject to be a nominal (more precisely, the correlative pronoun TO 'this') taking a clause with a ŻE-type complementiser.

If the subject can be realised by some different phrases alongside `np(str)`, constraints corresponding to it are added as additional off-path disjuncts, as in (8.98) – the off-path disjuncts used in this constraint correspond to the first three realisations (`np(str)`, `cp(int)` and `cp(że)`) of the subject argument in (8.97), one of the schemata of the verb DOTRZEĆ 'reach':

```
(8.97)   subj{np(str); cp(int); cp(że); ncp(str,int); ncp(str,że)}
         + {prepnp(do,gen)}
```

```
(8.98)   (^ SUBJ PRED:
                 {{(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec}
                 (<- CORRELATIVE)~= + | (<- CLAUSE-TYPE)=c int
                 | (<- COMP-FORM)=c że | ...})
```

[19] The relevant fragment of the XLE documentation is available at `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N3.2`. However, the information provided there may be considered misleading: "Note that you cannot have off-path constraints in macros."

Let us proceed to the structural case assignment to objects. Plain cases where unlike category coordination is not involved were covered in § 7.4.2. Structural case assignment to unlike category objects is handled using the same mechanism as subjects, namely off-path constraints – see § 4.5.2 for theoretical discussion. A near counterpart of (4.78) discussed there is provided in (8.99)[20] in XLE notation:

```
(8.99)    (^ OBJ PRED:
                 {
                 "no negation at all: accusative"
                 ~(({XCOMP|XCOMP-PRED}* OBJ <-) NEG) (<- CASE)=c acc
                 |
                 "negation present at some level"
                 (({XCOMP|XCOMP-PRED}* OBJ <-) NEG)=c +
                    {
                    "local negation: obligatory GoN"
                    ((OBJ <-) NEG)=c +
                    (<- CASE)=c gen
                    |
                    "no local negation, only transferred: optional GoN"
                    ~((OBJ <-) NEG)
                    (<- CASE) $c {acc gen}
                    }
                 })
```

There are two differences to note: (8.99) uses `{XCOMP|XCOMP-PRED}` as the path instead of XCOMP used in (4.78). Secondly, the last off-path disjunct of (4.78), namely $(\leftarrow$ COMP-FORM$) =_c$ ŻEBY, is not included in (8.99) – the latter only provides the off-path constraint related to structural case assignment to the object. When unlike category coordination is involved, appropriate disjuncts can be added to (8.99).

Let us therefore consider an example: (8.100) is the schema taken from the entry of the verb POLECAĆ 'recommend' and (8.101) provides the converted constraints related to the first two realisations of the second argument labelled as `obj` in Walenty (`np(str)` and `cp(żeby)`):[21]

```
(8.100)   subj{np(str)} + obj,controllee{np(str); cp(żeby); ncp(str,że); infp(_)}
          + controller{np(dat)}
```

```
(8.101)   (^ OBJ PRED:
                 {{~(({XCOMP|XCOMP-PRED}* OBJ <-) NEG) (<- CASE)=c acc |
                 (({XCOMP|XCOMP-PRED}* OBJ <-) NEG)=c +
                 {((OBJ <-) NEG)=c + (<- CASE)=c gen |
                 ~((OBJ <-) NEG) (<- CASE) $c {acc gen}}}
```

[20] As explained at the end of § 8.3.2.2, while the entire statement defined in (8.99) can be assigned to a template in XLE, fragments corresponding to individual off-path constraints cannot be assigned to templates in the way this was done in ch. 4 in the theoretical part (see (4.78) and (4.71)–(4.77)).

[21] Choosing these 2 realisations makes it possible to avoid the issue of control into selected conjuncts, which would greatly complicate the notation. For discussion of how this issue can be solved, see § 4.7.

```
(<- CORRELATIVE)~= + | (<- COMP-FORM)=c żeby | ...})
```

The off-path constraint in (8.101) is the result of adding a disjunct corresponding to the `cp(żeby)` realisation, namely `(<- COMP-FORM)=c żeby`, to the statement provided in (8.99), which corresponds to the `np(str)` realisation of the argument bearing the `OBJ` grammatical function. The resulting constraint in (8.101), corresponding to the first two realisations of the argument labelled as `obj` in (8.100), is an implementation counterpart of the constraint defined in (4.78).

8.3.2.4 Realisations of container arguments

As discussed in § 8.2.6, certain types of arguments are classified according to their semantics – this applies to categories `xp` and `advp`.They have defined lists of realisation, see (8.26) for realisations of `xp(abl)` and (8.27) for `advp(abl)`, which is one of the realisations of `xp(abl)`.

Since these categories can be realised in a number of ways, including a coordination of these realisations, off-path constraints are used when converting relevant schemata from Walenty – in such constraints one off-path disjunct corresponds to one realisation of the given category.

An example is provided below: (8.103) is the converted constraint corresponding to `xp(abl)`, the last argument in the schema for the verb ABSORBOWAĆ 'absorb' provided in (8.102):

(8.102) `subj{np(str)} + obj{np(str)} + {prepnp(przez,acc)} + {xp(abl)}`

(8.103)
```
(^ OBL-ABL PRED:
          {(<- PRED FN)=c skąd | (<- PRED FN)=c skądkolwiek
          | (<- PRED FN)=c skądś | (<- PRED FN)=c skądże
          | (<- PRED FN)=c stamtąd | (<- PRED FN)=c stąd
          | (<- PRED FN)=c zewsząd | (<- PRED FN)=c znikąd
          | (<- PRED FN)=c skądinąd | (<- PRED FN)=c skądciś
          | (<- PRED FN)=c skądsiś | (<- CLAUSE-TYPE)=c int
          (<- ADJUNCT $ TYPE)=c int (<- ADJUNCT $ PRED FN)=c skąd
          | (<- PRED FN)=c z (<- OBJ CASE)=c gen
          (<- OBJ CHECK _CAT) $c {adj ppas pact}
          (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c od
          (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +
          | (<- PRED FN)=c spod (<- OBJ CASE)=c gen
          (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c spomiędzy
          (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +
          | (<- PRED FN)=c sponad (<- OBJ CASE)=c gen
          (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c spopod
          (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +
          | (<- PRED FN)=c spośród (<- OBJ CASE)=c gen
          (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c spoza
          (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +
          | (<- PRED FN)=c sprzed (<- OBJ CASE)=c gen
          (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c z
```

```
        (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +
        | (<- PRED FN)=c znad (<- OBJ CASE)=c gen
        (<- OBJ CORRELATIVE)~= + | (<- PRED FN)=c zza
        (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +})
```

The initial off-path disjuncts in (8.103) correspond to particular realisations of `advp(abl)` listed in (8.27), the first realisation of `xp(abl)` defined in (8.26), the remaining disjuncts correspond to the remaining realisations listed there.

(8.104) is a packed version of (8.103) – while the latter is redundant to a large degree, the former contains the same information (almost) without redundancy:

(8.104)
```
(^ OBL-ABL PRED:
        {(<- PRED FN) $c {skąd skądkolwiek skądś skądże stamtąd
        stąd zewsząd znikąd skądinąd skądciś skądsiś}
        | (<- CLAUSE-TYPE)=c int (<- ADJUNCT $ TYPE)=c int
        (<- ADJUNCT $ PRED FN)=c skąd | (<- PRED FN)=c z
        (<- OBJ CASE)=c gen (<- OBJ CHECK _CAT) $c {adj ppas pact}
        (<- OBJ CORRELATIVE)~= + | (<- PRED FN) $c {od spod
        spomiędzy sponad spopod spośród spoza sprzed z znad zza}
        (<- OBJ CASE)=c gen (<- OBJ CORRELATIVE)~= +})
```

8.3.2.5 Realisations of clausal arguments

While most clausal arguments such as `cp(że)` are rewritten to simple constraints, some clausal arguments defined in Walenty require more sophisticated constraints as they take the syntactic context into account (this is slightly similar to structural case assignment to objects). Such arguments include `cp(żeby2)` and `cp(gdy)`.

The category `cp(żeby)` represents a clause containing the complementiser of the `żeby` type – it is one of the realisations of the third argument in the schema of the verb BŁAGAĆ 'beg' provided in (8.106). The sentence in (8.105) illustrates this schema: the relevant argument is realised as a coordination of the first realisation (`prepnp(o,acc)`) and the second one (`cp(żeby)`).[22]

(8.105) BŁAGAŁ [O SPOTKANIA] I [ŻEBYM DO NIEGO WRÓCIŁA].
begged for meetings.ACC and that-1.SG to he.GEN returned
'He begged (me) for meetings and to return to him.' (NKJP)

(8.106)
```
subj{np(str)} + obj{np(str)}
+ {prepnp(o,acc); cp(żeby); prepncp(o,acc,żeby)}
```

The relevant constraint (taking the third realisation into consideration as well) is provided in (8.107) – the first off-path disjunct requires that the value of the `PFORM` attribute is equal to `o` and the case of the accompanying nominal is accusative, while the second off-path disjunct requires that the value of the `COMP-FORM` attribute is equal to `żeby`, the third off-path disjunct requires

[22] (8.105) features the word *żebym*: it is the result of attaching *m*, the agglutinate first person singular form of BYĆ 'be' (see Przepiórkowski and Woliński 2003), to the complementiser ŻEBY – as indicated in glosses.

`PFORM` equal to `o`, the accusative case from the correlative nominal (`+` value of `CORRELATIVE` attribute) and `żeby` as the value of `COMP-FORM` of the `COMP` argument of the correlative nominal. The packed version of (8.107) is provided in (8.108).

(8.107)
```
(^ OBL PRED:
      {(<- PFORM)=c o (<- CASE)=c acc (<- CORRELATIVE)~= +
      | (<- COMP-FORM)=c żeby | (<- PFORM)=c o (<- CASE)=c acc
      (<- CORRELATIVE)=c + (<- COMP COMP-FORM)=c żeby})
```

(8.108)
```
(^ OBL PRED:
      {(<- COMP-FORM)=c żeby | (<- CASE)=c acc (<- PFORM)=c o
      {(<- CORRELATIVE)~= + | (<- CORRELATIVE)=c +
      (<- COMP-FORM)=c żeby}})
```

The category `cp(żeby2)` is different from `cp(żeby)` since the former can be realised in two ways: always as `że` and as `żeby` only in scope of sentential negation.[23] Consider the following examples, which illustrate the schema for the verb WYOBRAZIĆ 'imagine' provided in (8.111):

(8.109) Ja *(nie) mogę sobie wyobrazić, żeby ktoś mógł zrobić coś takiego.
I NEG can SELF imagine that somebody could do.INF something such
'I can't imagine that somebody could have done something like this.' (NKJP)

(8.110) Ja (nie) mogę sobie wyobrazić, że ktoś mógł zrobić coś takiego.
I NEG can SELF imagine that somebody could do.INF something such
'I can (not) imagine that somebody could have done something like this.'

(8.111) `subj{np(str)} + {cp(żeby2)} + {lexnp(dat,_,'siebie',natr)}`

(8.112) `{(^ NEG)=c + (^ COMP COMP-FORM)=c żeby | (^ COMP COMP-FORM)=c że}`

The constraint corresponding to `cp(żeby2)`, the second argument of (8.111), is provided in (8.112) – it states that the complementiser `żeby` is only possible when negation is present, while `że` is possible at all times (there are no constraints on the value of `NEG`).

There is another clausal category which is similar to `cp(żeby2)` in that its realisation depends on the syntactic context – it is `cp(gdy)`. This category can be realised as `gdyby` only when the main clause verb is in the conditional mood (as in (8.114)), and as `gdy` elsewhere (see (8.113)). The constraint corresponding to `cp(gdy)`, the third argument of the verb OBLECIEĆ 'seize' in (8.115), is provided in (8.116):

(8.113) Obleciał/(*by) mnie strach, gdy ktoś szedł za mną ciemną ulicą.
seized/seize.COND me fear when somebody walked after me dark street
'I was seized by fear when somebody followed me in the dark street.' (NKJP)

[23] It must be noted, however, that `żeby` may also be used as the realisation of `cp(żeby2)` in generally negative contexts such as in (i), where the the verb WYOBRAZIĆ 'imagine' takes `cp(żeby2)` as one of its arguments:

(i) Z trudem mogę sobie wyobrazić, żeby...
with difficulty can SELF imagine that
'It is only with difficulty that I can imagine...'

(8.114) Obleciałby mnie strach, gdyby/*gdy ktoś szedł za mną ciemną ulicą.
seize.COND me fear if/when somebody walked after me dark street
'I would be seized by fear if somebody followed me in the dark street.'

(8.115)
```
subj{lexnp(str,sg,'strach',natr)} + {np(str)} + {cp(gdy)}
```

(8.116)
```
{(^ TNS-ASP MOOD)=c conditional (^ COMP COMP-FORM)=c gdyby
| (^ TNS-ASP MOOD)~= conditional (^ COMP COMP-FORM)=c gdy}
```

Note that, unlike in the case of `cp(żeby2)` discussed above, the distribution of particular realisations of `cp(gdy)` is complementary. This should be visible in (8.116): `gdyby` is required as the complementiser when the verb is in the conditional mood, while `gdy` is required when the mood is not conditional. By contrast, `cp(żeby2)` can always be realised as `że` and as `żeby` only when negation is available, as formalised in (8.112) – the distribution of particular realisations of `żeby2` is not mutually exclusive.

8.3.3 Passive voice formation

Passive voice formation was traditionally handled in LFG using a lexical rule such as the template `PASS` provided in (8.117) taken from the XLE documentation.[24] If the verb form is a passive participle, (8.117) transforms the base active schema by rewriting relevant grammatical functions – under passive voice the active object becomes the passive subject (`(^ OBJ)-->(^ SUBJ)`), while the active subject can be handled in two ways: it can be dropped (`(^ SUBJ)-->NULL`) or become an oblique *by*-phrase (`(^ SUBJ)-->(^ OBL-AG)`).

(8.117)
```
PASS(SCHEMATA) =
{
SCHEMATA
|
SCHEMATA
(^ PARTICIPLE)=c PAST
(^ OBJ)-->(^ SUBJ)
   {
   (^ SUBJ)-->(^ OBL-AG)
   |
   (^ SUBJ)-->NULL
   }
}.
```

(8.118)
```
PASS(SCHEMATA) =
{
SCHEMATA
@(PASSIVE -)
|
SCHEMATA
@(PASSIVE-SUBC +)
(^ OBJ)-->(^ SUBJ)
   {
   (^ SUBJ)-->(^ OBL-AG)
   @(OBL-FORM-CASE OBL-AG przez acc)
   |
   (^ SUBJ)-->NULL
   ~(^ OBL-AG)
   }
}.
```

Before the introduction of off-path constraints to handle unlike category coordination, Polish LFG grammar used the template defined in (8.118) to handle passive voice formation – it is an

[24] See `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N5.2`; note that the code in (8.117) was reformatted in order to make it more readable.

adapted version of the lexical rule defined in (8.117). It additionally ensures that disjuncts are mutually exclusive: `@(PASSIVE -)` in the first disjunct introduces a statement specifying that the relevant verb form is not passive (see (8.119) for its definition), while `@(PASSIVE-SUBC +)` in the second second disjunct checks that the verb is a passive form (see (8.120)).

(8.119)
```
PASSIVE(P) = "sets PASSIVE"
(^ PASSIVE)= P.
```

(8.120)
```
PASSIVE-SUBC(P) = "checks that the value of PASSIVE is equal to P"
(^ PASSIVE)=c P.
```

Furthermore, `@(OBL-FORM-CASE OBL-AG przez acc)` in the first embedded disjunct of (8.118) ensures that the *by*-phrase is realised by a prepositional phrase where the preposition PRZEZ requires the nominal to be specified for the accusative case. The template `OBL-FORM-CASE` is defined in (8.121). Definitions of templates called in (8.121) can be found in (8.79) and (8.74), respectively.

(8.121)
```
OBL-FORM-CASE(A F OO) = "form and case for oblique arguments"
@(PFORM-SUBC-PATH (^ A) F)
@(CASE-SUBC-PATH (^ A) OO).
```

Since many converted schemata from Walenty use off-path constraints, it would not be possible to use them as an argument to a template such as (8.118) – it is not possible to use off-path constraints in XLE templates, as explained at the end of § 8.3.2.2.

For this reason, the creation of passivised versions of relevant statements is handled at the time of converting the dictionary. Note that these solutions, the lexical rule and creating passivised schemata at the time of conversion from Walenty, are nearly equivalent – the output is almost identical. There is, however, a difference in output, which gives advantage to the latter method: when creating passivised schemata during the conversion of the valence dictionary rather than using a lexical rule in XLE, constraints imposed on grammatical functions affected by passivisation are changed accordingly. The case of the object control verb ZMUSIĆ ‘force’ can serve as an illustration of this issue – the relevant schema is provided in (8.122).

(8.122)
```
subj{np(str)} + obj,controller{np(str)} + controllee{infp(_)}
```

As indicated in (8.122), the argument labelled as `obj`, the object, serves as the controller of the argument which contains an infinitival phrase (`infp(_)`) as its only possible realisation. It is the subject of this infinitival phrase which is controlled by the controller – the control equation in (8.124) formalises this observation. However, such a relation holds for active verb forms, see (8.123), but it changes accordingly with passive verb forms – the active object becomes the passive subject, as in (8.125), which is formalised in the control equation provided in (8.126).

(8.123)	kilka	takich	protokołów	zmuszono	mnie	podpisać
	few	such	report	forced.IMPS	I.ACC	sign

‘They forced me to sign a few such reports.’ (NKJP)

(8.124)
```
(^ OBJ)=(^ XCOMP SUBJ)
```

(8.125) został zmuszony podpisać lojalkę
was forced.PPAS sign loyalty oath
‘He was forced to sign a loyalty oath.’ (NKJP)

(8.126)
```
(^ SUBJ)=(^ XCOMP SUBJ)
```

Since the lexical rule only manipulates the value of the `PRED` attribute (the following equation would be used with (8.122): `(^ PRED)=‘zmusić<(^ SUBJ)(^ OBJ)(^ XCOMP)>’`), it does not affect other constraints stored in the lexical entry of the relevant verb. As a result, the lexical rule cannot change the active control equation in (8.124) to (8.126), which should be used with passive verb forms – special constraints have to be used to account for this under the lexical rule account.[25]

Such changes are captured in the process of converting schemata from Walenty to XLE lexical entries, by contrast to the use of the lexical rule. Furthermore, it is perhaps better to use the conversion method since it is possible to add passivised schemata to passive forms and base, active frames to non-passive verb forms – in this way there is no need to call a disjunctive template checking whether the relevant verb form is passive or not in order to choose the appropriate version of the schema (this happens when a lexical rule such as in (8.118) is called).

Let us see how a sample schema with a passivisable object was converted. (8.127) is the schema taken from the entry of the verb ADMINISTROWAĆ ‘administrate’. Three converted frames are produced as a result of the conversion procedure: (8.128) is the active frame, added to non-passive verb forms, while the remaining frames in (8.129) and (8.130) are only used with passive verb forms – (8.129) requires an oblique argument, while (8.130) does not include this argument (the active subject is rewritten to `NULL`).

(8.127)
```
subj{np(str)} + obj{np(inst)}
```

(8.128)
```
(^ PRED)=’administrować<(^ SUBJ)(^ OBJ)>’
(^ SUBJ PRED:
        {(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec})
(^ SUBJ CORRELATIVE)~= +
(^ OBJ CASE)=c inst (^ OBJ CORRELATIVE)~= +
```

(8.129)
```
(^ PRED)=’administrować<(^ OBL-AG)(^ SUBJ)>’
(^ OBL-AG PFORM)=c przez (^ OBL-AG CASE)=c acc
(^ OBL-AG CORRELATIVE)~= +
(^ SUBJ PRED:
        {(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec})
(^ SUBJ CORRELATIVE)~= +
```

(8.130)
```
(^ PRED)=’administrować<NULL (^ SUBJ)>’
(^ SUBJ PRED:
        {(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec})
(^ SUBJ CORRELATIVE)~= +
```

[25] This is because lexical rules simply match and replace relevant strings in `PRED` attribute. See the relevant fragment of XLE documentation: `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N5.2` – it also discusses additional constraints needed to account for such phenomena.

8.3.4 Reducing frames

Schemata described in Walenty are maximal – the number of arguments may be reduced because sometimes not all arguments are realised when the verb is used. This is illustrated below: the schema for the verb DOWODZIĆ ‘command’ provided in (8.131) contains two arguments – a subject and a passivisable object. (8.132) shows that both arguments can be realised lexically, but they may also be omitted.[26]

(8.131) `subj{np(str)} + obj{np(inst)}`

(8.132)	(Mój	ojciec)	dowodził	(siłami	republikańskimi)	na	Północy.
	my.NOM	father.NOM	commanded	forces.INST	republican.INST	in	north

‘My father commanded republican forces in the north.’ (NKJP)

There are two main ways of handling this phenomenon in an implemented grammar. One way is to assume that the argument is present but it is not realised lexically – in this way the argument is represented syntactically, the relevant grammatical function attribute is present, but its value is ‘PRO’ – see the f-structure in (8.133), which corresponds to (8.132) with the object dropped. The alternative approach is to assume that the relevant argument is removed, that it is not present in the f-structure of the verb – this solution involves the creation of reduced frames, which have fewer arguments than the maximal frame, as in (8.134), where the object is removed from the list in PRED attribute.

(8.133)
$$\left[\begin{array}{ll}
\text{PRED} & \text{‘COMMAND}\langle\boxed{1},\boxed{2}\rangle\text{’} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{‘FATHER’} \\ \text{CASE} & \text{NOM} \\ \text{NUM} & \text{SG} \\ \text{PERS} & 3 \\ \text{ADJ} & \left\{\left[\begin{array}{ll} \text{PRED} & \text{‘MY’} \end{array}\right]\right\} \end{array}\right] \\
\text{OBJ} & \boxed{2}\left[\begin{array}{ll} \text{PRED} & \text{‘PRO’} \\ \text{CASE} & \text{INST} \end{array}\right] \\
\text{TENSE} & \text{PAST}
\end{array}\right]$$

(8.134)
$$\left[\begin{array}{ll}
\text{PRED} & \text{‘COMMAND}\langle\boxed{1}\rangle\text{’} \\
\text{SUBJ} & \boxed{1}\left[\begin{array}{ll} \text{PRED} & \text{‘FATHER’} \\ \text{CASE} & \text{NOM} \\ \text{NUM} & \text{SG} \\ \text{PERS} & 3 \\ \text{ADJ} & \left\{\left[\begin{array}{ll} \text{PRED} & \text{‘MY’} \end{array}\right]\right\} \end{array}\right] \\
\text{TENSE} & \text{PAST}
\end{array}\right]$$

The implemented grammar discussed in this work uses a hybrid solution: when there is evidence that the relevant argument is active syntactically despite the fact that it has no lexical realisation, it is treated as an implicit argument (‘PRO’, as in (8.133)). By contrast, if there is no such evidence, the argument in question is removed – it is not present in the f-structure, the frame of the relevant verb is reduced, as in (8.134).

There are two situations when implicit arguments are used in the implementation of the Polish LFG: the subject and any argument which acts as the controller. The reason for using implicit arguments (whose value of PRED is ‘PRO’, as explained earlier) is that these arguments are required by syntax: in Polish it is the subject which binds the SIEBIE anaphor (see (8.135))

[26] The bracketing in (8.132) assumes that *na Północy* is a dependent of the verb, but it could also be analysed as a dependent of *siłami*.

and controls participles (see (8.136)) – if it was removed from the schema, sentences with no lexical subject could not be parsed (because the subject position would have no value, resulting in incompleteness, see § 2.2.2) and would be expected to be ungrammatical, counter to fact:

(8.135) (Antek$_a$) opowiedział Erykowi$_e$ o sobie$_{a/*e}$.
Antek.NOM told Eryk.DAT about SELF
'(Antek) told Eryk about himself.'

(8.136) Wychodząc$_{a/*e}$, (Antek$_a$) pocieszał Eryka$_e$.
returning Antek.NOM comforted Eryk.ACC
'Leaving, (Antek) was comforting Eryk.'

The second group of arguments which may be implicit are controllers of infinitives and predicative complements – the reason for having implicit arguments is identical to the one used for controlling participles: the subject of the controlled element is structure-shared with the controller, so the controller must be present in the f-structure. In this case, however, the controller may be different than the subject, see the examples below:

(8.137) Dowódca kazał uciekać.
commander.NOM ordered escape.INF
'The commanding officer ordered to run away'.

(8.138) Antek zawsze uczyni (Eryka) szczęśliwym.
Antek always make Eryk happy
'Antek will always make (Eryk) happy.'

According to the schema in (8.139), the controller of the infinitival complement of the verb KAZAĆ 'order' in (8.137) is the dative nominal. By contrast, the schema in (8.140) specifies the passivisable object marked for structural case as the controller of the predicative complement of the verb UCZYNIĆ 'make' in (8.138).

(8.139) `subj{np(str)} + controller{np(dat)} + controllee{cp(żeby); infp(_)}`

(8.140) `subj{np(str)} + obj,controller{np(str)} + controllee{adjp(inst)}`

In (8.137)–(8.138) ((8.137) is a modified version of (8.16)) controllers may have no lexical realisation, they are nevertheless required by syntax (controlled phrases must have controllers), so they are represented in the f-structure representation as implicit arguments ('PRO' is the value of their PRED attribute) – the f-structure in (8.141) corresponds to (8.137), while (8.142)[27] provides a representation of (8.138) without the lexical object:

[27] Note that one of the arguments of MAKE in (8.142), namely $\boxed{2}$, which corresponds to OBJ, is placed outside the angle brackets – this represents the fact that it is considered to be a non-semantic argument. Such arguments include raised arguments ('John' in 'John seems to like asparagus.') and, which is the practice in ParGram English grammar, the argument which controls the predicative argument, which is the case in (8.142). Besides, XC-PRED is used in (8.142) as an abbreviation of XCOMP-PRED for reasons of space.

(8.141)
$$\left[\begin{array}{lll} \text{PRED} & & \text{`ORDER}\langle\boxed{1},\boxed{2},\boxed{3}\rangle\text{'} \\ \text{SUBJ} & \boxed{1} & \left[\begin{array}{ll}\text{PRED} & \text{`COMMANDER'}\\ \text{CASE} & \text{NOM}\end{array}\right] \\ \text{OBJ}_{\theta} & \boxed{2} & \left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{DAT}\end{array}\right] \\ \text{XCOMP} & \boxed{3} & \left[\begin{array}{ll}\text{PRED} & \text{`RUN\_\_AWAY}\langle\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{2}\end{array}\right] \end{array}\right]$$

(8.142)
$$\left[\begin{array}{lll} \text{PRED} & & \text{`MAKE}\langle\boxed{1},\boxed{3}\rangle\boxed{2}\text{'} \\ \text{SUBJ} & \boxed{1} & \left[\begin{array}{ll}\text{PRED} & \text{`ANTEK'}\\ \text{CASE} & \text{NOM}\end{array}\right] \\ \text{OBJ} & \boxed{2} & \left[\begin{array}{ll}\text{PRED} & \text{`PRO'}\\ \text{CASE} & \text{ACC}\end{array}\right] \\ \text{XC-PRED} & \boxed{3} & \left[\begin{array}{ll}\text{PRED} & \text{`HAPPY}\langle\boxed{2}\rangle\text{'}\\ \text{SUBJ} & \boxed{2}\end{array}\right] \end{array}\right]$$

Let us now consider how the reduction of frames is handled during the process of converting frames from Walenty:

(8.143) each argument (position) from the maximal schema is assigned a grammatical function,

(8.144) converted arguments are divided into obligatory arguments (which may not be reduced) and optional arguments (which may undergo reduction),

(8.145) a powerset[28] of the list of optional arguments is calculated (the function used here operates on lists and returns a list of lists (instead of a set of sets)),

(8.146) for each element of the powerset of optional arguments, a new list is created by joining the list of obligatory elements with the relevant list of optional arguments; a list of lists of arguments is the result,

(8.147) each obtained list of arguments is converted to a separate frame as discussed earlier in this section.

Obligatory arguments include: the subject, lexicalised arguments (`fixed`, `lexnp`, `preplexnp`, `comprepnp`) and the reflexive marker (`refl`).[29] The reason for treating lexicalised arguments as obligatory is that they influence the semantics of the verb used in the schema and therefore they should not be removed – the verb without such an argument has a meaning which is no longer phraseological.

Furthermore, it is worth noting that controllers are not classified as obligatory arguments: they are only obligatory in the situation when the corresponding controllee is present – if the argument labelled as `controllee` was reduced, the `controller` is no longer obligatory and it is treated as a plain argument with no control relation. If it happens to be a subject or a lexicalised argument, it cannot be reduced (because, as explained above, subjects and lexicalised arguments do not undergo reduction), but the remaining arguments may be reduced.

Let us consider an example where there are two obligatory arguments and three optional ones – (8.148) is a schema taken from the entry of the verb WYBIĆ ‘knock (sth) out’: it takes 5 arguments, two of which are obligatory (subject, `subj{np(str)}`, and the lexicalised prepositional argument, `preplexnp(z,gen,_,'głowa',atr)`), while the remaining three arguments are optional.

[28] Powerset is the set of all subsets of a given set, including the empty set and the input set itself.

[29] The inherent reflexive marker is not listed here because it is a part of the lemma of the verb (see (8.1) for a sample schema) – it cannot be reduced, because this would affect the semantics.

(8.148) `subj{np(str)} + {np(dat)} + {np(inst)} + {cp(że)}`
`+ {preplexnp(z,gen,_,’głowa’,atr)}`

The first step is (8.143), whereby grammatical functions are assigned to arguments from (8.148). This is done according to mapping rules presented in § 8.3.1.1 and, if applicable, using the ranking from § 8.3.1.2. The table in (8.149) shows the result of this procedure for (8.148):

(8.149)	`subj{np(str)}`	`{np(dat)}`	`{np(inst)}`	`{cp(że)}`	`{preplexnp(z,gen,_,`
					`’głowa’,atr)}`

	`SUBJ`	`OBJ-TH`	`OBL-INST`	`COMP`	`OBL`

The second step is (8.144), where arguments are divided into obligatory and optional. This is achieved in (8.150) by assigning variables to grammatical functions, where `o1` is the first obligatory argument and `1` is the first optional argument:

(8.150)	`SUBJ`	`OBJ-TH`	`OBL-INST`	`COMP`	`OBL`
	`o1`	`1`	`2`	`3`	`o2`

The third step is (8.145) – it involves calculating the powerset of the list of optional arguments from (8.150). There are three optional arguments, the powerset of `{1, 2, 3}` is shown in (8.151) (as mentioned in (8.145) the result is a list):

(8.151) `[[], [1], [2], [3], [1, 2], [1, 3], [2, 3], [1, 2, 3]]`

The last, fourth step is (8.146): the list of obligatory arguments (there are two in (8.150), so it is a two-element list: `[o1, o2]`) is added to each element of (8.151).

(8.152) `[[o1, o2], [o1, o2, 1], [o1, o2, 2], [o1, o2, 3], [o1, o2, 1, 2],`
`[o1, o2, 1, 3], [o1, o2, 2, 3], [o1, o2, 1, 2, 3]]`

In (8.152) the first element contains only obligatory arguments (`[o1, o2]`), while the last one contains all arguments (`[o1, o2, 1, 2, 3]`). Each of the elements between them contains obligatory arguments and appropriate optional arguments.

Once the variables are replaced with relevant grammatical functions according to the mapping in (8.150), reduced frames provided in (8.153)–(8.160) below correspond to relevant elements of the list given in (8.152): the first element of (8.152) is the list which contains only obligatory arguments, as in (8.153), the second element contains obligatory arguments and the first optional one, see (8.154), and so on. After taking reduced frames into consideration, there are 8 elements in (8.152), so 8 frames were produced as a result of the conversion of the schema for the verb WYBIĆ provided in (8.148) – these are shown in (8.153)–(8.160).

(8.153) `(^ PRED)=’wybić<(^ SUBJ)(^ OBL)>’`

(8.154) `(^ PRED)=’wybić<(^ SUBJ)(^ OBL)(^ OBJ-TH)>’`

(8.155) `(^ PRED)=’wybić<(^ SUBJ)(^ OBL)(^ OBL-INST)>’`

(8.156) `(^ PRED)=’wybić<(^ SUBJ)(^ OBL)(^ COMP)>’`

(8.157) `(^ PRED)=’wybić<(^ SUBJ)(^ OBL)(^ OBJ-TH)(^ OBL-INST)>’`

```
(8.158)  (^ PRED)='wybić<(^ SUBJ)(^ OBL)(^ OBJ-TH)(^ COMP)>'
```

```
(8.159)  (^ PRED)='wybić<(^ SUBJ)(^ OBL)(^ OBL-INST)(^ COMP)>'
```

```
(8.160)  (^ PRED)='wybić<(^ SUBJ)(^ OBL)(^ OBJ-TH)(^ OBL-INST)(^ COMP)>'
```

Constraints imposed on relevant arguments of (8.148), the unreduced schema, are provided in (8.161):

```
(8.161)  (^ PRED)='wybić<(^ SUBJ)(^ OBL)(^ OBJ-TH)(^ OBL-INST)(^ COMP)>'
         (^ SUBJ PRED:
                {((SUBJ <-) CHECK _CAT)=c ger (<- CASE)=c gen
                | ((SUBJ <-) CHECK _CAT) $c {pact ppas inf}
                | ~[((SUBJ <-) CHECK _CAT) $c {pact ppas inf}]
                {(<- CASE)=c nom | (<- CASE)=c acc (<- ACM)=c rec}})
         (^ SUBJ CORRELATIVE)~= +
         (^ OBJ-TH CASE)=c dat (^ OBJ-TH CORRELATIVE)~= +
         (^ OBL-INST CASE)=c inst (^ OBL-INST CORRELATIVE)~= +
         (^ COMP COMP-FORM)=c że
         (^ OBL PFORM)=c z (^ OBL CASE)=c gen (^ OBL CORRELATIVE)~= +
         (^ OBL PRED FN)=c głowa
```

When constraints are imposed for reduced frames from the list in (8.152), an appropriate subset of constraints shown in (8.161) is used.

Finally, it is important to note that when a schema is reduced in the process of conversion, the original grammatical function assignment is reviewed for obliques – if the full schema has 4 obliques (OBL, OBL2, OBL3 and OBL4) but the reduced schema has only one, this oblique corresponds to OBL grammatical function regardless of which oblique grammatical function it had in the full, unreduced schema.

8.4 Summary

This chapter discussed the formalism used in the Walenty valence dictionary of Polish (§ 8.2) and how the information coming from this source is converted to constraints used by the LFG grammar of Polish (§ 8.3). This involved the discussion of grammatical function assignment (§ 8.3.1), also for arguments which may be realised as a coordination of unlike categories. Subsequently, the procedure used for imposing constraints was presented (§ 8.3.2), including the formalisation of relevant constraints using off-path constraints for unlike category coordination and taking into consideration interactions with structural case assignment, which are very important in Polish. Finally, issues such as the formation of passive schemata and the reduction of maximal schemata were discussed.

Chapter 9

Implementation of unlike coordination

9.1 Introduction

This section presents an implementation of the coordination phenomena analysed in the theoretical part: § 9.2, together with relevant parts of ch. 8, corresponds to ch. 4, where coordination of unlike categories is discussed, while § 9.3 presents the implementation of lexico-semantic coordination discussed in ch. 5.

9.2 Coordination of unlike categories

The previous chapter presented how valence entries from Walenty are converted to LFG constraints, providing an account of unlike category coordination in § 8.3.2.2 by showing converted valence entries in the lexicon. One thing must be added in order to make the account of coordination of unlike categories complete – appropriate c-structure rules must be provided, so as to make it possible to coordinate phrases which are different with respect to the category they represent. When this is done, the lexical entry of the head can restrict the range of categories which can correspond to the relevant argument using lexicalised constraints which ensure that each conjunct satisfies relevant requirements.

This section is divided into two parts: § 9.2.1 provides the basic rule used for handling unlike category coordination, while § 9.2.2 provides the specialised rule designed for cases where one of the conjuncts under unlike category coordination is an infinitive, so control into selected conjuncts is involved (see § 4.7 for discussion).

9.2.1 Basic rule

The first version of the basic rule used for unlike category coordination is provided in (9.1):

(9.1)
```
unlike_coord -->
anyCAT: @inSET;
CONJ
anyCAT: @inSET.
```

(9.2)

```
anyCAT = {NP|PP|CP|AP|ADVP|XP}.
```

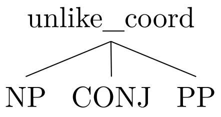
(9.3)

```
inSET = "adds an element to a set"
! $ ^.
```

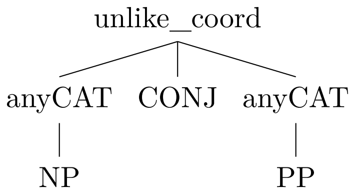
(9.1) rewrites to a coordination of the `anyCAT` category. According to the definition provided in (9.2), `anyCAT` rewrites to a disjunction of 6 different phrases: `NP` (nominal), `PP` (prepositional), `CP` (complementiser), `AP` (adjectival), `ADVP` (adverbial) and `XP` (with specific semantics, see § 8.2.6).

It is worth noting that unlike `unlike_coord` defined in (9.1), `anyCAT` it is not a plain category but a metacategory.[1] While plain categories use `-->` as the rewrite symbol and both sides of the rule are shown in the tree, metacategories use `=` and it is only the right-hand side of such a rule that is shown in the tree – the metacategory symbol is not represented in the tree. This difference is illustrated in (9.4)–(9.5), assuming that the two instances of `anyCAT` in (9.1) rewrite to `NP` and `PP`, respectively.

(9.4)



(9.5)



(9.4) is the tree built by rules (9.1)–(9.2), where (9.1) is a plain category, which is represented in the tree, while (9.2) is a metacategory, so only its right-hand side is shown in the tree (`NP` and `PP`, respectively, as explained above). By contrast, (9.5) is the tree which would be built if `anyCAT` defined in (9.2) was a plain category (`-->` instead of `=` as the rewrite symbol) – in such a situation the plain category is represented in the tree.

However, (9.1) was the first, preliminary version of the rule handling coordination of unlike categories. This is because (9.1) allows for coordination of any categories, possibly identical, while they should be distinct by definition. The enhanced version of (9.1) is provided in (9.6):

(9.6)

```
unlike_coord -->
anyCAT: @inSET
        (^ CHECK _CAT)=%C1;
CONJ
anyCAT: @inSET
        (^ CHECK _CAT)=%C2;
e: %C1~=%C2.
```

(9.6) ensures that `anyCAT` rewrites to non-identical categories in particular conjuncts by assigning the value of the attribute `_CAT` hosting the category of the head of the relevant phrase to local variables: `%C1` for the first conjunct and `%C2` for the other. Finally, a statement ensuring that the values of these variables are different (that the categories are unlike) is attached to `e` – the category which is present in rules (and to which constraints can be attached), but it is not represented in the tree.[2]

[1] This device was first used in § 4.7.2.3, see the discussion of (4.105)–(4.107). Also, see the relevant fragment of XLE documentation: `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N3.1`.

[2] See the relevant part of the XLE documentation: `http://www2.parc.com/isl/groups/nltt/xle/doc/notations.html#N1.1`.

Unfortunately, the solution presented in (9.6) has certain disadvantages: the `_CAT` attribute is introduced by the (semantic) head of the phrase, so non-semantic prepositional phrases (see § 2.3 for discussion) have `_CAT` value contributed by the nominal, so that it is not possible to distinguish a nominal phrase from a non-semantic prepositional phrase.

An alternative would be to enumerate all 2-element combinations of unlike category conjuncts defined in (9.2). The result is 30 pairs (6 categories × 5 possible combinations) listed below:

(9.7) (NP, PP)
(9.8) (NP, CP)
(9.9) (NP, AP)
(9.10) (NP, ADVP)
(9.11) (NP, XP)
(9.12) (PP, NP)
(9.13) (PP, CP)
(9.14) (PP, AP)
(9.15) (PP, ADVP)
(9.16) (PP, XP)
(9.17) (CP, NP)
(9.18) (CP, PP)
(9.19) (CP, AP)
(9.20) (CP, ADVP)
(9.21) (CP, XP)
(9.22) (AP, NP)
(9.23) (AP, PP)
(9.24) (AP, CP)
(9.25) (AP, ADVP)
(9.26) (AP, XP)
(9.27) (ADVP, NP)
(9.28) (ADVP, PP)
(9.29) (ADVP, CP)
(9.30) (ADVP, AP)
(9.31) (ADVP, XP)
(9.32) (XP, NP)
(9.33) (XP, PP)
(9.34) (XP, CP)
(9.35) (XP, AP)
(9.36) (XP, ADVP)

The next step is to create a rule based on (9.1), where each disjunct of the new rule corresponds to the right-hand side of (9.1) where respective `anyCAT` conjuncts are replaced with relevant elements of the given pair – (9.37) demonstrates the first two disjuncts of such a rule, which correspond to the first two pairs of unlike categories listed above: (9.7) and (9.8).

(9.37)
```
unlike_coord -->
{
NP: @inSET;
CONJ
PP: @inSET
|
NP: @inSET;
CONJ
CP: @inSET
|
...
}.
```

Though this approach successfully solves the problem with (9.6) described above, it lacks the elegance of (9.6). It seems, however, that a working solution is better than an elegant solution that does not work as desired.

To allow for unlike category coordination of more than two elements, a new list of tuples containing unlike categories must be generated – for example, to allow unlike category coordination of three elements (where each element is different), such a list would contain 120 3-tuples.

9.2.2 Rule for control into selected conjuncts

This subsection presents the c-structure rule needed to account for control into selected conjuncts, which was described in § 4.7. The rule in (9.38) corresponds to (4.105), the rule proposed in the theoretical analysis in § 4.7.2.3.

(9.38)

```
unlike_control -->
{
INFP: @inSET
      (! CONTROLLER)=(! SUBJ);
CONJ
NP: @inSET
|
NP: @inSET;
CONJ
INFP: @inSET
      (! CONTROLLER)=(! SUBJ)
}.
```

It is worth noting that (9.38) is slightly different from (4.105): while (4.105) allows only one ordering of conjuncts (`INFP` as the first conjunct and `NP` as the second), (9.38) also allows for the ordering where conjuncts are switched (`NP` comes first and `INFP` follows). Furthermore, (9.38) is formalised as a plain category while (4.105) is a metacategory (for the sake of easier presentation, as explained in fn. 33 in § 4.7.2.3).

9.3 Coordination of unlike grammatical functions

This section presents the XLE implementation of lexico-semantic coordination in the Polish LFG grammar: § 9.3.1 presents the top-level rule, § 9.3.2 discusses rules handling monoclausal coordination, while § 9.3.3 is devoted to multiclausal coordination. Finally, § 9.3.4 provides a discussion of rules rewriting to particular categories.

9.3.1 Top-level rule

The top-level rule handling lexico-semantic coordination provided in (9.39) corresponds to the rule in (5.229):

(9.39)

```
"two-way lex-sem: monoclausal (XPlxm) vs multiclausal (XPlxb)"
anyLEXSEM -->
{
"monoclausal: any TYPE"
anyXPlxm: ! $ (^ UDF)
|
"multiclausal: only interrogative"
XPlxb[int]
}.
```

(9.39) consists of two disjuncts, which handle different representational variants of lexico-semantic coordination: monoclausal (see § 5.5.1 for initial discussion and § 5.7.1 for the final, improved analysis) and multiclausal (§ 5.5.2 provides discussion and the first analysis, while § 5.7.2 contains the final analysis) – the implementation of rules required to obtain these two kinds of representation is provided in § 9.3.2 and § 9.3.3, respectively.

9.3.2 Monoclausal coordination

Monoclausal lexico-semantic coordination is handled by the rule `anyXPlxm` defined in (9.40): it rewrites to `XPlxm` of all types allowed by (9.41)[3] – these include interrogative pronouns (`int`), *n*-words (`neg`), -*kolwiek* pronouns (`any`) and pronouns expressing the universal quantifier (`all`). As its counterpart in (5.212), `XPlxm` is a complex category and lexico-semantic conjuncts in its right-hand side, `XPlxmC` defined in (9.42), can be rewritten in two ways (like its counterpart in (5.213)): the first disjunct in (9.42) rewrites to simple phrases (`XPextr`, which are not instances of lexico-semantic coordination – see § 5.5.1) of an appropriate `_type`, while the second disjunct is `XPlxm`, which handles embedded lexico-semantic coordination of the given `_type` (as discussed in § 5.6.4, see (5.206)).

(9.40)
```
"monoclausal lex-sem of any TYPE"
anyXPlxm =
{ XPlxm[int] | XPlxm[neg] | XPlxm[any] | XPlxm[all] }.
```

(9.41)
```
"monoclausal lex-sem"
XPlxm[_type $ {int neg any all}] -->
XPlxmC[_type]: @inSET;
[COMMA
XPlxmC[_type]: @inSET]*
CONJ
XPlxmC[_type]: @inSET.
```

(9.42)
```
"monoclausal: particular conjuncts"
XPlxmC[_type $ {int neg any all}] -->
{
"no embedding"
XPextr[_type]
|
"embedded"
XPlxm[_type]
}.
```

Conjuncts in (9.41) are annotated using the `inSET` template defined in (9.3), repeated in (9.43), which adds them to a set according to the standard LFG analysis of coordination.

(9.43)
```
inSET = "adds an element to a set"
! $ ^.
```

[3] Note that the list of types in (9.41) could be extended so as to include the types discussed in § 5.8.

As mentioned above, in the first disjunct of (9.42) particular conjuncts rewrite to simple (non-coordinate) phrases of the relevant type (corresponding to the value of the variable `_type`) – see the rewrite rule for `XPextr` provided in (9.44):

(9.44)
```
XPextr[_type $ {int neg any all}] -->
{
NP[_type]: ^=!
           ((UDF ^) XPATH {SUBJ|OBJ|OBJ-TH|OBL-STR|OBL-GEN|OBL-INST
                          |XCOMP-PRED})=!
|
PP[_type]: {
            ^=!
            ((UDF ^) XPATH (XCOMP-PRED) {OBL|OBL2|OBL-AG|OBL-COMPAR
                                        |XCOMP-PRED})=!
            |
            ^=!
            ! $ ((UDF ^) XPATH ADJUNCT)
            }
|
ADVP[_type]: ^=!
             ! $ ((UDF ^) XPATH ADJUNCT)
|
AP[_type]: {
           "as a modifier"
           ^=!
           ! $ (%AGRCTR ADJUNCT)
           ((UDF ^) XPATH {SUBJ|OBJ|OBJ-TH|OBL-STR|OBL-GEN|OBL-INST
                          |XCOMP-PRED})=%AGRCTR
           @(AGR-ATTR-PATHS %AGRCTR !)
           _type =c int
           |
           "as a predicative complement"
           ((UDF ^) XPATH XCOMP-PRED)=!
           }
}.
```

(9.45)
```
XPATH = {COMP|XCOMP}*.
```

(9.44) is an implementation counterpart of (5.209); the difference is that specific grammatical functions are provided in (9.44) instead of the underspecified GF$^+$ used in (5.209). This was done for practical reasons, namely in order to reduce overgeneration: by reducing the amount of possible analyses (here: grammatical function assignments), it is possible to reduce the amount of resources (such as memory and time) needed to produce the relevant structures.

The second disjunct of (9.42) accounts for constructions where lexico-semantic coordination is embedded: using (9.41), each conjunct can be a non-coordinate lexico-semantic conjunct (`XPextr`) of some `_type` or an instance of lexico-semantic coordination (`XPlxm`) of the same type, which makes it possible to handle embedding of lexico-semantic coordination in any of the conjuncts (see § 5.6.4 and § 5.7.1.2 for discussion and examples of embedding).

9.3.3 Multiclausal coordination

Let us now proceed to multiclausal lexico-semantic coordination structures. By contrast with monoclausal structures, multiclausal structures can only be interrogative,[4] which is expressed by the type of the `XPlxb` complex category used in (9.39): `XPlxb[int]`.

As in § 5.7.2, the theoretical section devoted to the formalisation of multiclausal lexico-semantic coordination, two variants are presented here: the first can produce a representation which involves more than two clauses (when there are at least three lexico-semantic conjuncts), while the second one always yields a representation with exactly two clauses (one for the *yes/no* question particle CZY, while the other contains the remaining lexico-semantic conjuncts).

9.3.3.1 Multiclausal representation

This subsection presents the implementation corresponding to the formalisation presented in § 5.7.2.1. For the definition of `XPlxb` as the implementation counterpart of (5.230), see (9.46):

(9.46)
```
"multiclausal lex-sem"
XPlxb[_type $ {int}] -->
QUBbicl[_type]: @inSET;
[COMMA
XPextrbicl[_type]: @inSET]*
CONJ
XPextrbicl[_type]: @inSET.
```

(9.46) specifies the possible values of `_type`: it can only be `int`. According to this rule, the `XPlxb` category rewrites to a coordinate phrase where the first conjunct is `QUBbicl` (defined in (9.47)), while the remaining conjuncts are `XPextrbicl` (see (9.55) for its definition); all conjuncts are added to a set using `inSET` template (see (9.43)). The last conjunct is preceded by a conjunction (`CONJ`), while the optional preceding conjuncts are added with a comma (`COMMA`) in front of each of them.

Let us start with the first conjunct of (9.46), `QUBbicl[int]`, which corresponds to the *yes/no* question particle CZY. The rule `QUBbicl` defined in (9.47) is an implementation counterpart of (5.175): it rewrites to `QUB[int]` bearing the co-head annotation (`^=!`) and a call to the template `PRODROP-LEXSEM-BICL` defined in (9.48).

[4] As discussed in § 5.8.4, it seems that lexico-semantic coordination of relative pronouns would also require multiclausal representation. However, to ensure parallelism with the analysis presented in the theoretical section, this possibility is not included in the implementation. To take this into account, the range of types allowed by (9.56) would have to be extended to include relative pronouns (`rel`) and the first conjunct should be either `QUBbicl` (for interrogative items) or `XPextrbicl` (for relative items).

(9.47)
```
QUBbicl[int] -->
QUB[int]: ^=!
  @(PRODROP-LEXSEM-BICL).
```

(9.48)
```
PRODROP-LEXSEM-BICL = "optional implicit argument for
multiclausal lex-sem"
"implicit SUBJ: nominative"
{@(PRODROP-LEXSEM-BICL-GF SUBJ nom)}
"implicit OBJ: accusative, genitive, instrumental"
{@(PRODROP-LEXSEM-BICL-GF OBJ %C)
%C $ {acc gen inst}}
"implicit OBJ-TH: dative"
{@(PRODROP-LEXSEM-BICL-GF OBJ-TH dat)}
"implicit OBL-STR: accusative, genitive (structural)"
{@(PRODROP-LEXSEM-BICL-GF OBL-STR %C)
%C $ {acc gen}}
"implicit OBL-INST: instrumental"
{@(PRODROP-LEXSEM-BICL-GF OBL-INST inst)}
"implicit OBL-GEN: genitive (lexical)"
{@(PRODROP-LEXSEM-BICL-GF OBL-GEN gen)}.
```

(9.49)
```
PRODROP-LEXSEM-BICL-GF(GF C) = "implicit GF (with CASE C)"
(^ GF PRED)='pro'
@(CASE-PATH (^ GF) C).
```

As formalised in (9.48), the call to the template PRODROP-LEXSEM-BICL (a counterpart of (5.172)) can optionally introduce implicit arguments under multiclausal lexico-semantic coordination, ensuring appropriate argument saturation (see § 5.5.3 for discussion) in order not to violate the completeness principle. The template PRODROP-LEXSEM-BICL makes optional (each statement is enclosed in curly brackets) calls to the template PRODROP-LEXSEM-BICL-GF defined in (9.49), providing two parameters: the grammatical function and the possible values of case. Let us consider some examples: the first call to (9.49) in (9.48) introduces an implicit subject (SUBJ is provided as the first parameter of the call to PRODROP-LEXSEM-BICL-GF) marked for the nominative case (nom as the second parameter). By contrast, the second call introduces a passivisable object (OBJ) marked for one of the cases to which %C variable can rewrite: the statement %C $ {acc gen inst} assigns the value of %C to a member of the three-element set, which is equivalent to a three-way disjunction (acc or gen or inst). Instead of calling PRODROP-LEXSEM-BICL-GF three times with different values of case, one call is used, where the value of case is a variable whose value belongs to the relevant set.

Since calls to PRODROP-LEXSEM-BICL are placed in the rewrite rules of all conjuncts of (9.46) (see (9.47) and (9.55)), relevant implicit arguments are added independently in all clauses produced as a result of multiclausal representation. Finally, note that the call to this template is not placed in the lexical entry of CZY (see (9.50)) – this is because it only applies under multiclausal lexico-semantic coordination rather than in every context where the *yes/no* question particle appears (the same applies to XPextrbicl defined in (9.57)).

Let us now proceed to constraints introduced by the lexical entry of the *yes/no* question particle CZY:

```
(9.50)    czy QUB[int] * @(QUB-CZY).

(9.51)    QUB-CZY = "marker: interrogative (CZY, CZYŻ, CZYŻBY, AZALI, AZALIŻ, LI)"
          @(CZY).

(9.52)    CZY = "interrogative yes/no marker CZY"
          @(CL-TYPE int)
          "NOTE: no multiple wh-questions with CZY"
          @(TYPE-NEQ-PATH (^ GF+) int).

(9.53)    CL-TYPE(CL) = "sets clause type"
          (^ CLAUSE-TYPE)= CL.

(9.54)    TYPE-NEQ-PATH(PATH T) = "checks that TYPE in PATH is not equal to T"
          (PATH TYPE)~= T.
```

The lexical entry of CZY provided in (9.50) contains a call to the template QUB-CZY (see (9.51)), which in turn calls the template CZY defined in (9.52). (9.52) is the lexicalised template for the interrogative particle: it calls the template CL-TYPE defined in (9.53) with the parameter int, introducing an annotation which states that the relevant clause is interrogative ((^ CLAUSE-TYPE)= int). Moreover, it ensures that there are no other interrogative items in the same clause (locally): a call to the template TYPE-NEQ-PATH is used for this purpose (see (9.54) for its definition).

Finally, let us now proceed to the remaining conjuncts in (9.46), namely to the category XPextrbicl (a counterpart of (5.232), whose definition is provided in (9.55)).

```
(9.55)    XPextrbicl[_type $ {int}] -->
          XPextr[_type]:
          ! $ (^ UDF)
          @(PRODROP-LEXSEM-BICL).
```

It rewrites to XPextr, which in turn, according to its definition in (9.44), rewrites to a simple, non-coordinate lexico-semantic phrase.

As explained in detail in § 5.7.2.1, this implementation of multiclausal lexico-semantic coordination produces a representation where each conjunct is put in a separate clause, so there are as many clauses as there are conjuncts: two for two (see (5.234), which corresponds to (5.233)), three for three (as in (5.239) for (5.238)) and so on.

9.3.3.2 Biclausal representation

This subsection presents the implementation corresponding to the formalisation presented in § 5.7.2.2. For the definition of XP1xb, the implementation counterpart of (5.247), see (9.56):

(9.56)
```
"biclausal lex-sem"
XPlxb[_type $ {int}] -->
QUBbicl[_type]: @inSET;
{
"genuine conjunction between CZY and wh-phrase"
CONJ
|
"comma between CZY and wh-phrase: conjunction from the second clause
structure-shared with the verbal coordination"
COMMA: (^ $ UDF $ COORD-FORM)=(^ COORD-FORM);
}
XPextrbicl[_type]: @inSET.
```

(9.56) is a modified version of (9.46) in § 9.3.3.1 – in the same way as the latter, (9.56) restricts the possible values of `_type` to `int`. However, according to (9.56), the `XPlxb` category always has exactly two conjuncts, unlike (9.46), which allows at least two conjuncts – (9.56) rewrites to a coordinate phrase where the first conjunct is `QUBbicl` (defined in (9.47)), while the second conjunct is `XPextrbicl` (see (9.57) for its definition, note that it is different than the one used by (9.46), defined in (9.55)); both conjuncts are added to a set using `inSET` template (see (9.43)). These conjuncts are joined using a conjunction (first disjunct in (9.56): `CONJ`) or a comma (second disjunct: `COMMA`). The annotation accompanying the comma in (9.56) is used to structure-share the conjunction used in the second conjunct, which is expected to feature embedded monoclausal lexico-semantic coordination (for a detailed explanation, see (5.244) and the accompanying discussion; the relevant rule is provided in (5.247)).

(9.57)
```
XPextrbicl[_type $ {int}] -->
{
XPextr[_type]:
! $ (^ UDF)
@(PRODROP-LEXSEM-BICL)
|
XPlxm[_type]:
! $ (^ UDF)
@(PRODROP-LEXSEM-BICL)
}.
```

Let us now proceed to the second conjunct in (9.56), namely to the category `XPextrbicl` (a counterpart of (5.243), whose definition is provided in (9.57)). Unlike in (9.55), it has two disjuncts: the first one is `XPextr` defined in (9.44) – this disjunct is used for multiclausal coordination where the second conjunct is a single, non-coordinate lexico-semantic phrase. By contrast, the second disjunct of (9.57) contains `XPlxm`, a monoclausal lexico-semantic coordinate phrase whose definition can be found in (9.41).

As a result of such a definition of (9.57), multiclausal structures are always biclausal structures where the first clause contains the *yes/no* question particle CZY, while the second clause contains the remaining conjuncts – the former are analysed as an instance of monoclausal lexico-semantic coordination (see the discussion following (5.237) in § 5.7.2). While the implementation in § 9.3.3.1 produces a representation involving three clauses for (5.238) (see (5.239)), the implementation presented here produces (5.244), which involves two clauses, as described above.

9.3.4 Rules for conjuncts of particular type

Finally, rules for rewriting relevant complex categories to corresponding plain categories are provided below in (9.58)–(9.61). These rules are used by XPextr defined in (9.44).

```
(9.58)  NP[_type $ {int rel neg any all}] -->
        NP: ^=!
            (! GF* (POSS))=%L
            @(TYPE-SUBC-PATH %L _type).

(9.59)  PP[_type $ {int rel neg any all}] -->
        PP: ^=!
            (! (OBJ) (ADJUNCT $) (POSS))=%L
            @(TYPE-SUBC-PATH %L _type).

(9.60)  ADVP[_type $ {int rel neg any all}] -->
        ADVP: ^=!
              !=%L
            @(TYPE-SUBC-PATH %L _type).

(9.61)  AP[_type $ {int rel neg any all}] -->
        AP: ^=!
            !=%L
            @(TYPE-SUBC-PATH %L _type).

(9.62)  GF = {SUBJ|OBL|OBL-AG|OBJ|OBJ-TH|OBL-STR|OBL-INST|OBL-GEN
        |XCOMP-PRED|ADJUNCT $}.
```

Each of the rules in (9.58)–(9.61) assigns an appropriate path to the %L variable: in (9.58) the path bound to the %L variable is (! GF* (POSS)) – it checks the path consisting of any sequence (including zero) of grammatical functions defined in (9.62) (the definition of GF) and optionally one POSS attribute corresponding to a genitive possessive modifier. In (9.59), %L corresponds to (! (OBJ) (ADJUNCT $) (POSS)) – it allows for an optional object (for semantic prepositions, see § 2.3), an optional adjunct and an optional possessive modifier (in this order). Finally, (9.60) and (9.61) use the same very simple path – %L corresponds to !, which is the category to which the annotation is attached.

This variable is passed to TYPE-SUBC-PATH template (see (9.63) for its definition) as one of its parameters. This template ensures that the item in the relevant path (%L) is of the type specified in its second parameter – it is _type in template calls in (9.58)–(9.61), which is the same as the value of the complex category of the left-hand side of these rules.

(9.63)
```
TYPE-SUBC-PATH(PATH T) = "checks that TYPE in PATH is equal to T"
(PATH TYPE)=c T.
```

As a result, the value of `TYPE` attribute in the f-structure the plain category (`AP`, for instance) is required to be the same as the `_type` variable in the complex category – this feature is used in rules handling lexico-semantic coordination to ensure that only conjuncts of the same type are coordinated (see (9.41) for monoclausal coordination and (9.56) for multiclausal coordination).

9.4 Summary

This chapter presented the implementation of phenomena discussed in ch. 4 and ch. 5, namely the rules required for handling the coordination of unlike categories (complementing the discussion of constraints introduced by particular predicates in the lexicon, provided in ch. 8) and coordination of different grammatical functions, respectively.

Chapter 10

Conclusion

The aim of this work was to provide a formal linguistic analysis of coordination which is not typical in the way that it either involves conjuncts of different morphosyntactic categories or different grammatical functions. Both types of coordination are occasionally accused by linguists of being outside the system of the language, of being "derailed", to use the term of Saloni (2005). Przepiórkowski and Patejuk 2014 argue that, when "system" is understood in the sense of the classical triad "system – norm – speech" (Coșeriu 1952), the (more controversial) coordination of unlike grammatical functions must be understood as part of both "system" and "norm" (and, more obviously, "speech"). However, another understanding of the term "system" is also briefly discussed there, namely, the one proposed in Bobrowski 2005, p. 44–45, where "system" is understood as the (often formal) system constructed by the linguist, not as the internalised system of lay native speakers. This dissertation firmly shows that unlike coordination is systematic also in this – perhaps less obvious – sense: as mentioned a few times, the account of proposed here forms a part of a much more extensive "system" (i.e. a formal grammar) of Polish (Patejuk and Przepiórkowski 2012c, 2015b).

For the average linguistic reader, the extent of formalisation of the proposed analysis may be daunting, if not prohibitive. However, this work reflects the belief that such formalisation greatly enhances the scientific quality of the resulting analysis: the analysis is not formulated in imprecise natural language and diluted by vague terms, but its predictions may be derived formally or semi-formally. That is, formalisation enhances falsifiability, believed by some philosophers of science to be an important criterion for calling a theory scientific (Popper 1935):

> But I shall certainly admit a system as empirical or scientific only if it is capable of being tested by experience. These considerations suggest that not the *verifiability* but the *falsifiability* of a system is to be taken as a criterion of demarcation. In other words: I shall not require of a scientific system that it shall be capable of being singled out, once and for all, in a positive sense; but I shall require that its logical form shall be such that it can be singled out, by means of empirical tests, in a negative sense: *it must be possible for an empirical scientific system to be refuted by experience.*
>
> (Popper 2005, p. 18)

This need for rigorous formalisation in linguistics was recognised over 60 years ago, as this oft-cited quotation testifies:

> The search for rigorous formulation in linguistics has a much more serious motivation than mere concern for logical niceties or the desire to purify well-established methods of linguistic analysis. Precisely constructed models for linguistic structure can play an important role, both negative and positive, in the process of discovery itself. By pushing a precise but inadequate formulation to an unacceptable conclusion, we can often expose the exact source of this inadequacy and, consequently, gain a deeper understanding of the linguistic data. More positively, a formalized theory may automatically provide solutions for many problems other than those for which it was explicitly designed. Obscure and intuition-bound notions can neither lead to absurd conclusions nor provide new and correct ones, and hence they fail to be useful in two important respects. I think that some of those linguists who have questioned the value of precise and technical development ot linguistic theory may have failed to recognise the productive potential in the method of rigorously stating a proposed theory and applying it strictly to linguistic material with no attempt to avoid unacceptable conclusions by ad hoc adjustments or loose formulation.
>
> (Chomsky 1955, p. 5)

Now, well over half a century later, formally-minded linguists have at their disposal additional tools for possibly falsifying their theories, tools offered by the field of computational linguistics. By not only *formalising* an analysis, but also *implementing* it in a computational system which is able to automatically decide whether the grammar accepts a sentence or not, the predictions of the analysis may be tested much more reliably and effectively: where a linguists may make mistakes in their deductive reasoning and will always need a considerable amount of time to perform this reasoning, computers will provide answers quickly and (almost) unfailingly. For this reason, much space is devoted in this dissertation to the issue of the faithful implementation of the proposed analysis.

Obviously, formalisation for the sake of formalisation is of little value if the formalised theory is empirically limited or linguistically uninsightful. The analysis of coordination of unlike categories proposed here, while rather straightforward in attributing the possibility of such coordination to idiosyncratic properties of lexical items, is empirically broad, as it is based on a valence dictionary of Polish encompassing over 11000 lexical entries and over 50000 valence schemata. The analysis of coordination of unlike functions is much more limited empirically, as it is concerned with a closed class of roughly pronominal lexemes, but the proposed account is the first comprehensive analysis of lexico-semantic coordination in Polish (which is a part of a large general grammar of Polish) and it extends previous accounts for other languages.

On the other hand, there are obvious ways in which the purely syntactic analysis proposed here is limited, the main being the lack of any semantic insight. This omission is not programmatic but practical: the material discussed here is already very broad and the formalisation already rather extensive, so adding semantic considerations would expand this work well beyond the usual confines of PhD dissertations. But it should be clear that, for the complete account, such semantic research is necessary for both types of unlike coordination. In case of coordination of unlike categories, the natural question is: to what extent is the possibility of categorially different phrase types filling the same syntactic position determined by the lexical semantics of the governing predicate? In case of coordination of unlike functions, the question that immediately arises is: what is the full repertoire of semantic classes of pronominal or quantificational expressions that can be coordinated in this way? And, further, what are the common – if any – semantic properties of all these classes: can all their elements be described as quantificational or perhaps as expressing indefinite variables? Such questions should undoubtedly be a matter of further research.

Appendix A

Summary in English

This dissertation[1] focuses on two coordination phenomena which are non-standard: conjuncts are not identical categorially or they do not correspond to the same grammatical function. The dissertation is based on rich attested data taken from the National Corpus of Polish (NKJP) and retrieved using Google. Furthermore, the dissertation is accompanied by an XLE implementation of a grammar of Polish which takes such coordination phenomena into account.

Unlike category coordination

The issue of coordination of unlike categories was noticed in constraint-based theories of grammar as early as 1985, see the discussion in Sag *et al.* 1985. In LFG such coordination was mentioned in the so-called COMP vs OBJ debate (e.g. Alsina *et al.* 2005), but since this discussion focused on how particular grammatical functions should be defined, no formalised account of coordination of unlike categories was offered and no constraints necessary to handle this phenomenon were provided. As a result, this debate did not touch upon the issue of imposing different constraints on particular conjuncts under coordination, which turns out to be problematic in LFG because of the way in which disjunctive statements are interpreted in this context.

This dissertation aims to fill this gap by discussing how unlike category coordination can be modelled in LFG and showing in detail how the lexicon should be designed to account for the coordination of unlike categories.

Consider the following examples from Polish:

(A.1) Doradził mu [wyjazd] i [żeby nie wracał].
advised him departure.ACC and that NEG come back
'He advised him to leave and not to come back.' (Kallas 1993, p. 92, ex. (48a))

(A.2) (Wcale) nie doradził mu [wyjazdu] ani [żeby nie wracał].
not at all NEG advised him departure.GEN nor that NEG come back
'He did not advise him to leave nor not to come back.'

In both examples provided above, the object is a coordinate phrase which consists of a nominal (a form of the noun WYJAZD, 'departure') and a clause (*żeby nie wracał*, 'not to come back').

[1] This summary is based on the abstract of a paper presented during the PhD session of LFG'14 conference: http://lfg-conference.org/resources/Patejuk_diss.pdf.

However, these examples differ consistently depending on whether negation (the negation marker NIE) is present – in (A.1) the nominal conjunct takes the accusative case, while in (A.2) the object is marked for the genitive case. Such variation is attributable to the fact that the verb DORADZIĆ 'advise' may take a structurally case-marked object: its case marking depends on the syntactic environment, namely on the availability of sentential negation.

When plain disjunctive LFG constraints are used, one of the disjuncts is selected and it is distributed to all elements of the coordinate structure. As a result, all conjuncts must satisfy the same constraint, which is not the desired effect for unlike category coordination where different conjuncts may satisfy different constraints.

An alternative is to convert plain constraints into off-path constraints so that each conjunct can satisfy a different constraint – the disjunctive statement is evaluated for each element of the coordinate phrase separately. Furthermore, it is demonstrated how complex phenomena such as structural case assignment to the nominal conjunct can be modelled successfully using this formal device.

This solution assumes that all elements of the coordinate structure correspond to the same grammatical function. However, since sometimes the choice of the common grammatical function is problematic and controversial, issues related to its choice are discussed.

Coordination of different grammatical functions

On the other hand, there is the phenomenon of coordination of different grammatical functions – it is known under a wide range of names, including hybrid coordination, lexico-semantic coordination and – more narrowly – coordinated *wh*-questions.

This dissertation provides evidence that real coordination is involved in this phenomenon. It shows that conjuncts must belong to the same semantic type and that the range of possible types is usually restricted to *wh*-words and items which express various quantifiers, though it also discusses less frequent conjunct types (together with how they can be modified). Different types of dependents may be coordinated: arguments, modifiers and even particles. Furthermore, conjuncts do not have to be dependents of the same head – they may belong to different substructures of the relevant f-structure. The dissertation offers a formalised analysis which takes into account main (most frequent) classes of conjuncts taking part in such coordination.

Let us take a closer look at a few relevant examples from Polish, focusing on *wh*-words:

(A.3) Co, komu i z czym się kojarzy, to jego prywatna sprawa.
what.NOM who.DAT and with what.INST REFL associate is SELF.GEN own business
'Who associates what with what is their own business.' (NKJP)

(A.4) Kontrola w firmie – jakie i kto może ponieść konsekwencje?
audit in company what.ACC and who.NOM can bear consequences.ACC
'Company audit – who can suffer what consequences?' (Google)

(A.5) Tytuł brzmiał prosto i uczciwie: "Czy *(i) jaki jest Bóg"
title sounded simply and honestly PART and what.NOM is God.NOM
'The title sounded simple and honest: "Does God exist and what is he like?"'(NKJP)

In (A.3) all three conjuncts are arguments of the verb KOJARZYĆ 'associate': the first one (*Co*) is the subject, the second one (*komu*) is the object, while the last one (*z czym*) is the oblique object. In (A.4) particular conjuncts also belong to the class of question words, but they depend on different heads: the first conjunct (*jakie*) is the modifier of the object of the verb PONIEŚĆ 'bear', which is the infinitival complement of the main verb MÓC 'can', whose subject is the second conjunct, *kto*. In (A.5) the *yes/no* question particle CZY is coordinated with *jaki*, the predicative complement of the verb BYĆ 'be'. Unlike in previous examples, the conjunction cannot be removed from (A.5) – this results in ungrammaticality, as indicated in the relevant example. This is attributable to the fact that CZY cannot co-occur with other question words in the same clause. For this reason, coordination of different grammatical functions involving the question particle is analysed differently.

The formal account of this phenomenon involves two types of f-structure representation: monoclausal, where all conjuncts belong to the same clause, and multiclausal, where particular conjuncts belong to different clauses. Though both representations are possible in theory, a wide range of tests arguing in favour or against particular representation is discussed, and their validity and conclusiveness when applied to Polish data is assessed. As a result of the overview of these arguments, monoclausal representation is adopted for all instances of coordination of different grammatical functions. There is, however, one notable exception: when one of the conjuncts is the question particle, multiclausal representation should be assumed instead – see (A.5) and related discussion above.

The matter of representing the conjunction which is used under this variety of coordination is given a considerable amount of attention: the issue of where it should be placed in the f-structure is discussed, taking into account the possibility of embedding of such coordination, as well as potential problems related to the interaction of coordination of different grammatical functions with verbal coordination.

Finally, less frequent types of conjuncts taking part in such coordination are discussed. The issue of how conjuncts under such coordination can be modified is also given due attention. See the selected examples below:

(A.6) Nie wszyscy i nie każdemu mogą sprzedać broń.
NEG all.NOM and NEG everybody.DAT can sell arms
'It is not the case that everyone can sell arms to everybody.' (Google)

(A.7) W pracy mało kto i mało kogo tak naprawdę lubi.
at work few someone.NOM and few someone.ACC really likes
'Hardly anybody really likes hardly anyone at work.' (Google)

(A.8) Podobno mężczyźni wiele obiecują, ale tylko niektórzy (i tylko niektóre
supposedly men much promise but only some.NOM and only some.ACC
obietnice) spełniają.
promises.ACC keep
'Supposedly men promise a lot, but only some keep some promises.' (Google)

(A.9) Każde dziecko musi dostać tyle samo i takich samych cukierków w
every child must get the same amount and the same candies at

tym samym czasie.
the same time
'Every child must get the same amount of identical candy at the same time.' (Google)

In (A.6) conjuncts have the semantics of the universal quantifier under the scope of negation (roughly, 'not every'). In (A.7) *mało* ('few') is used as the modifier of words which are identical in shape to corresponding *wh*-words but have the semantics of existential pronouns. In (A.8) each conjunct is accompanied by *tylko* ('only') – it does not change the semantics of the pronouns involved, though when it is used, the meaning of the entire phrase is not neutral, it suggests that only a few elements of the set satisfy the relevant condition. In (A.9) both conjuncts, *tyle* ('that many') and *takich* ('such'), are modified by elements which express identity in comparison to some point of reference (the same amount of candy, the same type of candy for each child), *samo* and *samych*, respectively.

Appendix B

Summary in Polish

Niniejsza rozprawa skupia się na dwóch zjawiskach niestandardowej koordynacji: poszczególne człony koordynacji nie są identyczne pod względem kategorii składniowej lub nie odpowiadają tej samej funkcji gramatycznej. Rozprawa jest oparta na bogatych autentycznych danych z Narodowego Korpusu Języka Polskiego (NKJP) i znalezionych przy pomocy wyszukiwarki Google. Ponadto rozprawie towarzyszy zaimplementowana w XLE gramatyka języka polskiego, która bierze takie zjawiska pod uwagę oraz obejmuje je analizą.

Koordynacja różnych kategorii

Zagadnienie koordynacji różnych kategorii zostało zauważone w teoriach unifikacyjnych już w 1985 (zob. Sag *et al.* 1985). W LFG kwestia takiej koordynacji została poruszona w tzw. COMP *vs* OBJ *debate* (m.in. Alsina *et al.* 2005), jednak ponieważ owa debata skupiła się na tym, jak powinny być zdefiniowane poszczególne funkcje gramatyczne, nie zaproponowano w niej żadnej formalizacji koordynacji różnych kategorii – nie zostały zaproponowane ograniczenia umożliwiające objęcie tego zjawiska analizą. W rezultacie wspomniana debata nie poruszyła zagadnienia narzucania różnych ograniczeń na poszczególne człony koordynacji, co okazuje się być problematyczne w teorii LFG z powodu sposobu, w jaki są interpretowane w tym kontekście ograniczenia z alternatywą.

Niniejsza rozprawa stawia sobie za cel wypełnienie tej luki poprzez omówienie, w jaki sposób koordynacja różnych kategorii może być opisana w formalizmie LFG, pokazując szczegółowo, jak powinien być zorganizowany leksykon, aby opisać koordynację różnych kategorii.

Rozważmy następujące przykłady:

(B.1) Doradził mu [wyjazd] i [żeby nie wracał]. (Kallas 1993, p. 92, ex. (48a))

(B.2) (Wcale) nie doradził mu [wyjazdu] ani [żeby nie wracał].

W obu powyższych przykładach dopełnienie jest skoordynowaną frazą, która składa się z frazy nominalnej (formy rzeczownika WYJAZD) oraz frazy zdaniowej ze spójnikiem podrzędnym (*żeby nie wracał*). Należy jednak zauważyć, że te przykłady różnią się w zależności od tego, czy występuje negacja zdaniowa (obecny jest znacznik NIE) – w (B.1) człon nominalny występuje w bierniku, podczas gdy w (B.2) odpowiadający mu człon nominalny jest w dopełniaczu. Wskazane różnice wynikają z tego, że czasownik DORADZIĆ może wystąpić z dopełnieniem o przypadku strukturalnym: jego wartość przypadka zależy od kontekstu składniowego, mianowicie od występowania negacji zdaniowej.

Gdy używane są zwykłe ograniczenia LFG z alternatywą, wybierany jest jeden z jej członów i jest on dystrybuowany na wszystkie elementy struktury skoordynowanej. W efekcie wszystkie człony koordynacji muszą spełniać te same ograniczenia, co nie jest pożądanym rezultatem przy koordynacji różnych kategorii, gdzie poszczególne człony mogą spełniać różne ograniczenia.

Alternatywą jest przepisanie zwykłych ograniczeń na tzw. *off-path constraints*, dzięki czemu każdy z członów koordynacji może spełniać inne ograniczenia – alternatywa w ograniczeniu jest rozstrzygana niezależnie dla każdego elementu skoordynowanej frazy. Ponadto w pracy zaprezentowano, w jaki sposób przy pomocy takich ograniczeń mogą być modelowane złożone zjawiska składniowe takie jak nadawanie przypadka strukturalnego członowi nominalnemu.

Opisywane rozwiązanie korzystające z *off-path constraints* zakłada, że wszystkie elementy struktury skoordynowanej odpowiadają tej samej funkcji gramatycznej. Jednak ponieważ zdarza się, że wybór wspólnej funkcji gramatycznej jest problematyczny i kontrowersyjny, omówione są kwestie związane z jej wyborem.

Koordynacja różnych funkcji gramatycznych

Koordynacji mogą podlegać również różne, odmienne funkcje gramatyczne – to zjawisko jest znane w literaturze pod wieloma nazwami, m.in. jako koordynacja hybrydowa (*hybrid coordination*), koordynacja leksykalno-semantyczna (*lexico-semantic coordination*) oraz, w węższym znaczeniu, koordynacja zaimków pytajnych (*coordinated wh-questions*).

Niniejsza praca dowodzi, że we wspomnianym zjawisku występuje prawdziwa koordynacja. Pokazuje, że poszczególne człony koordynacji muszą należeć do tej samej klasy semantycznej oraz że zakres możliwych klas jest przeważnie ograniczony do zaimków pytajnych oraz elementów wyrażających rozmaite kwantyfikatory, chociaż dyskutowane są również mniej częste klasy (wraz z możliwymi rodzajami modyfikacji). Skoordynowane mogą być różne rodzaje podrzędników: argumenty, modyfikatory oraz znaczniki. Ponadto, skoordynowane elementy nie muszą być podrzędnikami tego samego nadrzędnika (głowy) – mogą należeć do różnych podstruktur odpowiedniej f-struktury. Praca proponuje sformalizowaną analizę, która obejmuje główne (najczęstsze) klasy elementów podlegających takiej koordynacji.

Przyjrzyjmy się bliżej paru przykładom, skupiając się na pytaniach:

(B.3) Co, komu i z czym się kojarzy, to jego prywatna sprawa. (NKJP)

(B.4) Kontrola w firmie – jakie i kto może ponieść konsekwencje? (Google)

(B.5) Tytuł brzmiał prosto i uczciwie: Ćzy *(i) jaki jest Bóg" (NKJP)

W (B.3) wszystkie trzy człony koordynacji są argumentami czasownika KOJARZYĆ: pierwszy (*Co*) jest podmiotem, drugi (*komu*) jest dopełnieniem, z kolei ostatni (*z czym*) jest dopełnieniem przyimkowym. W (B.4) poszczególne człony koordynacji również należą do klasy zaimków pytajnych, jednak są one podrzędnikami różnych nadrzędników (głów): pierwszy (*jakie*) jest modyfikatorem dopełnienia czasownika PONIEŚĆ, który jest dopełnieniem bezokolicznikowym czasownika głównego MÓC, którego podrzędnikiem jest z kolei drugi człon koordynacji, *kto*, który jest jego podmiotem. W (B.5) znacznik CZY występujący w pytaniach o rozstrzygnięcie jest skoordynowany z dopełnieniem predykatywnym czasownika BYĆ. W przeciwieństwie do wcześniejszych przykładów, spójnik nie może zostać usunięty z (B.5), ponieważ prowadzi to do niegramatyczności. Wynika to z tego, że CZY nie może występować we frazie, gdzie znajdują się inne zaimki pytajne. Z tego powodu koordynacja różnych funkcji gramatycznych, gdzie występuje znacznik CZY, jest analizowana w inny sposób niż pozostałe przypadki takiej koordynacji.

Formalna analiza zjawiska koordynacji różnych funkcji gramatycznych zakłada dwa rodzaje reprezentacji na poziomie f-struktury: jednozdaniową, gdzie wszystkie człony koordynacji należą do tego samego zdania, oraz wielozdaniową, gdzie poszczególne człony koordynacji należą do różnych zdań (są podrzędnikami różnych predykatów). Chociaż obie reprezentacje są teoretycznie możliwe, dyskutowane jest szerokie spektrum argumentów za lub przeciw danemu sposobowi reprezentacji; poddana jest też ocenie kwestia tego, na ile przekonujący i istotny jest dany argument w odniesieniu do danych z języka polskiego. W wyniku przeglądu wspomnianych argumentów przyjęta została analiza jednozdaniowa jako odpowiednia dla wszystkich przypadków koordynacji różnych funkcji gramatycznych. Istnieje jednak istotny wyjątek: gdy jednym z członów takiej koordynacji jest znacznik CZY, przyjmowana jest analiza wielozdaniowa – zob. (B.5) oraz dyskusję powyżej.

Dużo uwagi poświęcono również zagadnieniu reprezentacji spójnika występującego w takiej koordynacji: omówiono kwestię tego, w którym miejscu f-struktury powinien być reprezentowany spójnik, biorąc pod uwagę możliwość zagnieżdżania takiej koordynacji oraz potencjalne trudności związane z interakcjami między koordynacją różnych funkcji gramatycznych i koordynacją na poziomie fraz werbalnych.

Wreszcie zostały wspomniane również mniej częste typy semantyczne elementów mogących wziąć udział w takiej koordynacji. Poświęcono również uwagę kwestii tego, w jaki sposób mogą być modyfikowane człony takiej koordynacji. Spójrzmy na przykłady podane poniżej:

(B.6) Nie wszyscy i nie każdemu mogą sprzedać broń. (Google)

(B.7) W pracy mało kto i mało kogo tak naprawdę lubi. (Google)

(B.8) Podobno mężczyźni wiele obiecują, ale tylko niektórzy (i tylko niektóre obietnice) spełniają. (Google)

(B.9) Każde dziecko musi dostać tyle samo i takich samych cukierków w tym samym czasie. (Google)

W (B.6) człony koordynacji mają semantykę odpowiadającą kwantyfikatorowi uniwersalnemu w zasięgu negacji (w przybliżeniu: *nie każdy* – istnieje element, który nie spełnia danego warunku). W (A.7) *mało* jest modyfikatorem słów, które, choć ich forma jest identyczna do zaimków pytajnych, mają semantykę zaimków egzystencjalnych. W (B.8) przy każdym z członów koordynacji występuje słowo *tylko*, które nie zmienia semantyki odpowiednich zaimków, jednak sprawia, że znaczenie odpowiedniej frazy nie jest już neutralne, ponieważ sugeruje, że jedynie niewielka część elementów odpowiedniego zbioru spełnia dany warunek. W (B.9) oba człony koordynacji, *tyle* i *takich*, są modyfikowane przez *samo* oraz *samych* – słowa wyrażające tożsamość przy określonym punkcie odniesienia (taka sama ilość cukierków, taki sam rodzaj cukierków).

Bibliography

Alsina, A., Mohanan, T., and Mohanan, K. (2005). How to get rid of the COMP. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'05 Conference*, Stanford, CA. CSLI Publications.

Arka, I. W. and Simpson, J. (1998). Control and complex arguments in Balinese. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'98 Conference*, Stanford, CA. CSLI Publications.

Asudeh, A. (2011). Towards a unified theory of resumption. In A. Rouveret, editor, *Resumptive Pronouns at the Interfaces*, pages 121–188. John Benjamins.

Asudeh, A., Dalrymple, M., and Toivonen, I. (2013). Constructions with lexical integrity. *Journal of Language Modelling*, **1**(1), 1–54.

Babby, L. H. (1980a). *Existential Sentences and Negation in Russian*. Karoma Publishers, Ann Arbor, MI.

Babby, L. H. (1980b). The syntax of surface case marking. In W. Harbert and J. Herschensohn, editors, *Cornell Working Papers in Linguistics*, volume 1, pages 1–32. Department of Modern Languages and Linguistics, Cornell University.

Bayer, S. (1996). The coordination of unlike categories. *Language*, **72**(3), 579–616.

Beesley, K. and Karttunen, L. (2003). *Finite State Morphology*. CSLI Publications.

Berman, J. (2007). Functional identification of complement clauses in German and the specification of COMP. In A. Zaenen, J. Simpson, T. H. King, J. Grimshaw, J. Maling, and C. Manning, editors, *Architectures, Rules, and Preferences: Variations on Themes by Joan W. Bresnan*, pages 69–83. CSLI Publications, Stanford, CA.

Bień, J. S. and Saloni, Z. (1982). Pojęcie wyrazu morfologicznego i jego zastosowanie do opisu fleksji polskiej (wersja wstępna). *Prace Filologiczne*, **XXXI**, 31–45.

Bobrowski, I. (1988). *Gramatyka Generatywno-Transformacyjna (TG) a Uogólniona Gramatyka Struktur Frazowych (GPSG)*. Prace Habilitacyjne. Ossolineum, Wrocław.

Bobrowski, I. (2005). *Składniowy model polszczyzny*. Lexis, Cracow.

Börjars, K. and Vincent, N. (2008). Objects and OBJ. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'08 Conference*, pages 150–168, Stanford, CA. CSLI Publications.

Bresnan, J. (1982a). Control and complementation. *Linguistic Inquiry*, **13**(3). Also in Joan Bresnan, ed. 1982. The Mental Representation of Grammatical Relations, Chapter 5, pp. 282–390. Cambridge, MA: The MIT Press.

Bresnan, J., editor (1982b). *The Mental Representation of Grammatical Relations*. MIT Press Series on Cognitive Theory and Mental Representation. The MIT Press, Cambridge, MA.

Bresnan, J. (2000). *Lexical-Functional Syntax*. Blackwell Textbooks in Linguistics. Blackwell.

Butt, M. and King, T. H., editors (2007). *The Proceedings of the LFG'07 Conference*, Stanford, CA. CSLI Publications.

Butt, M. and King, T. H., editors (2010). *The Proceedings of the LFG'10 Conference*, Stanford, CA. CSLI Publications.

Butt, M. and King, T. H., editors (2012). *The Proceedings of the LFG'12 Conference*, Stanford, CA. CSLI Publications.

Butt, M. and King, T. H., editors (2014). *The Proceedings of the LFG'14 Conference*, Stanford, CA. CSLI Publications.

Butt, M., King, T. H., Masuichi, H., and Rohrer, C. (2002). The parallel grammar project. In N. J. Carroll and R. Sutcliffe, editors, *Proceedings of the Workshop on Grammar Engineering and Evaluation*, pages 1–7.

Bîlbîie, G. and Gazdik, A. (2012). Wh-coordination in Hungarian and Romanian multiple questions. *Empirical Issues in Syntax and Semantics 9*, pages 19–36.

Błaszczak, J. (2001). *Covert Movement and the Genitive of Negation in Polish*, volume 15 of *Linguistics in Potsdam*. Universität Potsdam, Potsdam.

Calzolari, N., Choukri, K., Declerck, T., Loftsson, H., Maegaard, B., Mariani, J., Moreno, A., Odijk, J., and Piperidis, S., editors (2014). *Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014*, Reykjavík, Iceland. ELRA.

Chaves, R. P. (2006). Coordination of Unlikes without Unlike Categories. In S. Müller, editor, *The Proceedings of the 13th International Conference on Head-Driven Phrase Structure Grammar*, pages 102–122, Stanford. CSLI Publications.

Chaves, R. P. and Paperno, D. (2007). On the Russian hybrid coordination construction. In S. Müller, editor, *The Proceedings of the 14th International Conference on Head-Driven Phrase Structure Grammar*, pages 46–64, Stanford. CSLI Publications.

Chomsky, N. (1955). *The Logical Structure of Linguistic Theory*. Plenum Press, New York. Published in 1975.

Chomsky, N. (1980). On binding. *Linguistic Inquiry*, **11**(1), 1–46.

Chomsky, N. (1981). *Lectures on Government and Binding*. Foris, Dordrecht.

Citko, B. (2013). The puzzles of *wh*-questions with coordinated *wh*-pronouns. In T. Biberauer and I. Roberts, editors, *Challenges to Linearization*, number 114 in Studies in Generative Grammar, pages 295–330. De Gruyter Mouton, Berlin, Boston.

Citko, B. and Gračanin-Yüksek, M. (2012). Wh-coordination in free relatives. Abstract of a talk presented at the 43rd annual meeting of the North East Linguistic Society, City University of New York, 19–21 October 2012; `http://nels2012.commons.gc.cuny.edu/?page_id=14`.

Citko, B. and Gračanin-Yüksek, M. (2013). Towards a new typology of coordinated *wh*-questions. *Journal of Linguistics*, **49**, 1–32.

Coșeriu, E. (1952). *Sistema, norma y habla*. Universidad de la Republica, Montevideo.

Crouch, D., Dalrymple, M., Kaplan, R., King, T., Maxwell, J., and Newman, P. (2011). XLE documentation. `http://www2.parc.com/isl/groups/nltt/xle/doc/xle_toc.html`.

Dalrymple, M. (1993). *The Syntax of Anaphoric Binding*. Number 36 in CSLI Lecture Notes.

CSLI Publications, Stanford, CA.

Dalrymple, M. (2001). *Lexical Functional Grammar*. Academic Press.

Dalrymple, M. and Hristov, B. (2010). Agreement patterns and coordination in Lexical Functional Grammar. In Butt and King (2010), pages 186–206.

Dalrymple, M. and Kaplan, R. M. (2000). Feature indeterminacy and feature resolution. *Language*, **76**(4), 759–798.

Dalrymple, M. and Lødrup, H. (2000). The grammatical functions of complement clauses. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'00 Conference*, Stanford, CA. CSLI Publications.

Dalrymple, M., King, T. H., and Sadler, L. (2009). Indeterminacy by underspecification. *Journal of Linguistics*, **45**, 31–68.

Danielewiczowa, M. (1996). *O znaczeniu zdań pytajnych w języku polskim. Charakterystyka struktury tematyczno-rematycznej wypowiedzeń interrogatywnych.* Wydawnictwa Uniwersytetu Warszawskiego, Warsaw.

Dyvik, H., Meurer, P., Rosén, V., and De Smedt, K. (2009). Linguistically motivated parallel parsebanks. In *Proceedings of The Eighth Workshop on Treebanks and Linguistic Theories (TLT8)*, pages 71–82, Milano, Italy. EDUCatt.

Dyła, S. (1984). Across-the-board dependencies and case in Polish. *Linguistic Inquiry*, **15**(4).

Dziwirek, K. (1990). Default agreement in Polish. In K. Dziwirek, P. Farrell, and E. Mejías-Bikandi, editors, *Grammatical Relations: A Cross-Theoretical Perspective*, pages 147–161. CSLI Publications, Stanford, CA.

Forst, M. (2006). COMP in (parallel) grammar writing. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'06 Conference*, Stanford, CA. CSLI Publications.

Gazdar, G., Klein, E., Pullum, G. K., and Sag, I. A. (1985). *Generalized Phrase Structure Grammar*. Blackwell / Harvard University Press, Cambridge / Cambridge, MA.

Gazdik, A. (2010). Multiple Questions in French and Hungarian: An LFG Account. In Butt and King (2010), pages 249–269.

Gazdik, A. (2011). *Multiple Questions in French and Hungarian. A Lexical-Functional Analysis with Special Emphasis on the Syntax-Discourse Interface*. Ph.D. dissertation, Université Paris 7 – Paris Diderot.

Hall, B. C. (1965). *Subject and Object in Modern English.* Ph.D. dissertation, Massachusetts Institute of Technology.

Heinz, W. and Matiasek, J. (1994). Argument structure and case assignment in German. In Nerbonne *et al.* (1994), pages 199–236.

Kallas, K. (1993). *Składnia współczesnych polskich konstrukcji współrzędnych.* Wydawnictwo Uniwersytetu Mikołaja Kopernika, Toruń.

Kazenin, K. I. (2001). On coordination of Wh-phrases in Russian. MS, University of Tübingen and Moscow State University.

King, T. H. (1995). *Configuring Topic and Focus in Russian.* Dissertations in Linguistics Series. CSLI Publications.

Kosek, I. (1999). *Przyczasownikowe frazy przyimkowo-nominalne w zdaniach współczesnego* *języka polskiego*. Wydawnictwo Uniwersytetu Warmińsko-Mazurskiego, Olsztyn.

Krasnowska, K. and Kieraś, W. (2013). Polish LFG treebank on a shoestring. In *Proceedings of The Twelfth Workshop on Treebanks and Linguistic Theories (TLT12)*, pages 109–120, Sofia, Bulgaria.

Kuhn, J. and Sadler, L. (2007). Single conjunct agreement and the formal treatment of coordination in LFG. In Butt and King (2007), pages 302–322.

Lipták, A. (2001). *On the Syntax of Wh-items in Hungarian*. LOT (Series). LOT, Netherlands Graduate School of Linguistics.

Lipták, A. (2012). Strategies of wh-coordination. *Linguistic Variation*, **11**, 149–188.

Lødrup, H. (2012). In search of a nominal COMP. In Butt and King (2012), pages 383–403.

Marciniak, M., Mykowiecka, A., Przepiórkowski, A., and Kupść, A. (2003). An HPSG-annotated test suite for Polish. In A. Abeillé, editor, *Treebanks: Building and Using Parsed Corpora*, volume 20 of *Text, Speech and Language Technology*, pages 129–146. Kluwer, Dordrecht.

Maxwell, J. T. and Manning, C. D. (1996). A theory of non-constituent coordination based on finite-state rules. In M. Butt and T. H. King, editors, *The Proceedings of the LFG'96 Conference*, Stanford, CA. CSLI Publications.

Mańczak, W. (1956). Ile rodzajów jest w polskim. *Język Polski*, **XXXVI**(2), 116–121.

Mel'čuk, I. (1988). *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, NY.

Neidle, C. (1982). Case agreement in Russian. In Bresnan (1982b), pages 391–426.

Neidle, C. (1988). *The Role of Case in Russian Syntax*. Kluwer, Dordrecht.

Nerbonne, J., Netter, K., and Pollard, C., editors (1994). *German in Head-Driven Phrase Structure Grammar*. Number 46 in CSLI Lecture Notes. CSLI Publications, Stanford, CA.

Paperno, D. (2012). *Semantics and Syntax of Non-Standard Coordination*. Ph.D. dissertation, University of California.

Patejuk, A. (2015). Coordinated *wh*-words in Polish: monoclausal or multiclausal? In D. Kavitskaya and M. Szajbel-Keck, editors, *Formal Approaches to Slavic Linguistics #23*, pages 222–241, University of Michigan, Ann Arbor. Michigan Slavic Publications. In print.

Patejuk, A. and Przepiórkowski, A. (2012a). A comprehensive analysis of constituent coordination for grammar engineering. In *Proceedings of the 24rd International Conference on Computational Linguistics (COLING 2012)*.

Patejuk, A. and Przepiórkowski, A. (2012b). Lexico-semantic coordination in Polish. In Butt and King (2012), pages 461–478.

Patejuk, A. and Przepiórkowski, A. (2012c). Towards an LFG parser for Polish: An exercise in parasitic grammar development. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, pages 3849–3852, Istanbul, Turkey. ELRA.

Patejuk, A. and Przepiórkowski, A. (2014a). Control into selected conjuncts. In Butt and King (2014), pages 448–460.

Patejuk, A. and Przepiórkowski, A. (2014b). In favour of the raising analysis of passivisation. In Butt and King (2014), pages 461–481.

Patejuk, A. and Przepiórkowski, A. (2014c). Lexico-semantic coordination in Polish: a critical review of tests for determining representation. In M. Gębka-Wolak, J. Kamper-Warejko, and

A. Moroz, editors, *Leksyka języków słowiańskich w badaniach synchronicznych i diachronicznych*, pages 119–134, Toruń. Wydawnictwo Naukowe Uniwersytetu Mikołaja Kopernika.

Patejuk, A. and Przepiórkowski, A. (2014d). Structural case assignment to objects in Polish. In Butt and King (2014), pages 429–447.

Patejuk, A. and Przepiórkowski, A. (2014e). Synergistic development of grammatical resources: a valence dictionary, an LFG grammar, and an LFG structure bank for Polish. In *Proceedings of The Thirteenth Workshop on Treebanks and Linguistic Theories (TLT13)*, Tübingen, Germany.

Patejuk, A. and Przepiórkowski, A. (2015a). LFG analysis of SIĘ marker in Polish. Abstract of a talk to be presented at the LFG'15 Conference.

Patejuk, A. and Przepiórkowski, A. (2015b). Parallel development of linguistic resources: Towards a structure bank of Polish. *Prace Filologiczne*, **LXV**. To appear.

Peterson, P. G. (2004). Coordination: Consequences of a lexical-functional account. *Natural Language and Linguistic Theory*, **22**(3), 643–679.

Pollard, C. (1994). Toward a unified account of passive in German. In Nerbonne *et al.* (1994), pages 273–296.

Pollard, C. and Sag, I. A. (1987). *Information-Based Syntax and Semantics, Volume 1: Fundamentals*. Number 13 in CSLI Lecture Notes. CSLI Publications, Stanford, CA.

Pollard, C. and Sag, I. A. (1994). *Head-driven Phrase Structure Grammar*. Chicago University Press / CSLI Publications, Chicago, IL.

Popper, K. (1935). *Logik der Forschung*. Julius Springer, Vienna.

Popper, K. (2005). *The Logic of Scientific Discovery*. Routledge, London. First published in 1935 as *Logik der Forschung*.

Przepiórkowski, A. (1999). *Case Assignment and the Complement-Adjunct Dichotomy: A Non-Configurational Constraint-Based Approach*. Ph.D. dissertation, Universität Tübingen, Germany.

Przepiórkowski, A. (2000). Long distance genitive of negation in Polish. *Journal of Slavic Linguistics*, **8**, 151–189.

Przepiórkowski, A. (2004a). *The IPI PAN Corpus: Preliminary version*. Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Przepiórkowski, A. (2004b). O wartości przypadka podmiotów liczebnikowych. *Biuletyn Polskiego Towarzystwa Językoznawczego*, **LX**, 133–143.

Przepiórkowski, A. (2009). A comparison of two morphosyntactic tagsets of Polish. In V. Koseska-Toszewa, L. Dimitrova, and R. Roszko, editors, *Representing Semantics in Digital Lexicography: Proceedings of MONDILEX Fourth Open Workshop*, pages 138–144, Warsaw.

Przepiórkowski, A. and Patejuk, A. (2012). On case assignment and the coordination of unlikes: The limits of distributive features. In Butt and King (2012), pages 479–489.

Przepiórkowski, A. and Patejuk, A. (2014). Koordynacja leksykalno-semantyczna w systemie współczesnej polszczyzny (na materiale Narodowego Korpusu Języka Polskiego). *Język Polski*, **XCIV**(2), 104–115.

Przepiórkowski, A. and Woliński, M. (2003). The unbearable lightness of tagging: A case study in morphosyntactic tagging of Polish. In *Proceedings of the 4th International Workshop on Linguistically Interpreted Corpora (LINC-03), EACL 2003*, pages 109–116.

Przepiórkowski, A., Kupść, A., Marciniak, M., and Mykowiecka, A. (2002). *Formalny opis języka polskiego: Teoria i implementacja*. Akademicka Oficyna Wydawnicza EXIT, Warsaw.

Przepiórkowski, A., Bańko, M., Górski, R. L., and Lewandowska-Tomaszczyk, B., editors (2012). *Narodowy Korpus Języka Polskiego*. Wydawnictwo Naukowe PWN, Warsaw.

Przepiórkowski, A., Hajnicz, E., Patejuk, A., and Woliński, M. (2014a). Extended phraseological information in a valence dictionary for NLP applications. In *Proceedings of the Workshop on Lexical and Grammatical Resources for Language Processing (LG-LP 2014)*, pages 83–91, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Przepiórkowski, A., Skwarski, F., Hajnicz, E., Patejuk, A., Świdziński, M., and Woliński, M. (2014b). Modelowanie własności składniowych czasowników w nowym słowniku walencyjnym języka polskiego. *Polonica*, **XXXIII**, 159–178.

Przepiórkowski, A., Hajnicz, E., Patejuk, A., Woliński, M., Skwarski, F., and Świdziński, M. (2014c). Walenty: Towards a comprehensive valence dictionary of Polish. In Calzolari *et al.* (2014), pages 2785–2792.

Raţiu, D. (2009). Coordintated questions vs. matching questions in Romanian. Abstract of a talk presented at ConSOLE XVII, Nova Gorica, 16–18 January 2009; `http://sabotin.ung.si/~jezik/console/abstracts/012.pdf`.

Rosén, V., Meurer, P., and Smedt, K. D. (2007). Designing and implementing discriminants for LFG grammars. In Butt and King (2007), pages 397–417.

Rouveret, A. and Vergnaud, J.-R. (1980). Specifying reference to the subject: French causatives and conditions on representations. *Linguistic Inquiry*, **11**(1), 97–202.

Sag, I. A. (2002). Coordination and underspecification. In J.-B. Kim and S. Wechsler, editors, *Proceedings of the HPSG 2002 Conference*, pages 267–291. CSLI Publications, Stanford, CA.

Sag, I. A., Gazdar, G., Wasow, T., and Weisler, S. (1985). Coordination and how to distinguish categories. *Natural Language and Linguistic Theory*, **3**, 117–171.

Saloni, Z. (2005). O przypadkach w języku polskim (na marginesie artykułu Adama Przepiórkowskiego). *Biuletyn Polskiego Towarzystwa Językoznawczego*, **LXI**, 27–48.

Saloni, Z., Woliński, M., Wołosz, R., Gruszczyński, W., and Skowrońska, D. (2012). *Słownik gramatyczny języka polskiego*. Warsaw, 2nd edition.

Saloni, Z. and Świdziński, M. (2001). *Składnia współczesnego języka polskiego*. Wydawnictwo Naukowe PWN, Warsaw, 5th edition.

Sannikov, V. Z. (1979). Sočinitel'nye i sravnitel'nye konstrukcii: ix blizost', ix sintaksičeskoe predstavlenie I. *Wiener Slawistischer Almanach*, **4**, 413–432.

Sannikov, V. Z. (1980). Sočinitel'nye i sravnitel'nye konstrukcii: ix blizost', ix sintaksičeskoe predstavlenie II. *Wiener Slawistischer Almanach*, **5**, 211–242.

Sulger, S., Butt, M., King, T. H., Meurer, P., Laczkó, T., Rákosi, G., Dione, C. B., Dyvik, H., Rosén, V., De Smedt, K., Patejuk, A., Cetinoglu, O., Arka, I. W., and Mistica, M. (2013). ParGramBank: The ParGram parallel treebank. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 550–560, Sofia, Bulgaria. Association for Computational Linguistics.

Szupryczyńska, M. (1996). Problem pozycji składniowej. In K. Kallas, editor, *Polonistyka Toruńska Uniwersytetowi w 50. Rocznicę Utworzenia UMK. Językoznawstwo*, pages 135–144. Wydawnictwo Uniwersytetu Mikołaja Kopernika, Toruń.

Świdziński, M. (1992). *Gramatyka formalna języka polskiego*, volume 349 of *Rozprawy Uniwersytetu Warszawskiego*. Wydawnictwa Uniwersytetu Warszawskiego, Warsaw.

Świdziński, M. (1993). Dalsze kłopoty z bezokolicznikiem. In J. Sambor, J. Linde-Usiekniewicz, and R. Huszcza, editors, *Językoznawstwo synchroniczne i diachroniczne*, pages 303–314. Wydawnictwa Uniwersytetu Warszawskiego, Warsaw.

Świdziński, M. (1994). Syntactic dictionary of Polish verbs. MS, Uniwersytet Warszawski and Universiteit van Amsterdam.

Świdziński, M. (1998). Syntactic dictionary of Polish verbs. Version 3a. MS, Uniwersytet Warszawski.

Świdziński, M. and Woliński, M. (2010). Towards a bank of constituent parse trees for Polish. In *Text, Speech and Dialogue: 13th International Conference, TSD 2010, Brno, Czech Republic*, Lecture Notes in Artificial Intelligence, pages 197–204, Berlin. Springer-Verlag.

Tajsner, P. (1990). *Case Marking in English and Polish: A Government and Binding Study*. Ph.D. dissertation, Uniwersytet im. Adama Mickiewicza, Poznań.

Tomaszewicz, B. (2011a). Against spurious coordination in multiple wh-questions. In *Proceedings of the 28th West Coast Conference on Formal Linguistics*, pages 186–195, Somerville, MA. Cascadilla Press.

Tomaszewicz, B. (2011b). Wh & wh: Syntactic and semantic arguments for clausal coordination. In *Formal Approaches to Slavic Linguistics 19*, pages 131–145, University of Michigan, Ann Arbor. Michigan Slavic Publications.

Trask, R. L. (1993). *A Dictionary of Grammatical Terms in Linguistics*. Routledge, London.

Vergnaud, J.-R. (1982). *Dépendances et Niveaux de Representation en Syntaxe*. Ph.D. dissertation, Université Paris 7.

Warren, D. H. D. and Pereira, F. C. N. (1980). Definite clause grammars for language analysis — a survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence*, **13**, 231–278.

Willim, E. (1990). On case-marking in Polish. *Papers and Studies in Contrastive Linguistics*, **XXV**, 203–220.

Woliński, M. (2004). *Komputerowa weryfikacja gramatyki Świdzińskiego*. Ph.D. dissertation, Institute of Computer Science, Polish Academy of Sciences, Warsaw.

Woliński, M. (2006). Morfeusz — a Practical Tool for the Morphological Analysis of Polish. In M. Kłopotek, S. T. Wierzchoń, and K. Trojanowski, editors, *Intelligent information processing and web mining*, pages 503–512. Springer-Verlag.

Woliński, M. (2010). Dendrarium — an Open Source Tool for Treebank Building. In M. Kłopotek, M. Marciniak, A. Mykowiecka, W. Penczek, and S. T. Wierzchoń, editors, *Intelligent Information Systems*, pages 193–204. Wydawnictwo Akademii Podlaskiej.

Woliński, M. (2014). Morfeusz reloaded. In Calzolari *et al.* (2014), pages 1106–1111.

Woliński, M., Głowińska, K., and Świdziński, M. (2011). A preliminary version of Składnica—a treebank of Polish. In Z. Vetulani, editor, *Proceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, pages 299–303, Poznań, Poland.