

**Case Matching Effects
in Free Relatives and Parasitic Gaps**

–

A Study on the Properties of Agree

An der Philologischen Fakultät der Universität Leipzig

eingereichte

DISSERTATION
zur Erlangung des akademischen Grades

Doctor philosophiae
(Dr. phil.)

von
Anke Himmelreich

Gutachter:

Prof. Dr. Gereon Müller
(Institut für Linguistik, Universität Leipzig)

Dr. habil. Fabian Heck
(Institut für Linguistik, Universität Leipzig)

Datum der Einreichung:
11. Juli 2017

Datum der Verleihung:
09. Oktober 2017

Contents

Acknowledgments **vii**

Abstract and Outline **xiii**

Abbreviations **xvii**

1. Introduction **1**

 1.1. Parasitic gaps 2

 1.2. Free Relatives 5

 1.3. Case Matching 7

I. An Analysis of Case Matching **9**

2. Introduction **11**

3. Data **13**

 3.1. German (Pittner (1995); Vogel (2001); Fanselow (1993); Kathol (2001)) 13

 3.2. Polish (Citko (2013)) 15

 3.3. Patterns 17

Contents

4. Analysis	21
4.1. The Structure	22
4.2. Case Agreement and Case Assignment	24
4.3. Agree	26
4.4. Variation	29
4.5. Derivations	34
4.5.1. German Parasitic Gaps	34
4.5.2. Polish Parasitic Gaps	37
4.5.3. German Free Relatives	40
4.5.4. Polish Free Relatives	43
4.5.5. Interim Summary	46
4.6. Bidirectional Agree and Opacity	47
5. Complicating the Pattern	51
5.1. Syncretisms	51
5.2. Case Hierarchy Effects	56
5.2.1. Case Hierachy Effects in German Free Relatives	57
5.2.2. Analysis	61
5.2.3. Derivations	64
5.3. More Patterns	70
5.3.1. German 2, Polish 2	72
5.3.2. Polish 1	76
5.3.3. (Modern) Greek	80
5.3.4. The Unknown Pattern	88

5.3.5. Interim Summary	92
6. Alternatives	95
6.1. Reanalysis	96
6.1.1. A Hybrid Account	96
6.1.2. Citko (2013)	102
6.2. Identity	103
6.2.1. Multidominance	104
6.2.2. Movement	105
6.3. Agreement	106
6.3.1. Only Upward Agree	106
6.3.2. Only Downward Agree	112
6.3.3. Only Syntactic Agree	114
6.3.4. Only Post-Syntactic Agree	116
7. Conclusion of Part I	119
 II. The Emergence of Agree in Free Relatives and Para-	
sitic Gap Constructions	123
 8. Agree in Parasitic Gap Constructions	125
8.1. An Analysis of Parasitic Gaps	126
8.1.1. Assumptions	127
8.1.2. Create-Dep	129
8.1.3. A Sample Derivation	136

Contents

8.2. Deriving the Behavior of Parasitic Gaps	140
8.2.1. Question I: Why are Parasitic Gaps ‘Selectively’ Sensitive to Islands?	140
8.2.2. Question II: Why Can Complex Antecedents Only be Recon- structed Into the Real Gap But Not the Parasitic Gap?	142
8.2.3. Question III: Why are Certain Properties of the Antecedent Only Important For Its Relation to the Real Gap But Not to the Parasitic Gap?	144
8.2.4. Question IV: Why Can Parasitic Gaps only be Licensed from Certain Positions?	148
8.3. Alternative Approaches	151
8.3.1. Binding	151
8.3.2. Operator Movement	152
8.3.3. Sideward Movement	154
8.4. Interim Summary	156
9. Agree in Free Relatives	157
9.1. An Analysis of Free Relatives	157
9.1.1. The Syntax of Free Relatives	158
9.1.2. Semantics of Free Relatives	163
9.2. Deriving the Behavior of the wh-Phrase	166
9.2.1. The wh-Phrase is Part of the Matrix Clause	166
9.2.2. The wh-Phrase is Part of the Embedded Clause	174

9.3. Alternative Approaches	176
9.3.1. External-Head Accounts	177
9.3.2. Comp Accounts	177
9.3.3. Multidominance	180
9.3.4. Derived-Head Accounts	181
9.4. Interim Summary	184
10. Conclusion of Part II	185
 III. Defining Agree	 187
11. Introduction	189
12. A Definition of Agree	191
12.1. A Definition of Agree-Link	191
12.2. A Definition of Agree-Copy	196
13. The Direction of Agree	199
13.1. Yes to Downward Agree	200
13.2. Yes to Upward Agree	201
13.3. No to Upward Agree	203
13.4. No to Downward Agree	207
13.5. Interim Summary	209
14. Conclusion	211
 References	 215

Acknowledgments

Things change. At some point, I thought I would never finish this thesis – and so, I guess, did a lot of the people around me. But things change. The fact that I was able to eventually put together all the research I have done over the years, all the ideas and data, all the definitions I have written and all the connections I slowly realized, still surprises me. But things change. And they wouldn't have changed without all the wonderful and inspiring people that accompanied me over the years. I'm truly grateful to *everyone* I have met along the way, because, in one way or the other, each of them has influenced me and each of them therefore owns a part of this thesis.

First, and foremost, I would like to thank Gereon Müller, who not only supervised this thesis, but who is the person that first thought I could do something in linguistics, even before I did. Over the years he has inspired, challenged and supported my work and he has always been patient for the results. Thank you, Gereon. My deepest gratitude also goes out to my second supervisor, Fabian Heck, who was the first person to think that I could do research in linguistics, even before I did. The joint research we did is also reflected in this thesis and I am happy that he taught me precision and perseverance.

The next people I would like to thank are Doreen Georgi, Philipp Weisser, and

Acknowledgments

Timo Klein – my fellow research assistants in LoMo. All of them were a lot faster in finishing their work but I cannot deny that being the last one of the pack also gave me the privilege to learn from their journeys. They are the pioneers. I also enjoyed our journey(s) together. The joint work we did in this research project with Gereon and Fabian shaped my skills as a linguist. Finally, I also appreciate their encouragement and all the fun they brought to my life outside of linguistics.

I would also like to thank all of the people I had the honor to work with and work around at our department: Sebastian Bank, Katja Barnickel, Laura Becker, Petr Biskup, Kristin Börjesson, Aaron Doliana, Imke Driemel, Siri Gjersøe, Martin Haspelmath, Johannes Hein, Daniela Henze, Stefan Keine, Sampson Korsah, Andrew Kostakis, Yuriy Kushnir, Martina Martinović, Mathías Guzman Naranjo, Andrew Murphy, Jegan Murugesan, Jude Nformi Awasom, Andreas Opitz, Ludger Paschen, Zorica Puškar, Marc Richards, Peter Staroverov, Barbara Stiebels, Jelena Stojković, Sandhya Sundaresan, Jochen Trommer, Joanna Zaleska, and Eva Zimmermann. One person is missing from this list because he deserves a special thanks: Martin Salzmann. Thanks, my fellow sufferer, for all your input on these horrible sharing constructions I've been dealing with.

There are also a number of people I met along the way who notably shaped my way of thinking and my research: Special thanks go to Gisbert Fanselow, Silke Fischer, Eric Fuß, Günther Grewendorf, Erich Groat, Greg Kobele, Tom McFadden, Andrew Nevins, Dennis Ott, Henk van Riemsdijk, Wolfgang Sternefeld, and the audiences at all the conferences where I have ever presented research of mine.

I'm greatly indebted also to the people who helped me with the data. First and

foremost, I would like to thank Heidi Klockmann, who provided me with a handout of Barbara Citko's work, which set the ball rolling for this thesis. Thanks to Artemis Alexiadou, Joanna Błaszczak, Anna Bondaruk, Marc Richards, Ewa Trutkowski, Joanna Zaleska, and all my fellow German speakers mentioned above for providing me with judgments.

The final thanks go out to everyone who supported me outside of linguistics, especially my husband Frank, who patiently supported me all these years.

Diese Dissertation ist meinen Eltern, Elke and Fred, gewidmet. Ich danke euch von tiefstem Herzen für alles, was ihr mir ermöglicht habt.

Abstract and Outline

The goal of this thesis is to show that the nature of the operation Agree is different from what is assumed in a standard derivational minimalist framework: First, Agree consists of two operations Agree-Link and Agree-Copy. Second, Agree has to be both upward and downward. There is no need to impose restrictions on the direction of Agree. The only restriction needed is c-command between probe and goal.

The first point has been discussed at great length in work by Karlos Arregi and Andrew Nevins, most prominently in Arregi and Nevins (2012). The data discussed in this thesis provide an additional argument for this fractioning of Agree into two operations.

The second point, the directionality of Agree, has received a lot of attention lately. In the standard definition of Agree in Chomsky (2000), the requirement that Agree has to be downward with the probe c-commanding the goal, is hard-wired as a constraint on Agree. I will challenge this assumption and show that Agree can be upward *and* downward.

The main argument that Agree should be defined in this specific way comes from case matching effects in free relatives and parasitic gap constructions. In the first part of the thesis, I will introduce the phenomenon and provide data from three

Abstract and Outline

languages: German, Polish, and Greek. Looking at case matching in these three languages and in the two constructions, it can be observed that there is not one universal case matching condition that holds for all constructions, but rather that case matching is construction-specific and can vary cross-linguistically. The main idea of the analysis is that case matching can be traced back to Agree and that case matching is just another empirical context for the matching condition of Agree. Crucially, this solution requires Agree to be defined as a two-step bidirectional operation.

The detailed study on case matching provides the basics to look at further aspects of Agree. In the second part, I will deal with the question why Agree relations exist. The answer that holds for Agree in the case of free relatives and parasitic gaps is that the probe features are introduced throughout the derivation. This part also contains a closer look at the syntax of these constructions and shows that an Agree-based approach to these two phenomena is highly promising.

In the final part of this thesis, I will provide a clear formal definition of Agree. This definition is based on an algorithmic understanding of derivations: derivations are programs that run and return a resulting structure. The Agree algorithm is part of this general program. Formalizing Agree in this way has the advantage that the conditions of Agree are clearly defined and are not based on imprecise formulations. In this last part, I will also argue that a bidirectional Agree operation is appropriate for virtually all instances of agreement.

The thesis makes two important new empirical contributions and three new theoretical contributions: As for the empirical part, the thesis works out the rather

complex pattern of case matching and it observes the previously undiscovered connection between parasitic gaps and free relatives, which both can be subsumed under the class of non-coordinative sharing constructions.

As for the theoretical contribution of this thesis, I show that the variation in case matching between constructions can be traced back to general properties of the constructions, especially the syntactic structure. Importantly, the universal absence of case hierarchy effects with respect to case matching in parasitic gaps follows systematically and does not have to be stipulated. A second major theoretical contribution is that the similarities between free relatives and parasitic gap constructions follow systematically since both can be derived in the same Agree-based approach to sharing. Finally, the last big contribution of this thesis is the formalization of Agree, especially the formalization of the operations Agree-Link and Agree-Copy in Arregi and Nevins (2012).

Abbreviations

1,2,3	1st, 2nd, 3rd person	ESSA	ongoing aspect
I, II, III	noun classes (Tsez)	FEM	feminine
∅	covert item	FR	free relative
α	overt item	FUT	future
A	A infinitive (Finnish)	G	gender
ABL	ablative	GEN	genitive
ABS	absolutive	IMPERF	imperfective
ACC	accusative	INF	infinitive
ANIM	animated	INSTR	instrumental
ARG	argument	ISL	island
C	case	LI	lexical item
CL	clitic	M	matrix clause
COP	copula	MASC	masculine
COMP	complementizer	N	number
DAT	dative	NEUTR	neuter
ERG	ergative	NMZL	nominalizer

Abbreviations

NOM	nominative
OP	operator
PARTCP	participle (default agr)
PERF	perfective
PG	parasitic gap
PL	plural
PRES	present
PRT	participle
PST	past
REL	relative (pronoun)
Q	question particle
SG	singular
SYN	syncretism
TOP	topic
WH	wh-phrase

1. Introduction

This thesis provides an in-depth study of case matching effects that occur in free relative and parasitic gap construction – two constructions that involve sharing of an argument phrase between two verbs. The starting point for this study will be a specific instance of case matching in German and Polish. These languages show an intricate pattern of linguistic variation that raises the question as to how this variation can be captured in a standard minimalist approach to syntax (Chomsky (1995) et seq.). Concretely, I will show that an account based on agreement is flexible enough to derive the variation that can be observed in the two languages.

Agreement relations in languages are usually captured by invoking the operation Agree that relates two items in the syntactic structure. One of the two items is analyzed as the probe which is underspecified with respect to a certain morphosyntactic feature and looks for a specification on another item, the goal, which can provide the specification – or the feature value – for the probe.

The main conclusion from the analysis of case matching effects is that the conditions on the application of Agree deviate from standard assumptions: Agree consists of a syntactic part and a post-syntactic part (an idea already proposed in Arregi and Nevins (2012)) and Agree is not confined by any restrictions on directionality.

1. Introduction

Based on the empirical study on case matching effects, I argue that Agree is bidirectional and applies in two steps. The thesis is structured as follows: In part I, the basic account for case matching is developed. Besides that, part I also contains a wider empirical look at case matching effects. This part ends with a summary and comparison of different approaches to free relatives and parasitic gaps, concluding that bidirectional Agree provides the best basis to capture case matching effects cross-linguistically.

In part II, I deal with the question as to how agreement in these constructions comes about. The main idea of this part of the thesis is that the agreement features in these constructions emerge in the course of the derivation. I will then continue to show how this idea captures the peculiar morphosyntactic behavior of free relatives and parasitic gaps.

Finally, in part III, I discuss the consequences of this theory of non-coordinative sharing constructions has for the theory of agreement in general. I will show that Agree can elegantly be defined algorithmically and I will dissect common arguments for different standpoints concerning the directionality of Agree. I conclude that restricting Agree to a certain direction faces empirical problems.

Before diving into the points mentioned above, the next two sections introduce the notions of parasitic gap and free relative.

1.1. Parasitic gaps

In parasitic gap constructions, there are two gaps that both seem to be the result of one movement operation. The two gaps are, however, not equal in status: One

gap depends on the existence of the other gap. In that sense, the dependent gap is parasitic. An example of a parasitic gap construction is given in (1). The difference between the two gaps is illustrated in (2).

- (1) Who would you like to meet $__1$ without first getting to know $__2$?
- (2) a. Who would you like to meet $__1$ without first getting to know *his mom*?
 b. *Who would you like to meet *his mom* without first getting to know $__2$?

As (2-a) shows, the gap $__1$ is grammatical despite the absence of the gap $__2$. In (2-b), however, $__1$ is missing and the sentence is ungrammatical. This shows that $__2$ is parasitic on $__1$.

Throughout this thesis, I mark the parasitic gap in these constructions as *pg*, while the non-parasitic gap is marked as *t*.¹

The parasitic gap construction was first mentioned by Ross (1967: 191ff.) and received its name in the early 80s by Taraldsen (1981) and Engdahl (1983), who provided the first major studies of these constructions. In what follows, parasitic gap constructions are defined as in (3) (cf. Taraldsen (1981); Engdahl (1983)).

(3) *Parasitic gap construction*

In a parasitic gap construction, the *parasitic gap* (*pg*) is licensed by the antecedent of another *true gap* (*t*) created by movement.

Constructions in several languages meet this definition. Consider the examples in

¹The marker “t” does not imply that an analysis of parasitic gap is based on a representational approach to dislocation phenomena.

1. Introduction

(4) and (5) from English and German, respectively.

(4) Which article did you file *t* [without reading *pg*]? (English)

(5) Welchen Artikel hast du [ohne *pg* zu lesen] *t* abgeheftet?
which article have you without to read filed
'Which article did you file without reading?' (German)

In these examples, the wh-phrase *which article* has moved from its base position in the matrix clause – which is the complement position of the verb *file* – to the left periphery of the sentence, thereby licensing a second gap in the complement position of the verb *read* inside the adverbial clause. This construction can be found in even more languages, some of which are listed in the examples below.

(6) Cuál informe archivaste *t* sin leer *pg*?
which article file.PST.2SG without read.INF
'Which article did you file without reading?' Spanish, Campos (1991: 118)

(7) nuw-ka enu sinmuwni-ul *pg* ikci anhko *t* chelhayss-ni?
who-NOM which newspaper-ACC read not file-PST-Q
'Which newspaper did you file without reading?' Korean, Lee (1998: 350)

(8) [Till himlen] är det inte säkert att alla som längtar *pg* kommer *t*.
To heaven it is not certain that everyone who longs get
'It is not certain that everyone who longs to (go to) heaven gets to go there.'
Swedish, Engdahl (1983: 17)

(9) devuška, kotoroj Ivan daval den'gi *t* do togo, kak (on) stal izbegat' *pg*
girl who Ivan gave money until (he) started to-avoid
'the girl who Ivan gave money to until he started to avoid her'
Russian, Franks (1993: 525)

1.2. Free Relatives

- (10) Raam-ne [Mohan-ke *pg* phadhne se pehle] kaunsa patr padh-liya tha?
Ram-ERG Mohan-of tearing of before which letter read be-PST
'Which letter did Ram read before Mohan tore off?'

Hindi, Malhotra (2009: 139)

- (11) Dit is het artikel waar ik *pg* over zei dat Harry een reactie *t* op
This is the article where I about said that Harry a reaction to
moest schrijven
should write

Dutch, Bennis and Hoekstra (1985: 61)

Despite the fact that the behaviors of parasitic gap constructions can vary, parasitic gap constructions in all languages show certain conspicuous properties. Those will be discussed in detail in part II. Over the last decades, a lot of theoretical possibilities have been explored in order to analyze this phenomenon. These approaches will also be summarized and discussed in part II.

1.2. Free Relatives

Like parasitic gaps, free relatives belong to a class of phenomena often called “sharing constructions”. In free relative constructions, the shared item in question is the *wh*-phrase, which appears to stand for the nominal head of the relative clause as well as the relative pronoun. Since it fulfills the function of both an item that is outside the embedded clause and an item that is inside the embedded clause, determining the position of this phrase presents an interesting challenge for syntacticians: the *wh*-phrase seems to be part of the embedded clause as well as the matrix clause.

An example of a free relative clause in German is given in (12).

1. Introduction

- (12) Ich werde niemandem zeigen was ich gefunden habe.
I will nobody.DAT show what I found have
'I won't show to anybody what I found.' (Ott 2011: 184)

(12) shows the two most prominent properties of free relatives that distinguish them from normal headed relative clauses: the lack of an external nominal head and the form of the relative pronoun, which is realized as a *wh*-phrase. Given the assumption that the relative pronoun moves to the left periphery, the structure of the example in (12) is as in (13).

- (13) Ich werde niemandem —₁ zeigen [_{FR} was ich —₂ gefunden habe].
I will nobody.DAT show what I found have
'I won't show to anybody what I found.' (Ott 2011: 184)

This structure is similar to the one of parasitic gaps insofar as there are two gaps which are associated with one overt item, *was*, in (13). The discussion about free relatives in part II will focus on this similarity and compare free relatives to the parasitic gap construction. This comparison includes a discussion of the properties of free relatives as well as a discussion of the theoretical approaches to free relatives.

One of the first comprehensive studies about this construction was done by Bresnan and Grimshaw (1978). Over the time, various ideas have been proposed to derive free relatives. The basic strategies, however, are the same as for parasitic gap constructions. Given that this phenomenon received attention about the same time as parasitic gaps, it is all the more surprising that both these constructions have not been related before.

Free relatives can be found in various languages. Some examples are given in (14)

(taken from Caponigro (2003)).

- | | | |
|------|---|-----------------|
| (14) | I'll marry [_{FR} who you choose]. | <i>English</i> |
| | | |
| (15) | Ammiro [_{FR} chi lavora duramente].
admire.1SG who works hard
'I admire those who work hard.' | <i>Italian</i> |
| | | |
| (16) | Ma imetlen [_{FR} keda sina imetled].
I admire who you admire
'I admire those who you admire.' | <i>Estonian</i> |
| | | |
| (17) | Ani ma'aric [_{FR} mi še-oved kaše].
I admire who COMP-works hard
'I admire the one/those who work(s) hard.' | <i>Hebrew</i> |
| | | |
| (18) | Ik bewonder [_{FR} wie hard werkt].
I admire who hard works
'I admire who works hard.' | <i>Dutch</i> |

Similar to parasitic gaps, the general properties of free relatives are identical cross-linguistically. There are, however, also differences between languages, first and foremost differences concerning the case of the *wh*-phrase.

1.3. Case Matching

Case matching is the name of a phenomenon where two cases have to be identical in order to get a grammatical structure. This phenomenon is a common property of sharing construction, where one item has to realize two different cases. An example of case matching in across-the-board constructions is given in (19).

1. Introduction

- (19) *Czego / *Co Jan nienawidzi _{—gen} a Maria lubi _{—acc}
 what.GEN / what.ACC Jan hates and Maria likes
 ‘What does Jan hate and Maria like?’ *Polish*, Citko (2005)

Originally, the term *matching effect* comes from studies on free relatives (Grimshaw (1977); Bresnan and Grimshaw (1978); Groos and Riemsdijk (1981)). Case matching effects are thus a subtype of matching effects. Matching can also concern other types of morphological marking. Other features that can be subject to matching are, for example, categorial and ϕ -features, shown in the German example in (20).

- (20) Ich zahle [_{FR} wie viel du bezahlst hast].

 I pay how much you have paid.

 ‘I’m paying as much as you have paid.’

The phenomenon of case matching is reviewed and analyzed in detail in part I.

Part I.

An Analysis of Case Matching

2. Introduction

The aim of this part is to explore the case matching property of free relatives and parasitic gaps in more detail. It is shown that the matching condition is subject to linguistic variation. More concretely, it is shown that case matching can vary within languages between constructions and between languages in the same construction.

The first set of data to consider come from German and Polish. The comparison of both languages will show that Polish is the mirror image of German when it comes to case matching effects: In German case mismatches are allowed with free relatives but not with parasitic gaps, in Polish case mismatches are allowed with parasitic gaps but not with free relatives. Afterwards, an analysis based on Agree is developed that is able to derive the differences between German and Polish. Finally, the system is extended and revised in order to capture all the pattern of case matching.

3. Data

This section shows in detail why Polish is considered the mirror image of German when it comes to case matching effects. First, the pattern of German is described. Afterwards the facts in Polish are introduced. The chapter concludes with a summary of the patterns shown in the data.

3.1. German (Pittner (1995); Vogel (2001); Fanselow (1993); Kathol (2001))

The examples in (21) show that the cases of the parasitic gap and the antecedent have to match in German.² In (21-a), both the parasitic gap and the antecedent bear accusative case. Thus, the sentence is grammatical. (21-b) shows that it is also possible to have dative case in both positions. In (21-c-d), the cases of the antecedent and the parasitic gap differ. Both sentences are ungrammatical. The cases in (21-e) differ, too, but this does not cause ungrammaticality. In contrast to (21-c,d), the differing cases in (21-e) are syncretic. This implies that the morphological cases rather than the syntactic cases is subject to the matching condition.

²Note that the parasitic gaps in (21) are licensed by scrambling. This is possible in German according to Felix (1985); Bennis and Hoekstra (1985); Webelhuth (1992); Mahajan (1990).

3. Data

(21) *Parasitic gaps: strict case matching*

- a. weil Hans die_{acc} Frau [ohne anzusehen_{acc}] geküsst_{acc} hat
because Hans the woman without to.look.at kissed has
‘because Hans has kissed the woman without looking at her’
- b. weil Hans der_{dat} Frau [anstatt zu helfen_{dat}] schadete_{dat}
because Hans the woman instead.of to help hurt
‘because Hans hurt the woman instead of helping her’
- c. weil Hans *der_{dat}/*die_{acc} Frau [anstatt zu helfen_{dat}]
because Hans the woman instead.of to help
behinderte_{acc}
hampered
‘because Hans hampered the woman instead of helping her’
- d. weil Hans *der_{dat}/*die_{acc} Frau [anstatt zu behindern_{acc}]
because Hans the woman instead.of to hamper
half_{dat}
helped
‘because Hans helped the woman instead of hampering her’
- e. weil Hans der_{gen/dat} Verstorbenen [anstatt ein Gedicht zu
because Hans the dead.one instead.of a poem to
widmen_{dat}] in einer Gradrede gedachte_{gen}
dedicate in a eulogy commemorate
‘because Hans commemorated the dead one in a eulogy instead of dedi-
cating a poem to her’

Turning to free relative clauses in German, we can observe a different pattern of case matching. (22-a) and (22-b) show that if the cases assigned to the wh-phrase by the embedded verb and the matrix verb are identical, the resulting sentence is fine, just as with parasitic gaps in (21-a-b). The sentences in (22-c-d) show that free relatives differ from parasitic gaps in the matching condition. While in (22-d), the differing cases lead to ungrammaticality just as in (21-d), the case mismatch in (22-c) can be resolved if the wh-phrase bears dative case, which is the case of the

embedded clause. This contrasts with (21-c), where the same case mismatch could not be resolved in a parasitic gap configuration. Finally, (21-e) shows that a case mismatch with two syncretic cases yields an acceptable sentence as well.

(22) *Free relatives: absence of strict case matching (with case restrictions)*

- a. Hans mag_{acc} [wen_{acc} (auch immer) Maria hasst_{acc}].
Hans likes who ever Maria hates
‘Hans likes whoever Maria hates.’
- b. Hans hilft_{dat} [wem_{dat} (auch immer) er vertraut_{dat}].
Hans helps who ever he trusts
‘Hans helps whoever he trusts.’
- c. Hans mag_{acc} [*wen_{acc}/wem_{dat} (auch immer) Maria vertraut_{dat}].
Hans likes who ever Maria trusts
‘Hans likes whoever Maria trusts.’
- d. Hans vertraut_{dat} [*wen_{acc}/*wem_{dat} (auch immer) Maria mag_{acc}].
Hans trusts who ever Maria likes
‘Hans trusts whoever Maria likes.’

3.2. Polish (Citko (2013))

The case matching pattern in Polish is different from the one in German. (23) shows this for parasitic gaps.³ As can be seen in (23-a) and (23-b), parasitic gaps in Polish are possible with accusative and dative case. (23-c-d) show that the case mismatches that weren’t allowed in German (21) are fine in Polish. Finally, (23-e) shows again

³All the following data are drawn from Citko (2013) and checked with three native speakers of Polish. Although the parasitic gap construction is only marginally acceptable for some speakers, two of the three speakers could confirm the basic intuitions about the acceptability difference between the two constructions. For one speaker the entire construction was impossible.

3. Data

that a syncretism in a case mismatch is fine.

(23) *Parasitic gaps: absence of strict case matching*

- a. To jest dziewczyna, która_{acc} Jan tolerował_{acc} [zanim polubił_{acc}].
this is girl which Jan tolerated before liked
'This is the girl Jan tolerated before he grew to like.'
- b. To jest dziewczyna, której_{dat} Jan towarzyszył_{dat} [zanim zaczął
this is girl which Jan accompanied before started
pomagać_{dat}].
help
'This is the girl who Jan kept company before he started to help.'
- c. To jest dziewczyna, która_{acc}/*której_{dat} Jan lubił_{acc} [zanim zaczął
this is girl which Jan liked before started
pomagać_{dat}].
help
'This is the girl Jan liked before he started to help.'
- d. To jest dziewczyna, której_{dat}/*która_{acc} Jan ufał_{dat} [zanim
this is girl which Jan trusted before
polubił_{acc}].
liked
'This is the girl Jan trusted before he got to like.'
- e. To jest dziewczyna, ?która_{acc/instr} Jan ignorował_{dat} [zanim się
this is girl which Jan ignored before REFL
zainteresował_{acc}].
interested
'This is the girl Jan ignored before he got interested in her.'

The data in (24) show the matching pattern for free relatives in Polish. If the cases are identical, as in (24-a) and (24-b), the resulting sentence is grammatical. However, if the two cases are not the same, as in (24-c) and (24-d), the free relative is not grammatical. The example in (24-e) confirms that, if the two mismatching cases are morphologically identical, a conflict can be prevented.

(24) *Free relatives: strict case matching*

- a. Jan lubi_{acc} [kogokolwiek_{acc} Maria lubi_{acc}].
 Jan likes whoever Maria likes
 ‘Jan likes whoever Maria likes.’
- b. Jan pomaga_{dat} [komukolwiek_{dat} ufa_{dat}].
 Jan helps whomever trusts
 ‘Jan helps whomever he trusts.’
- c. Jan lubi_{acc} [*kogokolwiek_{acc}/*komukolwiek_{dat} dokucza_{dat}].
 Jan likes whoever teases
 ‘Jan likes whoever he teases.’
- d. Jan ufa_{dat} [*komukolwiek_{dat}/*kogokolwiek_{acc} wpuścił_{acc} do domu].
 Jan trusts whoever let to home
 ‘Jan trusts whoever he let into the house.’
- e. Jan unika_{gen} [kogokolwiek_{gen/acc} wczoraj obraził_{acc}].
 Jan avoids whoever yesterday offended
 ‘Jan avoided whoever he offended yesterday.’

3.3. Patterns

The patterns we have seen in sections 3.1 and 3.2 are summarized in the tables in (25) to (28). In the columns, the cases assigned in the embedded clauses are listed; in the rows, the cases of the matrix clause are shown. The remarks in brackets indicate under which circumstances a case conflict can be resolved. In parasitic gap configurations in German and in free relatives in Polish, only a case syncretism can help to prevent ungrammaticality. In free relative configurations in German, on the other hand, case mismatches are fine as long as the *wh*-phrase bears the case of the embedded clause. The only mismatch that is not allowed is a configuration where the case in the embedded clause is accusative while the case of the matrix clause is dative.⁴ Similarly, in Polish case mismatches in parasitic gap structures are allowed

⁴This example is discussed at length in section 5.2.

3. Data

if the overt antecedent bears the case of the matrix clause.

(25) *Parasitic gaps in German*

PG \ M	Acc	Dat	Gen
Acc	✓	*	*
Dat	*	✓	✓(syn)
Gen	*	✓(syn)	✓

(26) *Free relatives in German*

FR \ M	Acc	Dat	Gen
Acc	✓	✓(dat)	✓(gen)
Dat	*	✓	✓(gen)
Gen	*	*	✓

(27) *Parasitic gaps in Polish*

PG \ M	Acc	Dat	Gen
Acc	✓	✓(acc)	✓(acc)
Dat	✓(dat)	✓	✓(syn)
Gen	✓(gen)	✓(syn)	✓

(28) *Free relatives in Polish*

FR \ M	Acc	Dat	Gen
Acc	✓	*	✓(syn)
Dat	*	✓	*
Gen	✓(syn)	*	✓

The patterns allow for the following observations: First, German and Polish are mirror images of each other when it comes to case matching: FR that lack case matching effects in German, show them in Polish, while parasitic gaps that allow case mismatches in Polish show strict matching in German. Second, free relatives and parasitic gaps are mirror images of each: if one of the two configurations show case matching effects, the other does not. These two observations are summarized in the simplified pattern in (29).

(29) *Case mismatches*

	<i>German</i>	<i>Polish</i>
<i>Parasitic gaps</i>	*	✓
<i>Free relatives</i>	✓	*

Finally, we have seen that, in both languages, syncretic forms can repair violations of the case matching condition. Thus, what seems to count for matching are not the abstract case features but rather the morphological form. Consequently, under the assumption that the morphological form does not count for narrow syntax, the case matching condition cannot be a principle of narrow syntax (see Trommer (2002); Riemsdijk (2006) for the same conclusion).

4. Analysis

In this chapter, I develop an analysis that captures the pattern we have seen so far. The analysis is based on the idea that the case matching effects observed in chapter 3 are instances of general matching effects as they occur with agreement. That is, case matching is seen as a result of case agreement (see also Himmelreich (2016)). After laying down the technical details and going through some sample derivations, the analysis will be modified in chapter 5 in order to derive further patterns of case matching. Additionally, the analysis will serve as the basis for a closer look at the parasitic gap and the free relative construction in part II and a theory of Agree in part II.

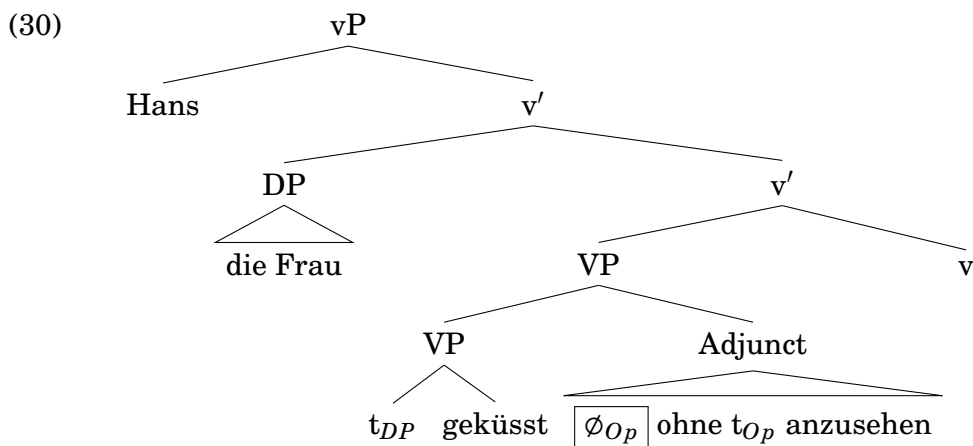
The analysis of the data is carried out in a derivational minimalist framework (Chomsky (1995) et seq.) combined with a derivational modular version of Distributed Morphology (Arregi and Nevins (2012); Halle and Marantz (1993)). In order to derive the patterns of case matches and case mismatches, the following points will play an important role: the syntactic structures of parasitic gap and free relative constructions, the directionality of Agree and the order of Agree operations.

4.1. The Structure

As for the first point, the syntactic structures, I assume that the “sharing” of the overt category in free relative and in parasitic gap constructions can be derived if there are actually two elements in the structure. Each of these elements independently receives a case value, but have to agree in case. Since, on the surface, only one of the two categories is overt, the other one has to be phonologically empty.

For parasitic gaps, this idea has more or less been the standard theory ever since (Chomsky (1982); Engdahl (1983); Kayne (1983); Chomsky (1986); Cinque (1990); Nissenbaum (2000)).⁵

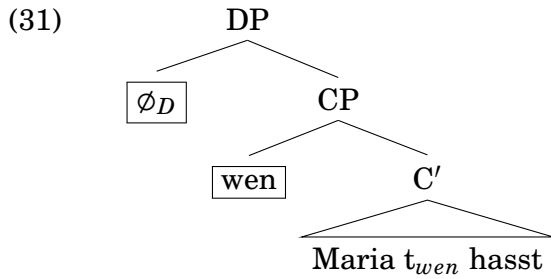
In part II, chapter 8, I will go into more detail as to what the nature of this phonologically empty element is. For now, it suffices to adopt the idea of Chomsky (1986) (see also Lee (1998); Nissenbaum (2000)), which takes the covert element to be an operator in the left periphery of the embedded clause. Given these assumptions, a German example like (21-a) has the structure shown in (30).



⁵But see Nunes (2004) for an approach to parasitic gaps that does without covert categories.

The main point of this analysis is that the phonologically empty element \emptyset_{Op} is generated inside the parasitic gap clause, where it receives case. From this position, it moves up to the embedded Spec-CP position. The gap created by movement of the operator is the parasitic gap. In the Spec-CP position, the operator can enter into a syntactic relation with the antecedent in the matrix clause. As discussed in chapter 8, this syntactic relation is agreement between \emptyset_{Op} and its antecedent. The antecedent is a phrase that is generated as an object in the matrix clause and moves to a position that c-commands the empty operator. In the example in (30), the DP *die Frau* is the object of the matrix verb *küssen*. This DP is scrambled to Spec-vP, a position that c-commands \emptyset_{Op} .

Turning to free relatives, a large number of analyses assume that the structure contains a covert element as well (see e.g. Bresnan and Grimshaw (1978); Groos and Riemsdijk (1981)).⁶ The exact position of this element has been subject to much dispute. In the present analysis, I will adopt the approach by Groos and Riemsdijk (1981) that builds on the idea that the covert element fills the position of the D head.⁷ Under this assumption, the structure for (22-a) is as shown in (31).



⁶Equally, people have proposed that only the wh-phrase is present in free relatives, e.g. Citko (2005),

Riemsdijk (2006); Donati and Cecchetto (2011).

⁷I will deal with the emergence of this element in part II, chapter 9.

4. Analysis

In (31), the wh-phrase *wen* moves from its base position as the direct object of the embedded verb *hassen* to the Spec-CP position of the embedded clause. Then, the CP is merged with the covert D head, resulting in a relative clause structure. At this point, the D head is c-commanding the wh-phrase and the two can enter into a syntactic relation. Just as for parasitic gaps, I assume that this relation is agreement (see part II, chapter 9).

In sum, the present analysis is based on the assumption that the relevant elements in parasitic gap and free relative configurations do not receive case from both the embedded and the matrix clause. Instead there is one overt element and one covert element, which independently receive cases, but have to agree in case features.

4.2. Case Agreement and Case Assignment

Assuming that the covert and the overt element in parasitic gap and free relative structures receive case independently, there must be some sort of case agreement between the two categories. Otherwise the matching effects cannot be derived. In the following, I propose that the case agreement is part of a more general agreement relation between the two elements. Such an agreement has already been proposed in previous approaches (see Assmann (2012) for parasitic gaps, Grosu (2003); Assmann (2013); Grewendorf and Groat (2013) among others for free relatives). The nature and the emergence of this agreement relation is explored further in part II.

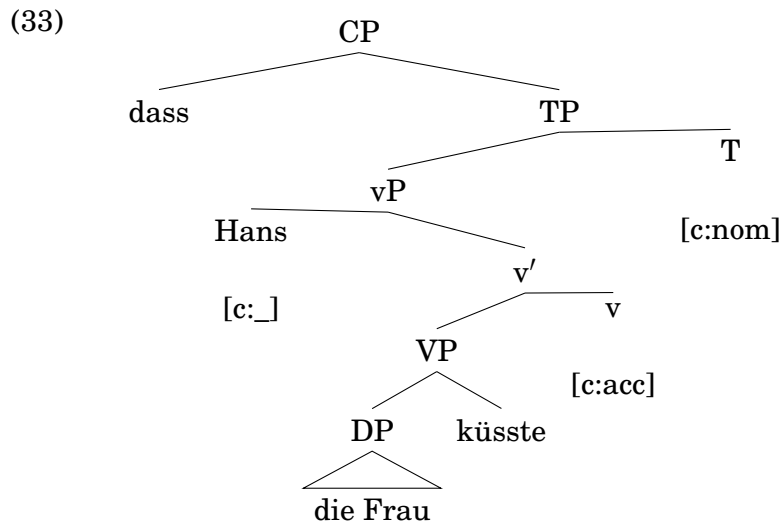
For now, one assumption about this agreement relation is crucial: Only one of the two acts as the probe. In that sense, the agreement relation between the two heads

is asymmetric.⁸ The agreement relation between the overt and the covert element in parasitic gap and free relative constructions is thus an instance of a standard agreement relation. Henceforth, asymmetric agreement relations are depicted as in (32), with an arrow leading from the probe to the goal.

$$(32) \quad X[F:_] \longrightarrow Y[F:v]$$

In (32), the head X carries a probe feature for agreement that needs to be valued, while the head Y carries the valued goal feature.

I further assume that case assignment from a verbal head to its argument is agreement in case features as well. Thus case assignment is not seen as a reflex of ϕ -feature agreement, but as a separate agreement relation. To illustrate this, (33) shows the structure of a simple transitive clause.



In (33), v agrees with the direct object *die Frau*, valuing its case feature with

⁸This assumptions will be revised for German in chapter 5.2. For now, the assumption suffices to derive the simplified pattern.

4. Analysis

accusative. T agrees with the subject *Hans*, assigning nominative case.

Both case agreement in sharing constructions and case assignment by functional heads can be implemented as Agree operations: the case features from a case assigning head are copied onto a head that probes for case features. In what follows, I will specify the properties of the operation Agree in detail.

4.3. Agree

Following Arregi and Nevins (2012), Agree consists of two operations: a syntactic Agree-Link and a post-syntactic Agree-Copy. Agree-Link establishes a syntactic relation between the probe and the goal. Post-syntactic Agree-Copy copies the case values from the probe onto goal. Note that the fact that the success of case agreement depends on the morphological form and not the abstract Case features strongly suggests that at least part of the case dependency must be post-syntactic.

The assumption that Agree cannot be entirely post-syntactic is confirmed by the fact that the case matching effects do not disappear if the covert category in parasitic gap or free relative constructions is not in a surface c-command relation with its antecedent. This is exemplified in (34) with a free relative in German. Even though, in some cases, case matching is not required, there are cases where a mismatch seems impossible. The relevant example is repeated in (34).

- (34) Hans vertraut_{dat} [*wen_{acc}/*wem_{dat} (auch immer) Maria mag_{acc}].
Hans trusts who ever Maria likes
'Hans trusts whoever Maria likes.'

Changing the tense to past⁹ shows that the free relative can actually be extraposed. Since, in German, DPs cannot be extraposed, it must be the CP that is moved, leaving behind the covert D head. But then, the D head no longer c-commands the wh-phrase. Still, a mismatch is not possible. Assuming that a mismatch is the result of an unsuccessful Agree relation and that Agree requires c-command, it follows that Agree must take place before CP-extrapolation.

- (35) Hans hat vertraut_{dat} [DP D_∅ t_{CP}] [CP *wen_{acc}/*wem_{dat}
 Hans has trusted who ever Maria likes
 (auch immer) Maria mag_{acc}].

‘Hans trusted whoever Maria likes.’

Thus, there is evidence that case agreement takes place quite early in syntax as well as evidence that it applies post-syntactically. These facts fit in well with the theory proposed in Arregi and Nevins (2012) where it is argued that Agree is not one operation but needs to be split up in a search operation Agree-Link and a valuation operation Agree-Copy. In chapter 12, I will develop a formal definition of these Agree operations.

There are two conditions that govern successful Agree: one concerning the positions of the probe and the goal and one concerning the properties of the probe and the goal. The first of these conditions applies to the syntactic Agree-Link and is defined in (36) (adapted from (Haegeman 1994: 147)).

⁹Note that past tense is usually realized as perfect tense with the auxiliary *haben* in German.

4. Analysis

(36) *C-Command*

A node A c-commands node B if neither A dominates B nor B dominates A and if the first branching node that dominates A also dominates B.

Given this condition, the direction in which Agree applies does not play a role: either the probe c-commands the goal or the goal c-commands the probe. Hence, there is upward as well as downward agreement for case features (see Baker (2008*b*); Zeijlstra (2012); Toosarvandani and van Urk (2014) for further instances of upward Agree). This assumption will receive further attention in part III, where I discuss the advantages of a bidirectional Agree operation.

The second condition is a matching condition that concerns Agree-Copy. Concretely, I assume that Agree-Copy is only successful if the case feature value of probe and goal do not conflict. Obviously, a conflict cannot arise if one of the two features is still unvalued. Thus, if a probe Agrees with two goals, the second Agree relation can potentially fail if the probe bears a feature value that the second goal cannot match.

The final point about Agree concerns the order of Agree operations. As Agree-Link applies in syntax, the order of Agree-Link operations is governed by Earliness (Pesetsky (1989); Řezáč (2004*b*)). Post-syntactic Agree-Copy cannot rely on Earliness because the structure is already built. Therefore, I would like to propose that the order is determined by the syntactic structure, which is still present at the stage when Agree-Copy applies (see Arregi and Nevins (2012)). Concretely, case probes that are lower in the structure receive their case features first, that is, Agree-Copy proceeds bottom-up. If a category probes twice for case features, which happens in free relative and parasitic gap constructions due to the additional Agree relation,

the order is free.¹⁰ With these assumptions in place, we can now turn to the origin of variation in case matching.

4.4. Variation

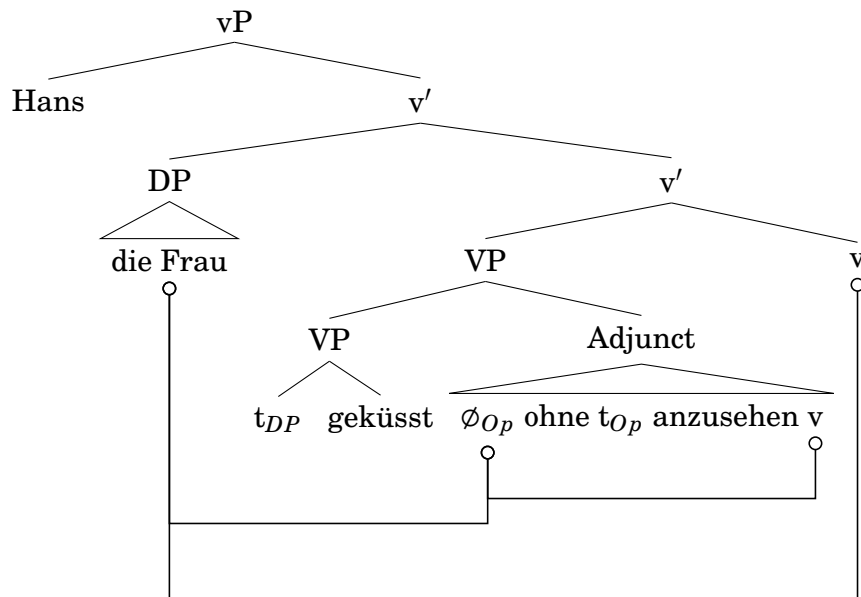
The constraints on Agree and Case assignment proposed above hold cross-linguistically. What can vary, however, is the configuration of probe and goal in the additional Agree relation in sharing constructions. In section 4.2, I proposed the idea that there is an additional case agreement relation in parasitic gap and free relative constructions between the overtly shared element and the covert category that I assume to be present in these categories. Because of this additional case agreement relation, case features must in principle be able to probe twice: once to receive case from a functional (verbal) head, such as *v* or *T*, and once to match their value against the case value of the second category in sharing constructions. This situation is abstractly depicted in (37).¹¹

¹⁰Note that this assumption will also be revised in section 5.2. For the simplified pattern, however, no further restriction is needed.

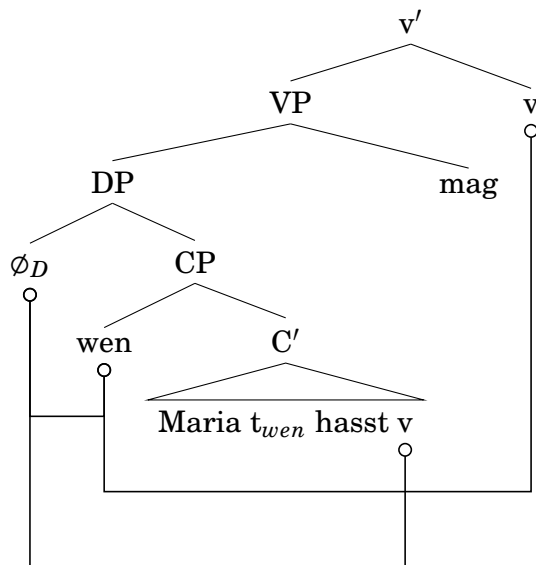
¹¹Note that this creates an instance of Multiple Agree Hiraiwa (2001). This, in turn, does not mean that case features are in general capable of Multiple Agree. In part II, we will see that the second Agree relation actually results from the derivation of these specific sharing constructions.

4. Analysis

(37) a. *Parasitic gaps*



b. *Free relatives*



In this example of a German parasitic gap constructions in (37-a), the antecedent agrees with both matrix **v** and the operator in the embedded clause. The operator agrees with the embedded **v**. Similarly, in the free relative configuration in (37-b), the wh-phrase agrees with the embedded **v** and the covert **D** head. The **D** head

receives case from matrix v.

Now, in order to derive the differences in case matching between German and Polish, I assume that the focus of the variation lies in this additional Agree relation: It can vary between constructions and, as I would like to suggest, also between languages. Concretely, the difference between German and Polish is whether the overt element or the covert element triggers case agreement in free relatives and parasitic gaps: In German, case agreement is triggered by the overt element, that is, the antecedent of a parasitic gap or the *wh*-phrase in free relatives.¹² In Polish, on the other hand, the case probe sits on the covert item.

This means that in German, the case feature of the overt item in a parasitic gap or free relative construction has a double probe property: it probes for a case feature on a case assigning head and it probes for the case value of the empty element in these configurations. In Polish, it is the covert item that probes for the case assigning head and the case feature of its antecedent. Henceforth, this double probe property will be depicted as [**c:**]. “*_*” stands for “I need a value”, while “** **” stands for “I want to probe a second time” (see Sternefeld (2006) for the notation ** **). Intuitively, in these structures case features need to be valued and additionally checked.

The proposed parameter leads to the four possible configurations shown in (38). Each configuration corresponds to one of the four patterns discussed in chapter 3.

¹²See section 5.2 for a more elaborate analysis of German.

4. Analysis

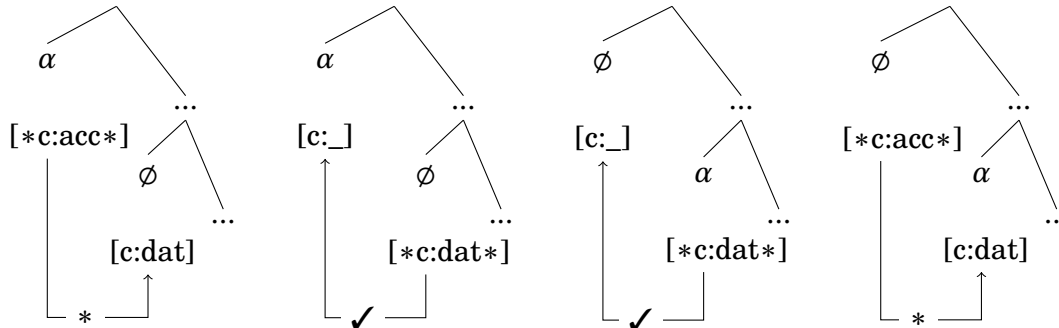
(38)

a. *German PG*

b. *Polish PG*

c. *German FR*

d. *Polish FR*



The structures show the case features at the time when the case feature on α/\emptyset undergoes post-syntactic Agree-Copy a second time. α stands for the overt element, while \emptyset is the covert item in the structure.

In German parasitic gap constructions (38-a), the overt antecedent is higher up in the structure than the empty operator. Therefore, the empty operator receives all its case values before the antecedent. Furthermore, we have downward Agree between the two categories because, in German, the antecedent probes for the case value of the empty operator. This brings us to a conflict in configurations where the two cases are not identical. In (38-a), the empty operator has received dative case from the embedded verb, its antecedent has received accusative case from the matrix verb. At the point when the antecedent wants to copy the case value of the empty operator, the matching condition on Agree is violated. Therefore, the case feature of the antecedent cannot be checked a second time and the derivation crashes.

In Polish (38-b), on the other hand, we have upward Agree in this configuration because the empty operator is the probe. At the time, the empty operator probes for the case value of the antecedent, the antecedent has not received a case value

yet. Thus, matching is trivially fulfilled as seen in (38-b). The result of this upward Agree operation is empty valuation on the empty operator. Later on, the antecedent receives its case value, but this value comes too late to cause a problem with Agree between antecedent and operator. Thus case mismatches show up on the surface.

Turning to free relatives, we see that the direction of Agree is reversed with respect to parasitic gaps. The reason for this is that in the structure of free relatives, the covert category (the D head) is higher in the structure. Therefore, it is the overt category that receives its case value first. Thus, in German free relatives (38-c) – because the overt item is the probe – we have upward Agree. Like in Polish parasitic gap structures, upward Agree leads to empty valuation with the result that case mismatches can be tolerated. In Polish free relatives, however, we have downward Agree since in Polish, the empty category is the probe.

In sum, the main idea of the analysis is that if the lower of the two elements is the probe, we have upward case agreement that results in empty valuation because the higher goal has not received its case features yet. Consequently, the case feature value of the higher goal will not count for matching and mismatches are allowed. If, on the other hand, the higher of the two elements is the probe, the lower element has already received its case feature value. In this case, both the case feature value of the higher probe and the lower goal will count for matching and strict case matching is required.

4.5. Derivations

In order to fully understand why downward Agree leads to strict case matching while upward Agree makes mismatching possible, the four types of derivations established in (38) are exemplified. We start with German parasitic gaps.

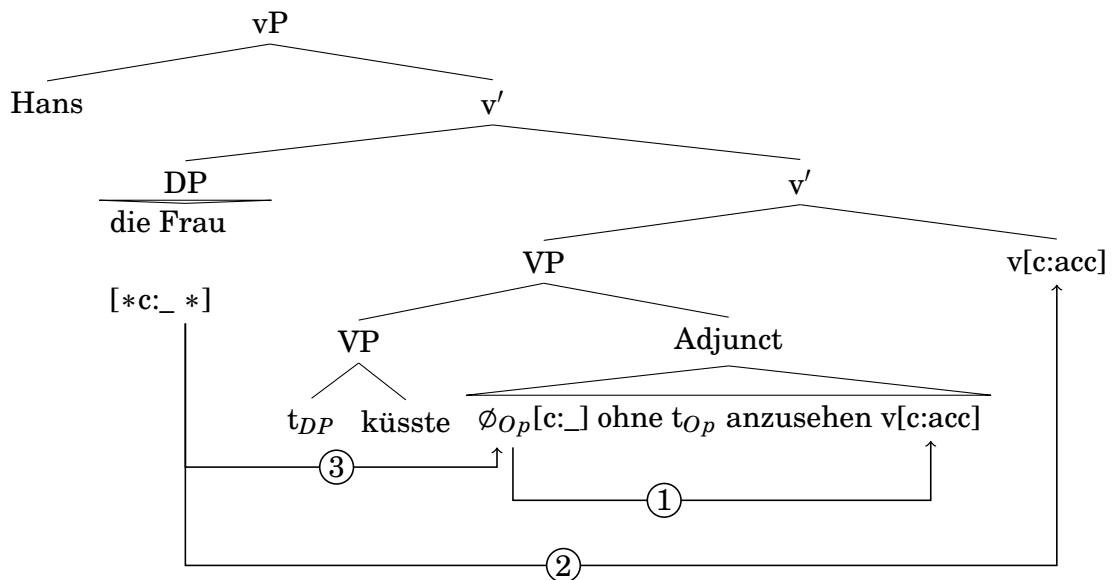
4.5.1. German Parasitic Gaps

To see how the proposed system works, (40)–(42) shows the crucial steps in the derivation of an example with matching case. The relevant example is repeated in (39).

- (39) weil Hans die_{acc} Frau [ohne anzusehen_{acc}] küsste_{acc}
 because Hans the woman without to.look.at kissed

The structure in (40) shows how the syntactic Agree-Link relations are established.¹³

- (40) *Syntactic Agree-Link*



¹³For convenience, only the Agree-Link relations relevant for case matching are shown in this structure.

At first, the covert operator ϕ_{Op} in the embedded clause Agrees with the embedded v and moves to the left periphery of the clause. In the matrix clause, the antecedent Agrees with the matrix v head. Afterwards, the antecedent *die Frau* is scrambled to matrix Spec-vP, where it c-commands ϕ_{Op} . In this configuration a third Agree-Link relation can be established between the *die Frau* and ϕ_{Op} .

In the post-syntactic component, Agree-Copy applies and the case features of the antecedent and the operator are valued. This valuation proceeds bottom-up with the lowest probe being valued first. For case agreement, this means that, in parasitic gap structures, ϕ_{Op} receives its accusative from the embedded v before the antecedent *die Frau* receives a value. This step is shown in (41).

$$(41) \quad \text{Step I: } \phi[c:_] \longrightarrow v[c:acc]: \phi[c:acc]$$

Next, the case feature of the antecedent is valued. Since its case feature is a double probe, there are two possible orders of case assignment. Both orders lead to the same result, as shown in (42). (Note that α stands for the antecedent.)

$$(42) \quad \begin{array}{lll} \text{Step II:} & \alpha[*c:_] & \longrightarrow v[c:acc]: \alpha[*c:acc*] \\ & \text{or } \alpha[*c:_] & \longrightarrow \phi[c:acc]: \alpha[*c:acc*] \\ \\ \text{Step III:} & \alpha[*c:acc*] & \longrightarrow \phi[c:acc]: \alpha[c:acc] \\ & \text{or } \alpha[*c:acc*] & \longrightarrow v[c:acc]: \alpha[c:acc] \end{array}$$

In the first option, the antecedent receives accusative case from matrix v and checks it afterwards against the accusative case value of ϕ_{Op} . Since there is no mismatch, Agree-Copy applies successfully both times.

4. Analysis

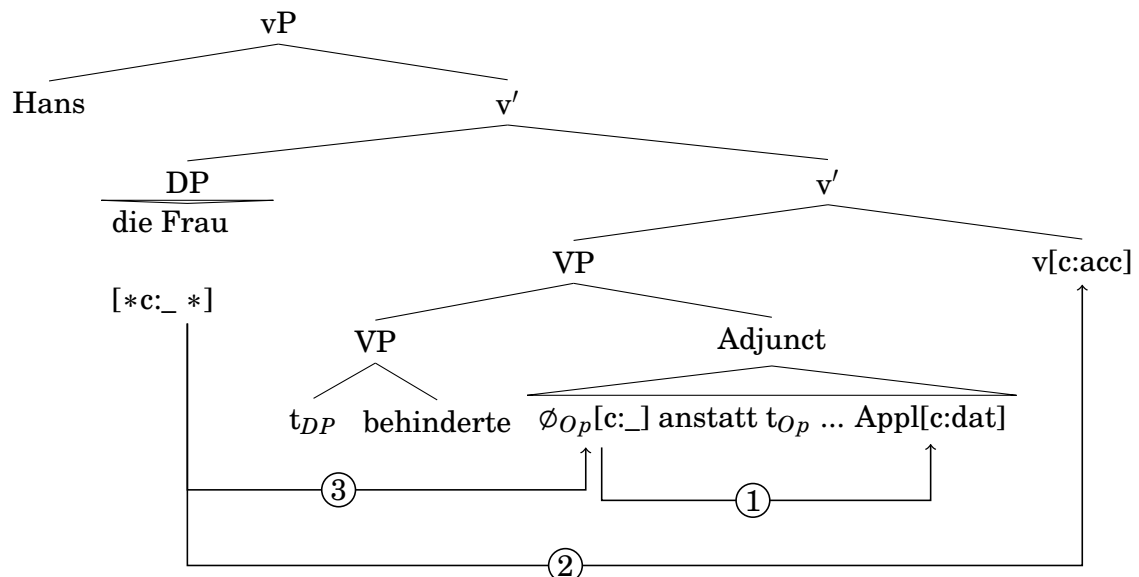
In the alternative order, the antecedent first receives accusative from \emptyset_{Op} and Agrees with matrix v afterwards. Again, since all the relevant case values are identical, Agree between the antecedent and the operator is possible.

Next, we see that if the two relevant cases are not identical, Agree between the antecedent and the operator in German parasitic gaps fails. The example that illustrates this configuration is repeated in (43).

- (43) weil Hans *der_{dat}/*die_{acc} Frau [anstatt zu helfen_{dat}]
 because Hans the woman instead.of to help
 behinderte_{acc}
 hampered

The syntactic derivation of (43) works as in 44. The difference lies in the cases being assigned. In the derivation of (43), \emptyset_{Op} receives dative case in the embedded clause. For concreteness, dative case is assigned by an empty applicative head Appl in the derivation in (44). Alternatively, dative can be assigned by an empty preposition or some other functional head. Nothing hinges on that.

- (44) *Syntactic Agree-Link*



The difference in the post-syntactic derivation is that one of the Agree-Copy operations cannot apply due to a violation of the matching condition on Agree. The two possible orders of the relevant Agree-Copy operations are shown in (45).

(45)	Step I:	$\emptyset[c: _]$	\longrightarrow	Appl[c:dat]:	$\emptyset[c:dat]$
<hr/>					
	Step II:	$\alpha[*c: _ *]$	\longrightarrow	v[c:acc]:	$\alpha[*c:acc*]$
	or	$\alpha[*c: _ *]$	\longrightarrow	$\emptyset[c:dat]:$	$\alpha[*c:dat*]$
	Step III:	$\alpha[*c:acc*]$	\longrightarrow	$\emptyset[c:dat]:$	$\not\rightarrow$
	or	$\alpha[*c:dat*]$	\longrightarrow	v[c:acc]:	$\not\rightarrow$

In the first step, the \emptyset_{Op} receives dative case in the embedded clause. Afterwards, the antecedent (*Hans*) probes for the matrix v head and the \emptyset_{Op} . If it gets accusative case from v first, as shown in the highest row in (45-Step II), a mismatch arises when it tries to Agree a second time with the dative bearing \emptyset_{Op} . Reversing the order as in the lower rows does not help either because after receiving dative case from \emptyset_{Op} , the antecedent can no longer Copy the accusative value v due to a violation of the matching condition.

In sum, the fact that in the final Agree-Copy operation, both the probe and the goal bear a case value, requires identity of the values. Otherwise the matching condition is violated.

4.5.2. Polish Parasitic Gaps

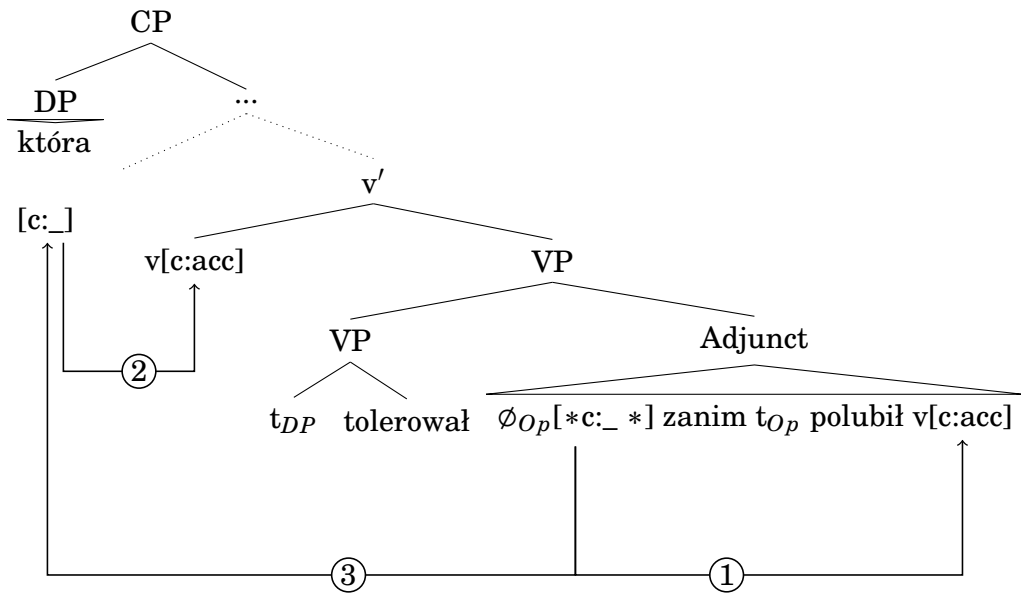
The difference between Polish and German is illustrated with the example in (46).

4. Analysis

- (46) To jest dziewczyna, która_{acc} Jan tolerował_{acc} zanim polubił_{acc}.
 this is girl which Jan tolerated before liked.

The syntactic derivation of parasitic gaps in Polish is basically the same as in German. The crucial difference between Polish and German is that, in Polish, \emptyset_{Op} bears the case feature that probes twice. Thus, we have upward agreement in parasitic gap constructions in Polish. This is shown in (47).

- (47) *Syntactic Agree-Link*



The two possible orders of post-syntactic Agree-Copy operations are given in (48).

- (48) Step I: $\emptyset[*c:_ *] \rightarrow v[c:acc]: \emptyset[*c:acc*]$
 or $\emptyset[*c:_ *] \rightarrow \alpha[c:_]: \emptyset[c:_]$
 Step II: $\emptyset[*c:acc*] \rightarrow \alpha[c:_]: \emptyset[c:acc]$
 or $\emptyset[c:_] \rightarrow v[c:acc]: \emptyset[c:acc]$

 Step III: $\alpha[c:_] \rightarrow v[c:acc]: \alpha[c:acc]$

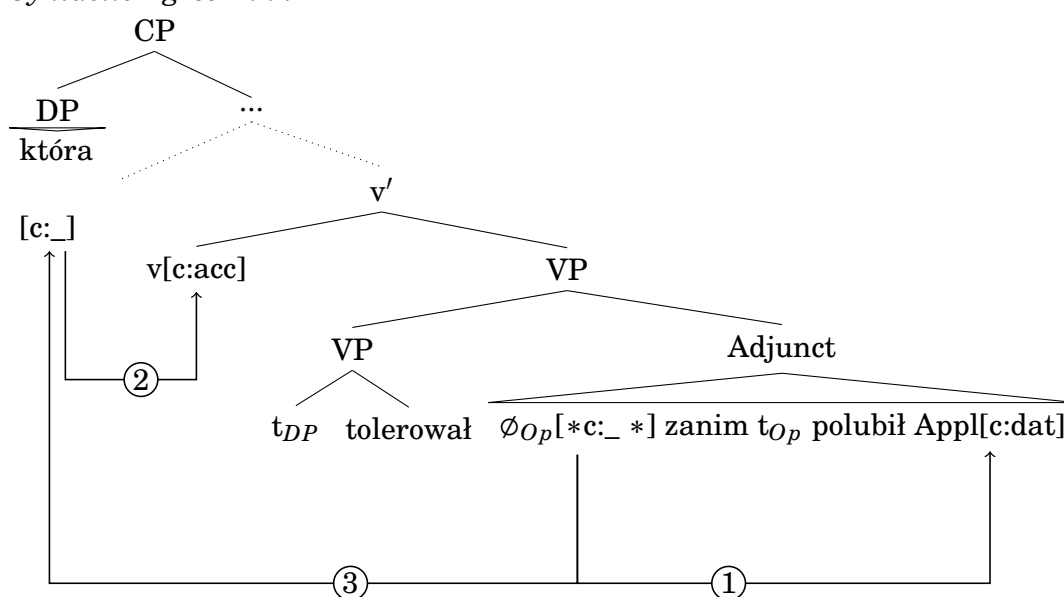
At first, \emptyset_{Op} copies the value from the embedded v and the antecedent *która*. Independent of the order between the two Agree-Copy operations, the \emptyset_{Op} ends up with an accusative case value from the embedded v because *która* does not bear a value at this point. In the final step, *która* receives accusative case from matrix v .

The difference between German and Polish becomes obvious if we look at parasitic gap structures where the cases assigned in the matrix and in the embedded clause are not identical. The relevant example from section 3.2 is repeated in (49).

- (49) To jest dziewczyna, która_{acc} / *której_{dat} Jan lubił_{acc} zanim zaczął
this is girl which Jan liked before started
pomagać_{dat}.
help.

In contrast to German, a mismatch in case features is tolerated in Polish parasitic gaps. This results from the the probe in the α - \emptyset agreement relation being lower than the goal, as shown in (50).

- (50) *Syntactic Agree-Link*



4. Analysis

In the post-syntactic component, upward Agree will result in empty valuation and the matching condition is trivially fulfilled. The Agree-Copy steps are shown in (51).

$$\begin{array}{llll}
 (51) \quad \text{Step I:} & \emptyset[*c:_*] & \longrightarrow & \text{Appl}[c:\text{dat}]: \emptyset[*c:\text{dat}*] \\
 & \text{or } \emptyset[*c:_*] & \longrightarrow & \alpha[c:_]: \emptyset[c:_] \\
 \\
 & \text{Step II:} & \emptyset[*c:\text{dat}*] & \longrightarrow \alpha[c:_]: \emptyset[c:\text{dat}] \\
 & \text{or } \emptyset[c:_] & \longrightarrow & \text{Appl}[c:\text{dat}]: \emptyset[c:\text{dat}] \\
 \hline
 & \text{Step III:} & \alpha[c:_] & \longrightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]
 \end{array}$$

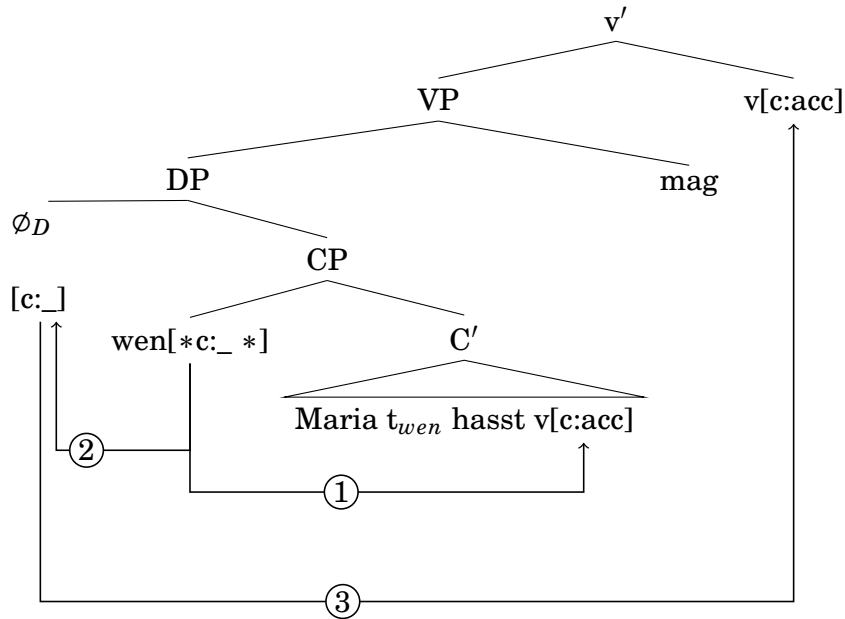
The derivation in (51) is basically the same as in (48), the only difference being that \emptyset_{Op} receives dative from an applicative head. Crucially, however, since *która* receives its accusative case feature in Step III after Agree with \emptyset_{Op} , it cannot conflict with the dative feature on \emptyset_{Op} . Thus, a violation of the matching condition is prevented by upward Agree.

4.5.3. German Free Relatives

After having shown explicitly how variation in case matching in parasitic gap constructions is derived, the next two derivations show how case matching effects in free relatives come about. The first example in (52) is from German.

$$\begin{array}{ll}
 (52) & \text{Hans mag}_{acc}, \text{ wen}_{acc} \text{ Maria hasst}_{acc}. \\
 & \text{Hans likes} \quad \text{who} \quad \text{Maria hates} \\
 & \text{'Hans likes who Maria hates.'}
 \end{array}$$

The syntactic derivation is summarized in the tree in (53).

(53) *Syntactic Agree-Link*

At first, the wh-phrase *wen* is merged as the object of the embedded clause where it establishes an Agree relation with the embedded v . Once the entire relative clause is built, *wen* moves up to the embedded Spec-CP position. After the D head is merged, *wen*, being overt, probes upward for the D head. Finally as the structure-building continues, the D head Agrees with matrix v .

The crucial difference to parasitic gaps in German is that, this time, the overt item is lower in the structure. Since, by assumption, the overt item carries the probe property in German, it is the overt item that probes for the covert item.

Turning to the post-syntactic part of Agree, we see that (54) is highly reminiscent of the derivation of parasitic gaps in Polish: The lower of the two categories, this time the overt one, probes for the higher category. Thus, we have upward Agree, which results in empty valuation. The derivation is shown in (54).

4. Analysis

- (54) Step I: $\alpha[*c:_*] \rightarrow v[c:acc]: \alpha[*c:acc*]$
or $\alpha[*c:_*] \rightarrow \emptyset[c:_]: \alpha[c:_]$
- Step II: $\alpha[*c:acc*] \rightarrow \emptyset[c:_]: \alpha[c:acc]$
or $\alpha[c:_] \rightarrow v[c:acc]: \alpha[c:acc]$
-
- Step III: $\emptyset[c:_] \rightarrow v[c:acc]: \emptyset[c:acc]$

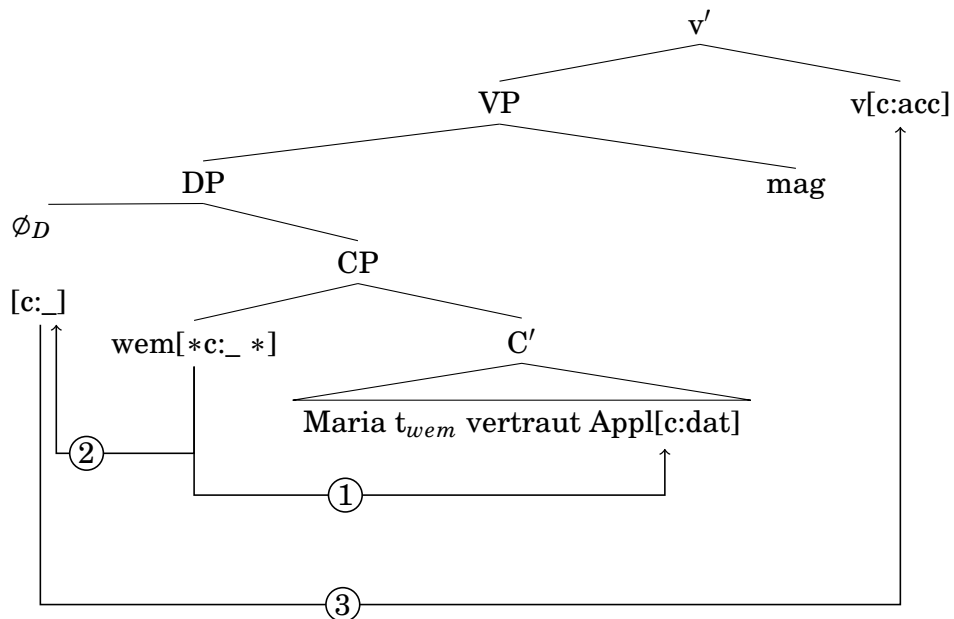
First, *wen* receives accusative from the embedded v. Since the D head does not bear a case value yet, *wen* ends up with accusative case from v after Step II. After *wen* has received its case feature values, the accusative case value on the D head is copied from matrix v (Step III).

That German free relatives allow case mismatches, just like Polish parasitic gaps, is shown with the example in (55).

- (55) Hans mag_{acc}, *wen_{acc}/wem_{dat} Maria vertraut_{dat}.
Hans likes who Maria trusts

The derivation of (55) is the same as in (53), but this time, the wh-phrase *wem* receives dative case from an applicative head in the embedded clause.

- (56) *Syntactic Agree-Link*



As in (51), the reason why this dative case value on *wem* does not conflict with the accusative case on the D head is that Agree-Copy between *wem* and D applies before D receives its value in Step III. That is, at the point where the matching condition of Agree applies, no conflict is given.

$$\begin{array}{llll}
 (57) \quad \text{Step I:} & \alpha[*c:_*] & \longrightarrow & \text{Appl}[c:\text{dat}]: \alpha[*c:\text{dat}*] \\
 & \text{or } \alpha[*c:_*] & \longrightarrow & \emptyset[c:_]: \alpha[c:_] \\
 \\
 & \text{Step II:} & \alpha[*c:\text{dat}*] & \longrightarrow \emptyset[c:_]: \alpha[c:\text{dat}] \\
 & \text{or } \alpha[c:_] & \longrightarrow & \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}] \\
 \hline
 & \text{Step III:} & \emptyset[c:_] & \longrightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]
 \end{array}$$

Being the lower of the two probes, the overt wh-phrase in the embedded clause, receives dative case from Appl first. Agree-Copy between the wh-phrase and the D head does not change that since no value is copied in this step. In the final step, the D head receives accusative case from the matrix v head. Again, the reason why the case mismatch here is tolerated, is that the matching condition of Agree between the covert and the overt item applies before both items have received a case value.

4.5.4. Polish Free Relatives

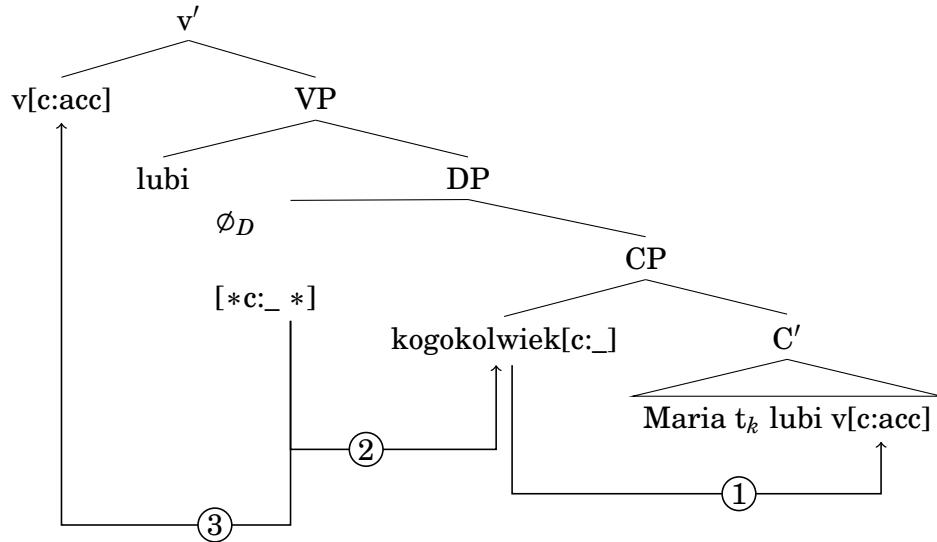
Finally, the derivation of Polish free relatives resembles the derivation of German parasitic gaps. An example for a Polish free relative is repeated in (58).

$$\begin{array}{llll}
 (58) & \text{Jan} & \text{lubi}_{acc} & \underline{\text{kogokolwiek}_{acc}} & \text{Maria} & \text{lubi}_{acc}. \\
 & \text{Jan likes} & & \text{whoever} & & \text{Maria likes}
 \end{array}$$

4. Analysis

In contrast to German free relatives, the probe of the additional case agreement relation between the covert D head and the wh-phrase is the covert D head, which is higher in the structure than the wh-phrase. This is shown in (59).

(59) *Syntactic Agree-Link*



The derivation in (59) works just like the one in (53) for German. The wh-phrase *kogokolwiek* merged as the object of *lubi* in the embedded clause enters into Agree-Link with the embedded v. Afterwards, it is moved to the Spec-CP position. When the D head is merged, a second Agree-Link relation between *kogokolwiek* and the D head is established. In contrast to German, the covert D head is the probe. Thus, the difference between German and Polish is that there is downward Agree between the D head and the wh-phrase. Otherwise, just as in (53), the D head has another Agree relation with the matrix v head.

The fact that there is downward Agree instead of Upward Agree in Polish leads to a different postsyntactic derivation, as shown in (60).

- (60)
- | | | | | |
|-----------|-------------------------|-------------------|------------------|----------------------|
| Step I: | $\alpha[c:_]$ | \longrightarrow | $v[c:acc]:$ | $\alpha[c:acc]$ |
| | | | | |
| Step II: | $\emptyset[*c:_*]$ | \longrightarrow | $\alpha[c:acc]:$ | $\emptyset[*c:acc*]$ |
| | or $\emptyset[*c:_*]$ | \longrightarrow | $v[c:acc]:$ | $\emptyset[*c:acc*]$ |
| Step III: | $\emptyset[*c:acc*]$ | \longrightarrow | $v[c:acc]:$ | $\emptyset[c:acc]$ |
| | or $\emptyset[*c:acc*]$ | \longrightarrow | $\alpha[c:acc]:$ | $\emptyset[c:acc]$ |

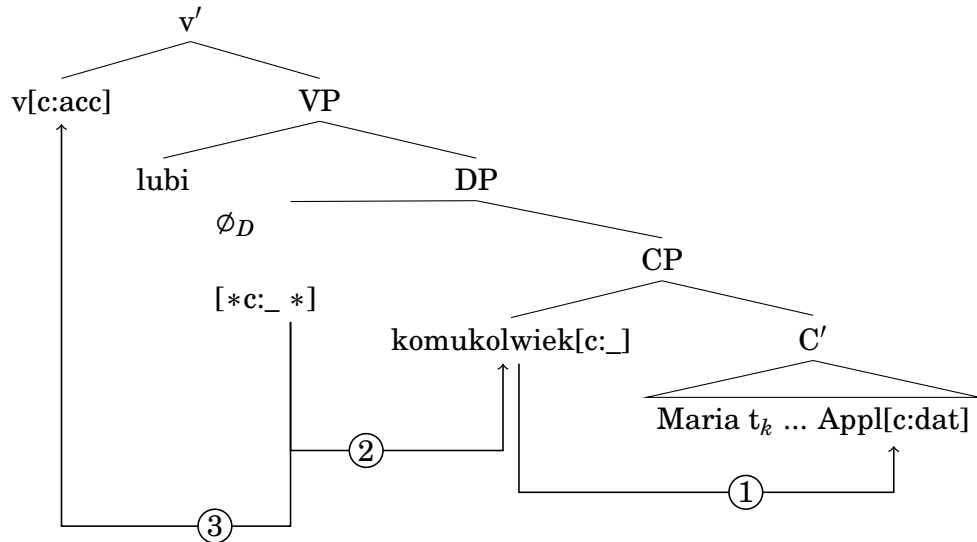
In (61), *kogokolwiek* receives accusative from the embedded *v* first. Afterwards, the D head gets case values from *kogokolwiek* and the matrix *v*. This is unproblematic as all the case values are the same.

However, if the cases assigned to \emptyset_D and the wh-phrase differ, the case mismatch will violate the matching condition. This is the case for the example in (61).

- (61) Jan lubi_{acc} *kogokolwiek_{acc}/?*komukolwiek_{dat} dokucza_{dat}.
 Jan likes whoever teases

The derivation in (62) shows why this is the case. In contrast to (59), an applicative head Appl Agrees with the wh-phrase in the embedded clause.

- (62) *Syntactic Agree-Link*



4. Analysis

The relevant Agree-Copy operations in the postsyntactic derivation are shown in (63).

(63)	Step I:	$\alpha[c:_]$	\longrightarrow	Appl[c:dat]:	$\alpha[c:dat]$
<hr/>					
	Step II:	$\emptyset[*c:_*]$	\longrightarrow	$\alpha[c:dat]:$	$\emptyset[*c:dat*]$
	or	$\emptyset[*c:_*]$	\longrightarrow	$v[c:acc]:$	$\emptyset[*c:acc*]$
	Step III:	$\emptyset[*c:dat*]$	\longrightarrow	$v[c:acc]:$	$\not\rightarrow$
	or	$\emptyset[*c:acc*]$	\longrightarrow	$\alpha[c:dat]:$	$\not\rightarrow$

First, *komukolwiek* receives dative case from the embedded applicative head. Then, the D head, being a double probe, has to decide, whether its first case value comes from *komukolwiek* or from the matrix v. If the first option is chosen, the D head receives dative case from *komukolwiek*, but cannot receive accusative case from matrix v anymore, due to a mismatch. On the other hand, if it receives accusative first, it cannot match the dative case of *komukolwiek*. This is reminiscent of the derivation of parasitic gaps in German.

4.5.5. Interim Summary

So far, we have seen that the case matching condition is subject to variation between constructions and between languages. Concretely, we have seen that German parasitic gap constructions and Polish free relative constructions adhere to the matching condition, while German free relatives and Polish parasitic gaps do not require matching. In this chapter, I have shown that this pattern can be derived if Agree applies in two steps and if the directionality of Agree is not restricted.

With these assumptions in place, the variation we have seen can be reduced to one parameter: the position of the probe of case agreement in parasitic gap and free relative. In cases where the position of the probe leads to downward Agree, case mismatches will inevitably violate the matching condition while, in instances of upward Agree, the matching condition applies before a case mismatch is established.

4.6. Bidirectional Agree and Opacity

The analysis developed in the previous sections gives rise to opacity in the sense of Kiparsky (1973, 1976): At the surface, both the overt category and the covert category bear case feature values. But only in some of the examples, conflicting case values cause a violation of the matching condition. These structures therefore involve opaque interactions of Agree operations.

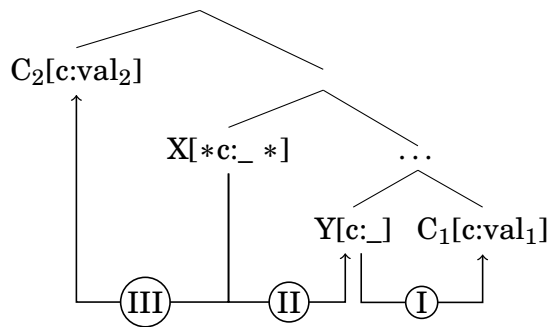
There are four basic kinds of opaque interaction: feeding, bleeding, counter-feeding, and counter-bleeding. A feeding interaction is given if the application of one operation O_1 creates the context for the application O_2 . In a bleeding interaction, the application of O_1 destroys the context of O_2 . If the application of O_1 could feed O_2 but does not feed it actually, this interaction is called counter-feeding. Finally, the term counter-bleeding describes an interaction where the application of O_1 could bleed O_2 but does not do so.

In the analysis above, the opacity of case matching is resolved by ordering the Agree-Copy operations. In all derivations there are three Agree-Copy relations that are important: (i) the relation between the lower element Y and a case assigning head C_1 , (ii) the relation between the higher element X and another case assigning

4. Analysis

head C_2 and (iii) the relation between X and Y. If the first two Agree-Copy relations result in feature valuation, the third relation is only successful if the two case values are identical. Otherwise, the matching condition of Agree would be violated. Thus, Agree between X or Y and $C_{1/2}$ can bleed Agree between X and Y. This bleeding configuration is illustrated in (64) and (65).

(64)



(65) Step I: $Y[c:_] \rightarrow C_1[c:val_1]: Y[c:val_1]$

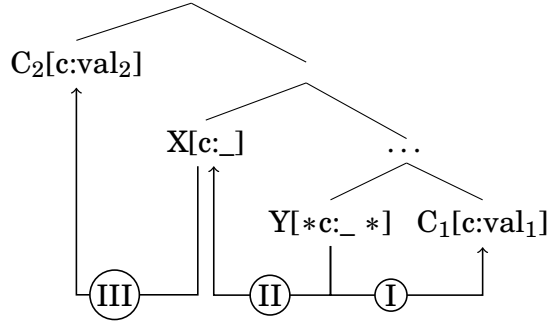
Step II: $X[*c:_*] \rightarrow C_2[c:val_2]: X[*c:val_2*]$

Step III: $X[*c:val_2*] \rightarrow Y[c:val_1]: \not\downarrow$

In the first two steps, X and Y receive case values independently from each other. Afterwards, X and Y cannot Agree because the matching condition is violated. This is what happens in German parasitic gap and Polish free relative constructions.

In the counter-bleeding configuration in (66) (67), X and Y Agree, before both have a case values. Thus matching is trivially fulfilled for Agree between X and Y. This is the case for Polish parasitic gap and German free relative constructions.

(66)

(67) Step I: $Y[*c:_ *] \rightarrow C_1[c:val_1]: Y[*c:val_1 *]$ Step II: $Y[*c:val_1 *] \rightarrow X[c:_]: Y[*c:val_1 *]$ Step III: $X[c:_] \rightarrow C_2[c:val_2]: X[c:val_2]$

In the second Agree-Copy operation in (67), the goal X does not bear a value. Therefore, the matching condition is trivially fulfilled.

The order of Agree-Copy operations, that is whether we have a bleeding or a counter-bleeding interaction, is determined by the position of the probe in the X-Y-Agree operation. Since lower probes are valued before higher probes due to the bottom-up nature of the post-syntactic derivation, Y will receive its case feature value first because it is lower in the structure than X. On the other hand, if Y is the probe in the X-Y-Agree operation, it copies the case from X before X actually receives a value, which results in empty feature valuation and trivially fulfills the matching condition independent from the case value X receives later. Thus, if there is upward Agree between X and Y, the matching condition is counter-bled (cf. Georgi (2014)). On the other hand, if there is downward Agree, that is, if the higher head X is the probe, Agree between X and Y will inevitably lead to a non-trivial application of the matching condition. Thus, a case mismatch between X and Y results in a bleeding configuration.

5. Complicating the Pattern

The pattern described and analyzed in chapter 4 is just the simplified version of an even more complex pattern of case matching. In this chapter, the entire complexity of the pattern is revealed and analyzed. Overall, there are three aspects that contribute to the increase of complexity: (i) the facts that syncretic forms do not cause a case mismatch effect, (ii) the fact that there are case hierarchy effects in German free relatives but not in parasitic gap construction, and (iii) the assumption that the languages vary as to where the probe in case agreement positions sit. In what follows, I turn to each of these aspects, culminating in a typology of case matching effects.

5.1. Syncretisms

In all four configurations discussed above, syncretic forms can remedy a violation of case matching. Thus, it seems that it is the morphological form and not the abstract case feature that is crucial for the matching effects. The split of Agree into a syntactic Agree-Link operation and a post-syntactic Agree-Copy operation can capture this fact without invoking a lot of special assumptions.

The examples in (68) repeat the syncretism effects in Polish free relatives. The

5. Complicating the Pattern

free relative clause in (68-b) should be ungrammatical due to a mismatch between accusative and genitive case. But since the wh-phrase *kogokolwiek* has a case form that matches both the accusative and the genitive, the matching violation is repaired.

- (68) a. To jest dziewczyna, której_{gen/dat} Jan się bał_{gen} zanim zaczął
this is girl which Jan REFL fear before started
pomagać_{dat}.
help.
'This is the girl Jan was afraid of before he started to help.'
- b. Jan lubi_{acc} kogokolwiek_{acc/gen} Maria nienawidzi_{gen}.
Jan likes whoever Maria hates
'Jan likes whoever Maria hates.'

Similarly, in the German example (69-a), a syncretic form can help with a case mismatch in parasitic gap constructions.

- (69) a. weil Hans der_{gen/dat} Verstorbenen [anstatt ein Gedicht zu
because Hans the dead.one instead.of a poem to
widmen_{dat}] in einer Gradrede gedachte_{gen}
dedicate in a eulogy commemorate
'because Hans commemorated the dead one in a eulogy instead of dedi-
cating a poem to her'
- b. Hans mag_{acc} was_{acc/nom} Maria zur Weißglut treibt_{nom}.
Hans likes what Maria to rage drives
'Hans likes whatever infuriates Maria.'

(69-b) also shows the use of a syncretic case form in German free relatives. The wh-phrase *was* is ambiguous between a nominative form and an accusative form. Again, the mismatch is fine.

In order to integrate these facts into the system developed in chapter 4, the question as to how syncretisms can be handled in general, must be answered. In the framework of Distributed Morphology, syncretic forms result from special morphological rules that alter the morphosyntactic features before the phonologically exponents are inserted via Vocabulary Insertion rules. For example, syncretisms can be due to language-specific feature changing syncretism rules (cf. Noyer (1992: 129)).¹⁴ Relevant examples for such rules in Polish are given in (70).

(70) *Syncretism rules in Polish*

- a. [c:acc] → [c:gen]/[anim:+]
- b. [c:dat] → [c:gen]/[rel:+], [g:fem]

At a certain point in the post-syntactic derivation, these rules apply and change the case features left of the arrows to the feature right of the arrow. Crucially, these rules apply only in a certain context. As for the rule (70-a), the accusative case feature becomes a genitive case feature if the head bearing this case feature is also animate. As for the rule (70-b), the dative becomes a genitive, if the head bearing the dative is a feminine relative pronoun. Thus, rule (70-a) applies in the example (68-b) and the rule (70-b) applies in the example (68-a).

Next, the point of the derivation where these rules apply must be specified. Probably the most intuitive assumption is that they apply as soon as their contexts are given. This point is reached once the case features are valued by applying Agree-Copy. The consequence for the derivations of parasitic gap and free relative constructions

¹⁴Note that it might be possible to reduce these special rules to impoverishment rules.

5. Complicating the Pattern

is that the syncretism rules apply before the last of the three Agree-Copy operations applies. In that sense, the syncretism rules must apply early (see Trommer (2002); Keine (2010) for similar conclusions).

Furthermore, the condition under which agreement fails must be refined as follows: Agree-Copy always adds a value to a probe feature. The matching condition is then defined in a way that a feature must not bear differing case values. Thus, only if a mismatch between two values cannot be repaired by a syncretism rule, the derivation fails. Put differently, the syncretism rules can feed Agree-Copy.

The derivation in (71) shows the interaction of Agree-Copy and syncretism rule (70-a) for Polish.

(71) Derivation of (68-b)

Step I:	$\alpha[c:_][anim: +]$	\longrightarrow	Appl[c:gen]:	$\alpha[c:gen][anim: +]$
<hr/>				
Step II:	$\emptyset[*c: _ *][anim: +]$	\longrightarrow	$\alpha[c:gen][anim: +]:$	$\emptyset[*c:gen*][anim: +]$
	or $\emptyset[*c: _ *][anim: +]$	\longrightarrow	$v[c:acc]:$	$\emptyset[*c:acc*][anim: +]$
	$[c:acc]$	\rightarrow	$[c:gen]/[anim: +]:$	$\emptyset[*c:gen*][anim: +]$
Step III:	$\emptyset[*c:gen*][anim: +]$	\longrightarrow	$v[c:acc]:$	$\emptyset[c:gen,acc][anim: +]$
	$[c:acc]$	\rightarrow	$[c:gen]/[anim: +]:$	$\emptyset[c:gen,gen] = \emptyset[c:gen]$
	or $\emptyset[*c:gen*][anim: +]$	\longrightarrow	$\alpha[c:gen][anim: +]:$	$\emptyset[c:gen,gen][anim: +]$
				$= \emptyset[c:gen][anim: +]$

Being a single probe, the overt wh-phrase receives genitive case first. Afterwards, the \emptyset_D receives a value from either the wh-phrase or the matrix v . If it Agrees with the wh-phrase, it receives genitive. When receiving the accusative case feature from

matrix v in Step III, the context for the syncretism rule is given¹⁵ and the accusative value is changed into genitive, which prevents a conflict on the case feature.

The other possibility is that the D head first receives accusative case from matrix v. The context for the syncretism rule is given at this point, so the accusative feature is immediately changed into a genitive feature. Then, Agree between D and the wh-phrase in Step III does not lead to a violation of the matching condition.

As for German, the following syncretism rules are needed.

(72) *Syncretism rules in German*

- a. $[c:gen] \rightarrow [c:dat]/[g:fem][n:sg]$
- b. $[c:acc] \rightarrow [c:nom]/[g:neutr]$

The result of including the rule (72-a) in the derivations of German parasitic gap constructions is shown in (73).

(73) *Derivation of (69-a)*

Step I:	$\emptyset[c:_][g:fem][n:sg]$	\rightarrow	Appl[c:dat]:	$\emptyset[c:dat][g:fem][n:sg]$
<hr/>				
Step II:	$\alpha[*c:_][g:fem][n:sg]$	\rightarrow	$\emptyset[c:dat][g:fem][n:sg]:$	$\alpha[*c:dat*][g:fem][n:sg]$
	or $\alpha[*c:_][g:fem][n:sg]$	\rightarrow	Appl[c:gen]:	$\alpha[*c:gen*][g:fem][n:sg]$
	$[c:gen]$	\rightarrow	$[c:dat]/[g:fem][n:sg]:$	$\alpha[*c:dat*][g:fem][n:sg]$
Step III:	$\alpha[*c:dat*][g:fem][n:sg]$	\rightarrow	Appl[c:gen]:	$\alpha[c:gen,dat][g:fem][n:sg]$
	$[c:gen]$	\rightarrow	$[c:dat]/[g:fem][n:sg]:$	$\alpha[c:dat,dat][g:fem][n:sg]$
				$= \alpha[c:dat][g:fem][n:sg]$
	or $\alpha[*c:dat*][g:fem][n:sg]$	\rightarrow	$\emptyset[c:dat][g:fem][n:sg]:$	$\alpha[c:dat,dat][g:fem][n:sg]$
				$= \alpha[c:dat][g:fem][n:sg]$

¹⁵By assumption, \emptyset and α are valued alike for features such as animacy and gender. This assumption is further justified in part II.

5. *Complicating the Pattern*

This derivation works just as the derivation in (71). The rule (72-a) applies as soon as the overt antecedent receives genitive case, changing it to dative case. This prevents a violation of the matching condition.

Finally, it should be mentioned that the syncretism rules also apply in the derivations that involve upward Agree, that is, in the derivations of German free relatives and Polish parasitic gaps. Here, these rules have the same effects as in the derivations in (71) and (73). But since case mismatches are tolerated in these derivations anyway due to the different order of Agree operations, the syncretism rules do not matter for the case matching effects. They only influence the phonological exponent chosen for the heads.

In sum, the syncretism effects of case matching can be derived by assuming that the morphological rules that are responsible for syncretism can interact with Agree. This is possible because there is a morphological component of Agree, namely the operation Agree-Copy. A purely syntactic approach to Agree would have to invoke an additional mechanism to capture these effects.

5.2. Case Hierarchy Effects

This section contains a discussion of case hierarchy effects that can be observed with case matching in free relatives, but not in parasitic gaps. Assuming that both types of case matching involve the same mechanisms of case assignment and case agreement, it is puzzling that a difference can be observed at all.

At first, I will describe the case hierarchy effect in German and will then go on to adapt the analysis of case matching effects in order to capture this effect. I will

conclude with the main claim of this chapter: The absence of hierarchy effects with parasitic gaps cross-linguistically is a systematic gap.

5.2.1. Case Hierarchy Effects in German Free Relatives

The hierarchy effects in German are exemplified in the data below. In the examples in (74), the matrix clause assigns nominative case to the free relative, that is, the free relative is the subject of the matrix clause.¹⁶

- (74)
- a. Uns besucht, wer Maria mag.
us visits_{nom} who_{nom} Maria likes_{nom}
 - b. Uns besucht, wen Maria mag.
us visits_{nom} who_{acc} Maria likes_{acc}
 - c. Uns besucht wem Maria vertraut.
us visits_{nom} who_{dat} Maria trusts_{dat}
 - d. Uns wird nützen wessen Maria sich gestern entledigt hat.
us will be.useful_{nom} who_{gen} Maria self yesterday got.rid.of_{gen} has

As can be seen in (74-a) through (74-d), any case combination is tolerated here, that is, the *wh*-phrase can receive nominative, accusative, dative, or genitive case in the embedded clause. The next examples in (75) illustrate the case combinations if the free relative is the accusative object of the matrix clause.

- (75)
- a. *Ich lade ein, wer Maria mag.
I invite_{acc} who_{nom} Maria likes_{nom}
 - b. Ich lade ein, wen Maria mag.
I invite_{acc} who_{acc} Maria likes_{acc}

¹⁶The data in (74)–(77) are either taken directly from or based on the observations of Vogel (2001).

5. Complicating the Pattern

- c. Ich lade ein, wem Maria vertraut.
I invite_{acc} who_{dat} Maria trusts_{dat}
- d. Ich kann gebrauchen wessen Maria sich gestern entledigt hat.
I can use_{acc} who_{gen} Maria self yesterday got.rid.of_{gen} has

In this case, the only possible cases assigned in the embedded clause are accusative, dative, and genitive case. A case mismatch with nominative is not possible. Turning to dative case, (76) shows what happens if the free relative is assigned dative case from the matrix verb.

- (76)
- a. *Ich vertraue, wer Maria mag.
I trust_{dat} who_{nom} Maria likes_{nom}
 - b. *Ich vertraue, wen Maria mag.
I trust_{dat} who_{acc} Maria likes_{acc}
 - c. Ich vertraue, wem Maria vertraut.
I trust_{dat} who_{dat} Maria trusts_{dat}
 - d. *Der Mann ähnelt, wessen Maria gestern in ihrer Rede
the man resembles_{dat} who_{gen} Maria yesterday in her speech
gedacht hat.
commemorated_{gen} has

If this is the case, nominative, accusative and genitive case are not possible in the embedded clause. Only dative case is allowed (76-c). Finally, (77) illustrates the effects with genitive case.

- (77)
- a. *Maria gedenkt, wer vor zehn Jahren bei dem Anschlag
Maria commemorates_{gen} who_{nom} ago ten years at the attack
ums Leben kam.
died_{nom}
 - b. *Maria gedenkt, wen die Terroristen vor zehn Jahren bei
Maria commemorates_{gen} who_{acc} the terrorists ago ten years at
dem Anschlag töteten.
the attack killed_{acc} have

- c. *Maria gedenkt, wem die Terroristen vor zehn Jahren bei
 Maria commemorates_{gen} who_{dat} the terrorists ago ten years at
 dem Anschlag Schmerzen zugefügt haben.
 the attack pain caused_{dat} have
- d. Maria bedarf, wessen ich mich gestern entledigt habe.
 Maria needs_{gen} who_{gen} I self yesterday got.rid.of_{gen} have

Similar to the dative case examples in (76), a genitive case assigned in the matrix clause can only be matched against a genitive case assigned in the embedded clause.¹⁷

The data in (74)–(77) yield the pattern in (78).

(78)

FR M	Nom	Acc	Dat	Gen
Nom	✓	✓(acc)	✓(dat)	✓(gen)
Acc	*	✓	✓(dat)	✓(gen)
Dat	*	*	✓	*
Gen	*	*	*	✓

As in section 3.3, the rows show the cases assigned in the matrix clause, while the

¹⁷Note that the syncretisms observed between genitive and dative case in German do not apply here. The syncretism is only given in the context of a feminine singular phrase. However, singular phrases are not possible wh-phrases in German free relatives, not even if there is no case mismatch.

- (i) *Maria hilft, welcher Frau Tom vertraut.
 Maria helps_{dat} which_{dat} woman Tom trusts_{dat}

5. Complicating the Pattern

columns display the cases assigned in the embedded clause. Summarizing both dative case and genitive case under oblique case, as in (79), an even clearer picture shows of what the case hierarchy effect in German actually is.

(79)

FR \ M	Nom	Acc	Obl
Nom	✓	✓(acc)	✓(obl)
Acc	*	✓	✓(obl)
Obl	*	*	✓

The pattern shows a hierarchy between the cases insofar as the nominative case in the matrix clause allows more combinations than the accusative clause, which in turn allows more combinations than the oblique cases. This hierarchy can be summarized as in (80).

(80) *Case Hierarchy*

NOM >> ACC >> DAT / GEN

The empirical constraint for case matching effects in German free relatives is the following (cf. Pittner (1991, 1995); Vogel (2001); Grosu (2003)): if the case assigned by the matrix clause is higher on the hierarchy than the case assigned within the embedded clause, the *wh*-phrase in the free relative may bear the case assigned to it inside the embedded clause. Importantly, if a case mismatch is allowed, the case marker on the *wh*-phrase *must* be the case assigned within the embedded clause.

5.2.2. Analysis

So far, the system developed in chapter 4 does not capture the case hierarchy effects. In order to achieve this, the conditions of Agree must be revised and the representation of case features needs to be refined.

5.2.2.1. The Representation of Case

In the derivations below, the case hierarchy is not a primitive concept or constraint but rather comes out as the result of decomposing the cases such that the cases are ordered in subset-superset relations (cf. Caha (2009), Béjar and Řezáč (2009)). I will assume the following decomposition of cases:

- (81)
- a. *Nominative*: [c:{nom}]
 - b. *Accusative*: [c:{nom,acc}]
 - c. *Dative*: [c:{nom, acc, dat}]
 - d. *Genitive*: [c:{nom, acc, gen}]

As shown in (81), the cases are represented as features that can bear sets of case feature values. Consequently, some cases are subsets of other cases. Ordering the cases according to their subset relations, as in (82), indicates that the decomposition of cases represents the case hierarchy.

(82) Case Hierarchy

$$\{\text{nom}\} \subset \{\text{nom,acc}\} \subset \{\text{nom, acc, dat}\}/\{\text{nom, acc, gen}\}$$

5. Complicating the Pattern

In the next section, I will develop a revised definition of the Agree operation that is able to deal with this representation of case and that derives the case hierarchy effect in German free relatives.

5.2.2.2. Agree (revised)

The first point about Agree that needs to be revised affects the asymmetry of Agree. In section 4.2, I have established that Agree operations are asymmetric: there is one probe and one goal. In order to capture the case hierarchy effects, I would like to propose a further possibility: In addition to asymmetric Agree, there is also symmetric Agree, where both categories involved in Agree, probe for each other. As for Agree-Copy, the two probes in a symmetric Agree relation are valued independently from each other. The two different types of Agree are shown in (83).

(83) a. *Asymmetric Agree*

$$X[F: _] \longrightarrow Y[F:v]$$

b. *Symmetric Agree*

$$X[F: _] \longleftrightarrow Y[*F:v*]$$

I further assume that variation in symmetry affects Agree between an argument and a case assigning functional head as well as Agree between the overt and the covert category in parasitic gap and free relative constructions. For German, Agree between an argument and a functional head is asymmetric and Agree in parasitic gap and free relative constructions is symmetric. Note that this deviates from the analysis in section 4.3, where I made the assumption that the latter kind of Agree

relation in German was asymmetric in order to derive a simplified version of the full pattern of case matching effects in German. In the next section, by looking at Modern Greek, I will show that asymmetric Agree in these constructions does exist.

A second point about Agree that needs refinement concerns the order of Agree-Copy operations. So far, I have assumed that the order of valuation of features that participate in more than one Agree operation is not determined and that it can proceed in any order. In the derivations below I will, however, assume that the order in these derivations is determined by the order in which Agree-Links have been established. Technically, this can be achieved by ordering the pairs of probes and goals in an ordered list. Leaving conceptual details aside, it suffices to assume that a feature can keep track of the order in which it has found its goals.

Finally, the matching condition of Agree is refined. In section 4.3, I have proposed that the matching condition of Agree says that Agree is only successful if the feature value of probe and goal do not conflict. The absence of a feature conflict is trivially given if one of the two features is still unvalued. If the probe and the goal are both already valued, a conflict arises under the following conditions:

(84) *Matching Conditions for Agree in case features*

- a. $G \subseteq P$: Agree between a DP and a case assigning functional head F fails if both probe and goal bear case values and if the case values on the goal are a superset of the case values on the probe.
- b. $G = P$: Agree between α and \emptyset in a free relative or parasitic gap construction fails if both bear cases and if the cases are not identical.

5. *Complicating the Pattern*

Intuitively, (84-a) boils down to the condition that every case value on the goal has to have a matching case value on the probe. This condition also prevents cases from being changed throughout the derivation, a condition which must be established independently in a system that allows agreement with more than one goal. The difference between (84-a) and (84-b) is that the conditions for case agreement in sharing constructions are stricter than in normal case assignment relations. This emphasizes the close relation between the overt and the covert category in these constructions.

In a nutshell, the idea of the analysis is that the case hierarchy effects come about if the final Agree-Copy relation is subject to a subset matching condition, or put differently, if the final Agree-Copy relation is between an argument and a functional head. Since in parasitic gap constructions the final Agree-Copy relation is subject to a strict identity matching condition, case matching effects in parasitic gaps cannot be restricted by the case hierarchy.

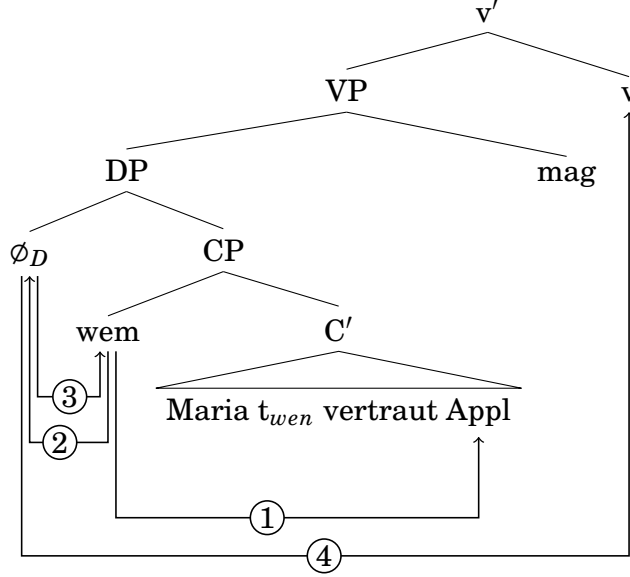
The following section shows in detail why German free relatives show case hierarchy effects and why German parasitic gap constructions do not show the same effects.

5.2.3. **Derivations**

The first derivation in (85) to (88) shows why there are case hierarchy effects in German free relatives. (85) repeats the syntactic structure that I assume for free relatives in German. In the section 5.2.2.2, I have proposed that Agree between the covert D head and the wh-phrase in German is symmetric, which means that

Agree happens in two steps. Other than that, the structure in (85) is the same as the structure in section 4.5.3.

(85) *Syntactic Agree-Link*



The derivation in (86) shows the order of the post-syntactic Agree-Copy operations that result from the syntactic structure in (85).

(86) *FR: $\phi[acc]$, $\alpha[dat]$*

Step I	$\alpha[*c:_*]$	\rightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\alpha[*c:\{dat,acc,nom\}*]$
Step II	$\alpha[*c:\{dat,acc,nom\}*]$	\rightarrow	$\phi[*c:_*]:$	$\alpha[c:\{dat,acc,nom\}]$
Step III	$\phi[*c:_*]$	\rightarrow	$\alpha[c:\{dat,acc,nom\}]:$	$\phi[*c:\{dat,acc,nom\}*]$
Step IV	$\phi[*c:\{dat,acc,nom\}*]$	\rightarrow	$v[c:\{acc,nom\}]:$	$\phi[c:\{dat,acc,nom\}]$ ($v \subseteq \phi$)

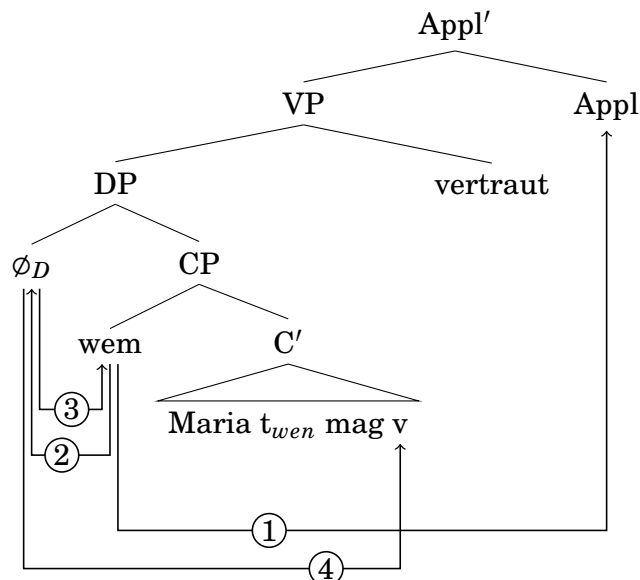
In the first step, the wh-phrase receives dative case from the embedded applicative head. According to the case decomposition in section 5.2.2.1, dative case is represented as the feature set consisting of the case values [dat], [acc], and [nom]. In

5. Complicating the Pattern

the second step, the wh-phrase Agrees with the D head. Since the D head has not received a case value at this step, the matching condition of Agree is trivially fulfilled. Afterwards, because Agree between the wh-phrase and the D head is symmetric, the D head copies the case values from the wh-phrase. Thus, the D head bears dative case as well. In the final step, the D head Agrees with the matrix v head. Since both the probe and the goal bear values now, the matching condition of Agree is only fulfilled, if the case value of the goal is a subset of the case value of the probe (see (84-a)). Since this is the case, Agree is successful despite a case mismatch between the matrix verb and the embedded verb.

In the derivation in (87) and (88), the verbs are switched. Thus, the matrix verb assigns dative case and the embedded verb assigns accusative case. Due to the subset condition that that applies here such a case mismatch should be ruled out.

(87) Syntactic Agree-Link



(88) *FR*: $\emptyset[\text{dat}]$, $\alpha[\text{acc}]$

Step I	$\alpha[*c:_*]$	\longrightarrow	$v[c:\{\text{acc,nom}\}]:$	$\alpha[*c:\{\text{acc,nom}\}^*]$
Step II	$\alpha[*c:\{\text{acc,nom}\}^*]$	\longrightarrow	$\emptyset[*c:_*]:$	$\alpha[c:\{\text{acc,nom}\}]$
Step III	$\emptyset[*c:_*]$	\longrightarrow	$\alpha[c:\{\text{acc,nom}\}]:$	$\emptyset[*c:\{\text{acc,nom}\}^*]$
Step IV	$\emptyset[*c:\{\text{acc,nom}\}^*]$	\longrightarrow	$\text{Appl}[c:\{\text{dat,acc,nom}\}]:$	$\nmid (\text{Appl} \not\subseteq \emptyset)$

The Agree operations proceeds in the same way as in (85) and (86). The difference lies in the case values: The *wh*-phrase and the D head bear accusative case from the embedded *v* head. In the final Agree relation between the dative assigning Appl and the D head, the matching condition (84-a) is violated and the derivation crashes because dative case is a superset of accusative case.

These examples show that, in general, case mismatches in free relatives are not tolerated if the case assigned in the matrix clause is a superset of the the case assigned in the embedded clause. If it is a subset, a case mismatch is fine. Due to subset/superset relations between the cases, the picture in (89) is emerges.

(89)

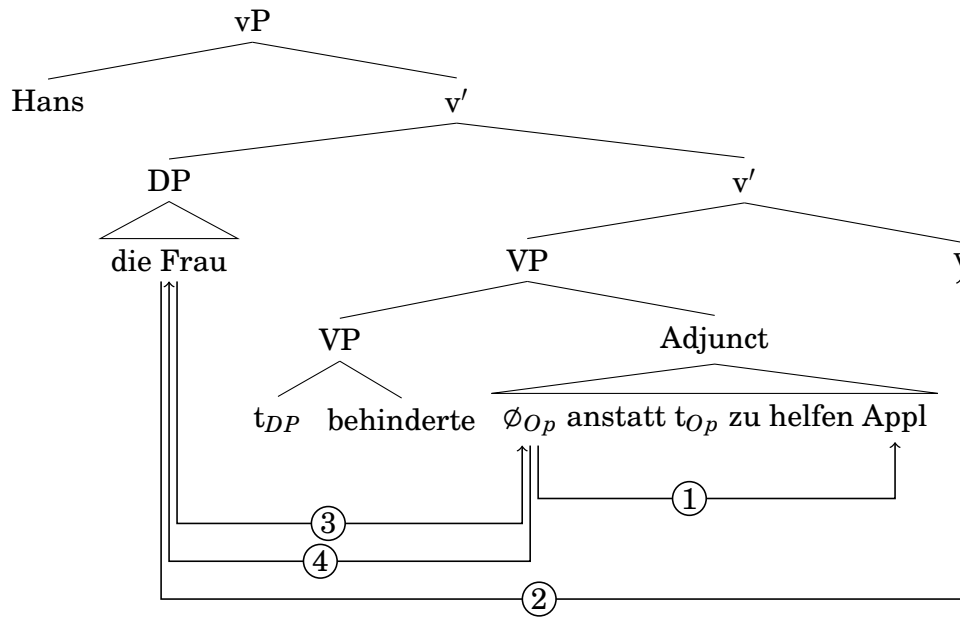
M \ FR			
	{nom}	{acc,nom}	{dat/gen,acc,nom}
{nom}	✓(\subseteq)	✓($((\subseteq))$)	✓($((\subseteq))$)
{acc,nom}	*(\supset)	✓(\subseteq)	✓($((\subseteq))$)
{dat/gen,acc,nom}	*(\supset)	*(\supset)	✓(\subseteq)

The subset and superset relations in (89) directly correspond to the case hierarchy effects in (79).

5. Complicating the Pattern

The next two derivations show why there are no case hierarchy effects in parasitic gap constructions in German. (90) shows the structure of a parasitic gap construction. As in the derivations of free relatives, the Agree relation between the covert element – here the empty operator – and the overt element – here matrix object *die Frau* – is symmetric.

(90) *Syntactic Agree-Link*



(91) summarizes the Agree-Copy relations that result from the syntactic structure in (90).

(91) *PG: \emptyset [dat], α [acc]*

Step I $\emptyset[*c:_*]$ \longrightarrow Appl[c:{dat,acc,nom}]: $\emptyset[*c:\{dat,acc,nom\}*]$

Step II $\emptyset[*c:\{dat,acc,nom\}*]$ \longrightarrow $\alpha[*c:_*]$: $\emptyset[c:\{dat,acc,nom\}]$

Step III $\alpha[*c:_*]$ \longrightarrow v[c:{acc,nom}]: $\alpha[*c:\{acc,nom\}*]$

Step IV $\alpha[*c:\{acc,nom\}*]$ \longrightarrow $\emptyset[c:\{dat,acc,nom\}]$: \nexists ($\alpha \neq \emptyset$)

Being low in the structure, the empty operator receives its case value first. Now, the order in which it receives values from the embedded applicative head and the scrambled matrix object is determined by the order in which the Agree-Links were established in the syntax. Thus, \emptyset_{Op} first gets dative case from Appl and then Agrees with the overt phrase *die Frau*, which does not bear a case value at this point. Afterwards, the *die Frau* receives its case value. The order in which Agree-Copy proceeds for this higher case probe here is different from the order for the D head in free relative constructions. This difference is the direct result of the structural differences between the two constructions: Before the operator can Agree with the scrambled object *die Frau*, the object has already established an Agree-Link in its base position. Consequently, Agree-Copy between the scrambled object and the matrix v (Step III in (91)) applies before Agree between the scrambled object and the operator (Step IV in (91)). But in Step IV, both the probe and the goal of the Agree relation already bear a case value. Since this last Agree relation requires the stricter matching condition (84-a), the case mismatch between accusative and dative case is not tolerated.

Obviously, this case mismatch can not be resolved either if dative and accusative case are reversed as in (92), where accusative is assigned in the embedded clause and dative case is assigned in the matrix clause.

5. Complicating the Pattern

(92) *PG*: $\emptyset[acc]$, $\alpha[dat]$

Step I	$\emptyset[*c:_*]$	\longrightarrow	$v[c:\{acc,nom\}]:$	$\emptyset[*c:\{acc,nom\}*]$
Step II	$\emptyset[*c:\{acc,nom\}*]$	\longrightarrow	$\alpha[*c:_*]:$	$\emptyset[c:\{acc,nom\}]$
Step III	$\alpha[*c:_*]$	\longrightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\alpha[*c:\{dat,acc,nom\}*]$
Step IV	$\alpha[*c:\{dat,acc,nom\}*]$	\longrightarrow	$\emptyset[c:\{acc,nom\}]:$	$\nexists (\alpha \neq \emptyset)$

Again, this case mismatch violates the matching condition (84-b) and the derivation crashes because of it.

To sum up the advantage of the system proposed in this section, is that the distribution of the case hierarchy effects results from the structural differences between free relative and parasitic gap constructions. The rules of case assignment and case agreement do not have to be construction-specific.

5.3. More Patterns

The analysis of case matching effects developed so far is able to derive the fact that case matching effects can differ between constructions, that case matching effects differ between languages and that the case hierarchy plays a role in case matching effects in free relatives but not in parasitic gap constructions. In the theory, this empirical variation is based on three parameters, which are repeated in (93).

- (93)
- a. Symmetry case Agree F – argument?: symmetric vs. asymmetric
 - b. Symmetry case Agree $\alpha - \emptyset$?: symmetric vs. asymmetric
 - c. (if (93-b) asymmetric) Probe on α or \emptyset ? α vs. \emptyset

These parameters follow the Borer-Chomsky-Conjecture (Borer (1984); Chomsky (1995); Baker (2008a)) that cross-linguistic variation concerns lexical items only: In all the parameters for Agree relations in (93), the variation is reduced to the question if a category is a probe or not. consequently, the parameters can be reduced to the following in (94).

- (94) a. $F[*case*:val]$ vs. $F[case:val]$
 b. $\alpha[*case*:_]$ vs. $\alpha[case:_]$
 c. $\emptyset[*case*:_]$ vs. $\emptyset[case:_]$

These three binary parameters lead to eight possible combinations, two of which are excluded if the agreement relation between α and \emptyset is seen as an essential property of free relative and parasitic gap constructions. If such a combination were possible, there should exist languages in which both constructions allow case mismatches. To my knowledge, such a case has not been reported in the literature.

This leaves us with six patterns. For five of these six patterns, examples can be found in the literature. The table in (95) summarizes the patterns and provides examples. In the rest of this section, I will discuss each of these patterns and show how the analysis deals with them.

5. Complicating the Pattern

(95) Typology of Case Matching

F	α	\emptyset	mismatch	mismatch	Pattern
[*c*]?	[*c*]?	[*c*]?	FR?	PG?	
✓	✓	✓	*	*	German 2, Polish 2 (sec. 5.3.1)
		*	✓	*	Greek (sec. 5.3.3)
	*	✓	*	✓	Polish 1 (sec. 5.3.2)
		*	—	—	_18
*	✓	✓	hierarchy	*	German 1 (sec. 5.2)
		*	✓	*	Greek (sec. 5.3.3)
	*	✓	hierarchy	✓	??? (sec. 5.3.4)
		*	—	—	_18

5.3.1. German 2, Polish 2

In section 5.2, the German pattern described in section 3.1 is analyzed. This pattern, however, is only available to one group of German speakers. There is a second group of German speakers for whom (non-syncretic) case mismatches in free relatives are not acceptable (see Riemsdijk (2006)). For these speakers, case mismatches are excluded in both free relative and parasitic gap constructions. In the table in (95), this variety of German is called German 2, in contrast to German 1 in sections 3.1 and 5.2.1. An example of German 2 is given in (96).

- (96) Hans mag_{acc}, *wen_{acc}/(*)wem_{dat} Maria vertraut_{dat}.
 Hans likes who Maria trusts
 ‘Hans likes whomever Maria trusts.’

¹⁸These combinations are excluded if α or \emptyset have to Agree.

For speakers of German 2, the *wh*-phrase in (96) has to match the dative case of the embedded verb *vertrauen* as well as the accusative case of the matrix verb *mögen*. This conflict cannot be resolved.

A similar split between speaker judgments can be observed in Polish. There is a group of Polish speaker who do not allow for non-syncretic case mismatches in parasitic gap constructions (Bondaruk (1996)). The stricter variety is called Polish 2 here, in order to distinguish it from the variety described in section 3.2. An example that illustrates the judgments in Polish 2 is given in (97).

- (97) To jest dziewczyna, *która_{acc} / *której_{dat} Jan lubił_{acc} zanim zaczął
 this is girl which Jan liked before started
 pomagać_{dat}.
 help.

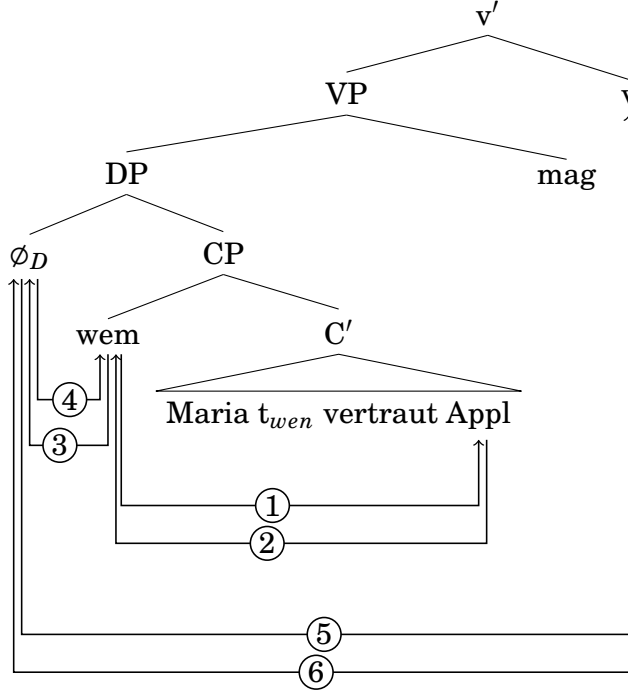
Similar to German 2, the case conflict between accusative and dative case cannot be resolved for these speakers.

Both German 2 and Polish 2 can be derived under the assumption that, for these speakers, Agree is always symmetric. Put differently, the strict varieties in both languages have the probes of German 1 and the probes of Polish 1.

The derivation in (98) and (99) shows why non-syncretic case mismatches are not allowed in these varieties. Note that the basic syntactic structures of free relatives in German 2/Polish 2 is the same as in German 1 (see (85)). The only difference concerns the number of Agree operations.

5. Complicating the Pattern

(98) Syntactic Agree-Link



These syntactic Agree-Link relations lead to the following order of post-syntactic Agree-Copy operations in (99).

(99) *FR*: $\phi[acc]$, $\alpha[dat]$

Step I	$Appl[*c:\{dat,acc,nom\}*]$	\longrightarrow	$\alpha[*c:_*]:$	$Appl[c:\{dat,acc,nom\}]$
Step II	$\alpha[*c:_*]$	\longrightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\alpha[*c:\{dat,acc,nom\}*]$
Step III	$\alpha[*c:\{dat,acc,nom\}*]$	\longrightarrow	$\phi[*c:_*]:$	$\alpha[c:\{dat,acc,nom\}]$
Step IV	$\phi[*c:_*]$	\longrightarrow	$\alpha[c:\{dat,acc,nom\}]:$	$\phi[*c:\{dat,acc,nom\}*]$
Step V	$\phi[*c:\{dat,acc,nom\}*]$	\longrightarrow	$v[*c:\{acc,nom\}*]:$	$\phi[c:\{dat,acc,nom\}]$ ($v \subseteq \phi$)
Step VI	$v[*c:\{acc,nom\}*]$	\longrightarrow	$\phi[c:\{dat,acc,nom\}]:$	\nexists ($\phi \not\subseteq v$)

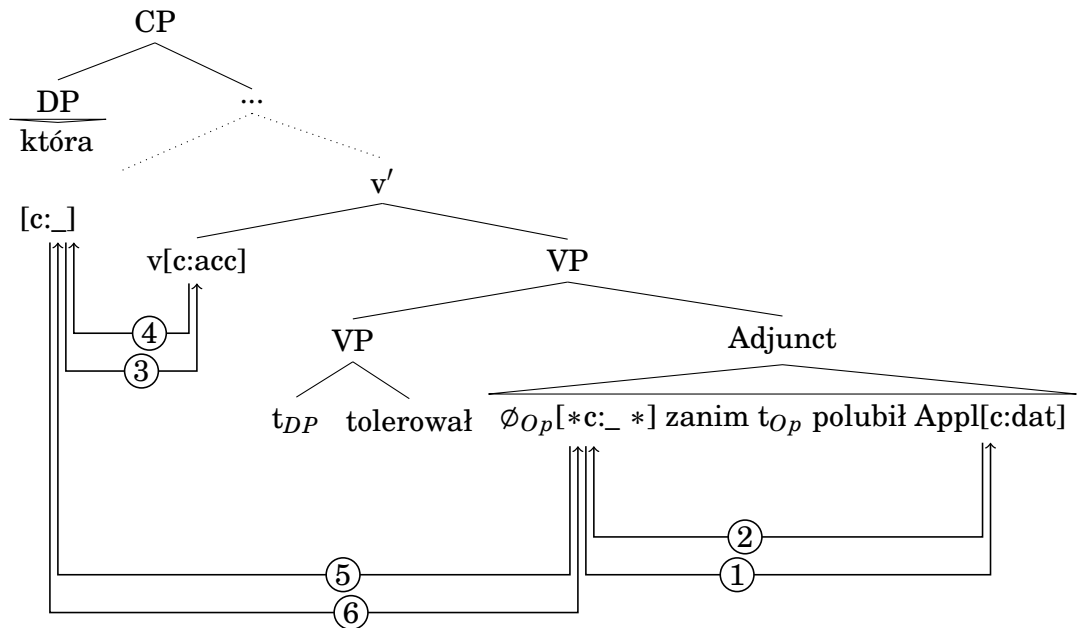
Being the lowest of the case probes, Appl checks its case feature first against the unvalued case feature of the wh-phrase (Step I). Next, the case probe of the wh-phrase is satisfied. Due to the order, in which the Agree-Links have been established

for the *wh*-phrase, it first receives dative case from Appl (Step II) and then probes again for the unvalued case feature of the D head (Step III). The third probe to be valued is the D head. Starting with Agree-Copy that targets the *wh*-phrase, the D head receives dative case first (Step IV) before checking this case against the accusative case of the matrix *v* (Step V). The outcome of Agree at this point is successful, because the respective matching condition (84-a) is fulfilled. The reason why the derivation fails is to be found in Step VI: This Agree operation between the matrix *v* and the D head is subject to the same matching condition (84-a). However, in this case, the subset condition is not given and Agree fails.

Thus, the additional Agree relations that German 2 has in contrast to German 1 leads to a stricter case matching condition for free relatives.

For the same reason, case mismatches are not tolerated in parasitic gap constructions. This is illustrated for Polish 2 in (100) and (101).

(100) *Syntactic Agree-Link*



5. Complicating the Pattern

(101) *PG: $\emptyset[\text{dat}]$, $\alpha[\text{acc}]$*

Step I	$\text{Appl}[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longrightarrow	$\emptyset[*c: _ *]:$	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\longrightarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[*c: _ *]:$	$\emptyset[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step IV	$v[*c:\{\text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[*c: _ *]:$	$v[c:\{\text{acc}, \text{nom}\}]$
Step V	$\alpha[*c: _ *]$	\longrightarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\alpha[*c:\{\text{acc}, \text{nom}\} *]$
Step VI	$\alpha[*c:\{\text{acc}, \text{nom}\} *]$	\longrightarrow	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]:$	$\nexists (\alpha \neq \emptyset)$

The reason why this case mismatch leads to a crash of the derivation is the same as in (90): After the case feature of both the relative pronoun *ktora* and the operator \emptyset_{Op} has been valued, the two enter into an Agree-Copy-relation. Due to a mismatch in the values, the matching condition (84-b) is violated and Agree fails.

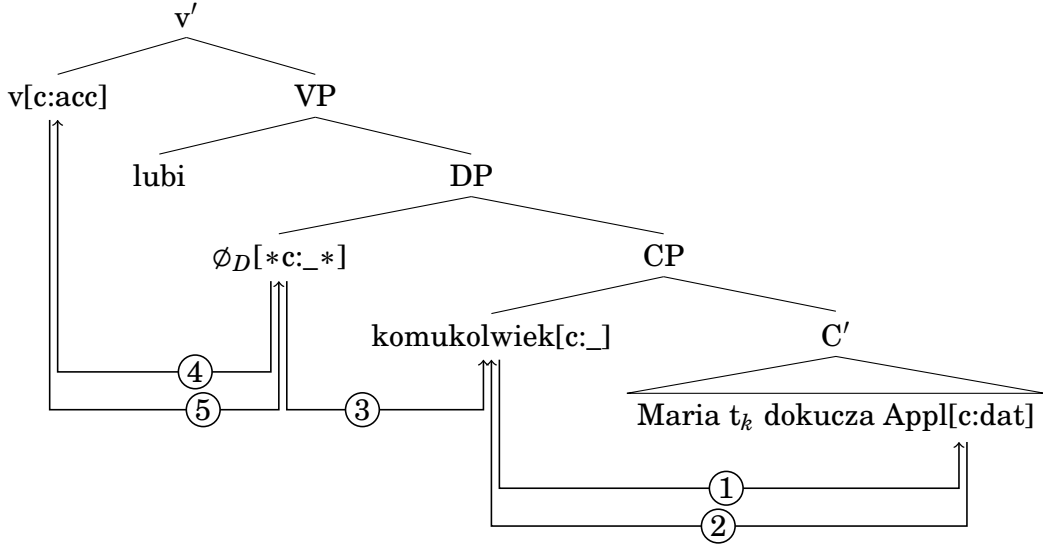
In sum, case matching in German 2 and Polish 2 is strict all case features are probes in these varieties. This leads to configurations where case values always have to be identical between probe and goal and case mismatches are hence not tolerated.

5.3.2. Polish 1

The next pattern that needs to be looked at under the revised system of case agreement is exemplified by Polish 1. In Polish 1, case mismatches are ungrammatical for free relatives but not for parasitic gaps (see section 3.2 for details).

As summarized in the table in 95, case agreement in Polish is symmetric for relations between an argument and a functional head but asymmetric for case agreement in free relatives and parasitic gap constructions, with the covert element being the probe in these relations.

These assumptions derive that (i) case mismatches are not allowed in free relatives and that (ii) case mismatches may occur with parasitic gaps, but do not follow the case hierarchy here. The following four derivations show why this is the case. We start with free relatives.

(102) *Syntactic Agree-Link*

In the structure in (102), the embedded applicative head would assign dative case to the embedded wh-phrase and the matrix *v* would assign accusative case to the *D* head. The example that this structure is based on is, however, ungrammatical (see section 3.2, example (24)). (103) shows why.

(103) *FR: Ø[acc], α[dat]*

Step I	Appl[*c:{dat,acc,nom}*]	→	α[c:_]:	Appl[c:{dat,acc,nom}]
Step II	α[c:_]	→	Appl[c:{dat,acc,nom}]:	α[c:{dat,acc,nom}]
Step III	Ø[*c:_*]	→	α[c:{dat,acc,nom}]:	Ø[*c:{dat,acc,nom}*]
Step IV	Ø[*c:{dat,acc,nom}*]	→	v[*c:{acc,nom}*]:	Ø[c:{dat,acc,nom}] (Ø ≠ v)
Step V	v[*c:{acc,nom}*]	→	Ø[c:{dat,acc,nom}]:	⚡ (Ø ≠ v)

5. Complicating the Pattern

The reason, why Agree in Step V fails is the same as in (99): Due to symmetric Agree between functional heads and arguments, the cases of the D head and of the case assigning functional head in the matrix clause have to be identical. Otherwise, the subset condition will not be fulfilled either at Step IV or at Step V.

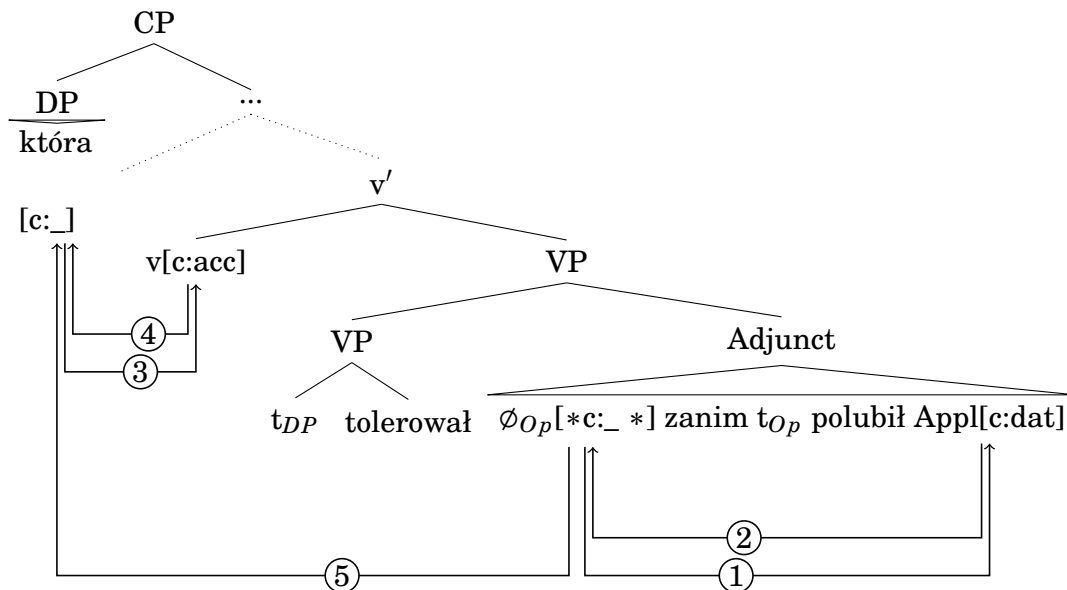
Therefore, even if the *v* head and the applicative head switch their positions in the structure in (103), the outcome has to be ungrammatical. This is shown in the derivation in (104).

(104) *FR: $\phi[acc]$, $\alpha[dat]$*

Step I	$v[*c:\{acc,nom\}^*]$	\rightarrow	$\alpha[c:_]:$	$v[c:\{acc,nom\}]$
Step II	$\alpha[c:_]$	\rightarrow	$v[c:\{acc,nom\}]:$	$\alpha[c:\{acc,nom\}]$
Step III	$\phi[*c:_^*]$	\rightarrow	$\alpha[c:\{acc,nom\}]:$	$\phi[*c:\{acc,nom\}^*]$
Step IV	$\phi[*c:\{acc,nom\}^*]$	\rightarrow	$Appl[*c:\{dat,acc,nom\}^*]:$	\nmid ($Appl \not\subseteq \phi$)

Turning to parasitic gaps, in (105)–(107) show why case mismatches are fine and why the case hierarchy does not play a role for Polish parasitic gaps either.

(105) *Syntactic Agree-Link*



(106) *PG: $\emptyset[\text{dat}]$, $\alpha[\text{acc}]$*

Step I	$\text{Appl}[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longrightarrow	$\emptyset[*c: _ *]:$	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\longrightarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[c: _]:$	$\emptyset[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step IV	$v[*c:\{\text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[c: _]:$	$v[c:\{\text{acc}, \text{nom}\}]$
Step V	$\alpha[c: _]$	\longrightarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{acc}, \text{nom}\}]$

The derivation in (106) is successful because the Agree-Copy relation that made the mismatch impossible in (101) is missing here. This basically means that, at the point, the step where the matching condition ((84-b)) would rule out the case mismatch does not apply. This is independent of which cases are concretely chosen. As can be seen in the derivation in (107), also a different constellation of cases does not cause a conflict.

(107) *PG: $\emptyset[\text{acc}]$, $\alpha[\text{dat}]$*

Step I	$v[*c:\{\text{acc}, \text{nom}\} *]$	\longrightarrow	$\emptyset[*c: _ *]:$	$v[c:\{\text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\longrightarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[c: _]:$	$\emptyset[c:\{\text{acc}, \text{nom}\}]$
Step IV	$\text{Appl}[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longrightarrow	$\alpha[c: _]:$	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step V	$\alpha[c: _]$	\longrightarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$

To sum up, free relatives show strict case matching in Polish 1 because of two reasons: First, case agreement between arguments and functional heads is symmetric and, thus, requires identity of case values in order to be successful. Second, this identity is required at a point when all case features are already valued. Thus, the matching

5. Complicating the Pattern

condition cannot be counter-bleed. Parasitic gaps, on the other hand, do not require strict matching because the agreement between arguments and functional heads applies before all the respective case features are valued. So, in parasitic gaps in Polish 1, we have a counter-bleeding interaction between the different Agree-Copy operations. This counter-bleeding also leads to the absence of case hierarchy effects.

5.3.3. (Modern) Greek

A different pattern of case matching effects can be observed in free relatives in Modern Greek. All the following data are taken from Daskalaki (2011). In the examples in (108), the external predicate assigns nominative to the free relative.

- (108) a. θa se voiθísi ópjós se ayapá
will you help_{nom} who_{nom} you love_{nom}
'Whoever loves you will help you.'
- b. Írθan ópji / *ópjus káleses.
came_{nom} who_{nom} / *who_{acc} (you) invited_{acc}
'Whoever you invited came.'
- c. Me efχarístisan ópji *(tus) / *ópjon ίχα ὀόσι leftá.
me thanked_{nom} who_{nom} *(CL_{gen}) / *who_{gen} (I) had given_{gen} money
'Whoever I had given money to thanked me.'

As can be seen in (108), all combinations are possible, similarly to German 1, but, unlike German 1, the *wh*-phrase in Greek must realize the case assigned by the matrix predicate; the realization of the embedded case leads to ungrammaticality.

(109) shows examples where the matrix predicate assigns accusative case.

- (109) a. Efχarístisa ópjus / *ópji me voiθísan.
I thanked_{acc} who_{acc} / *who_{nom} me I helped_{nom}
'I thanked whoever helped me.'

- b. Kálesa ópjus íða.
I invited_{acc} who_{acc} I saw_{acc}
'I invited whoever I saw.'
- c. Γnórisa ópjon *(tu) / *ópju éðosan tin ipotrofía.
I met_{acc} who_{acc} *(CL_{gen}) / who_{gen} gave_{gen} the scholarship
'I met whoever they gave the scholarship to.'

In contrast to the German 1 examples (75), Greek allows all case mismatches, as long as the wh-phrase bears the accusative case form. Finally, (110) shows examples where the matrix case is the genitive.

- (110) a. Éðosa leftá ópju me voíθise.
I gave_{gen} money who_{gen} me helped_{nom}
'I gave money to whoever helped me.'
- b. ?Tiléfonisa ópju íχα ḡósi leftá.
I phoned_{gen} who_{gen} I had given_{gen} money
'I phoned whoever I had given money to.'

Again, mismatches are fine as long the wh-phrase bears the case assigned by the matrix clause, which is genitive in the examples in (110). Summarizing these data in a table as in (111), the pattern of case matching effects in Modern Greek becomes apparent.

(111)

FR \ M	Nom	Acc	Gen
Nom	Nom	Nom	Nom (+ Cl)
Acc	Acc	Acc	Acc (+ Cl)
Gen	Gen	Gen	Gen

5. Complicating the Pattern

The empirical generalization for Modern Greek seems to be the following:

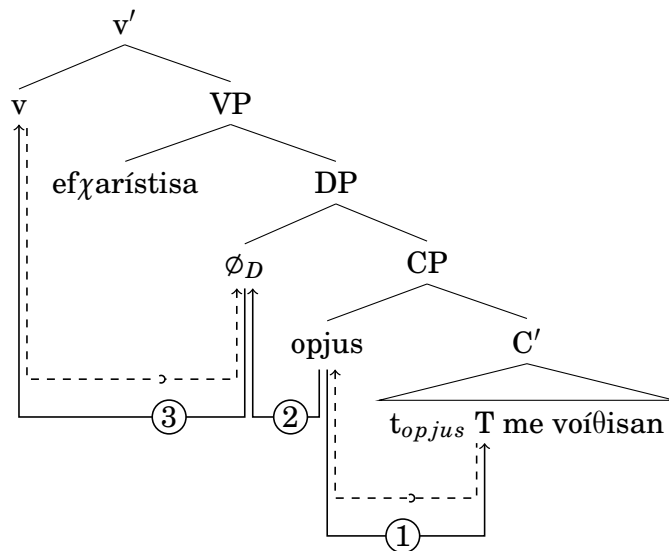
(112) *Case Matching in free relatives in Modern Greek*

The *wh*-phrase in free relatives has to bear the case assigned by the matrix clause.

This means that case mismatches are fine in free relatives in Greek and that there are no case hierarchy restrictions.

This pattern emerges if the overt *wh*-phrase in free relatives is the case probe and if the D head is not a case probe. Note, that in this case the symmetry of the Agree relation between arguments and functional heads does not matter. The tree in (113) shows the structure of a free relative in Greek. The dashed lines stand for Agree Links where a functional head probes for the case features.

(113) *Syntactic Agree-Link*



In the structure in (113), the wh-phrase *opjus* is the subject of the embedded verb *voíθisan* (“help”) and Agrees with the embedded T head. The free relative clause is the object of the matrix verb *efχarístisa* (“thank”) and Agrees with the matrix v. This structure leads to the following Agree-Copy relations in (114).

(114) *FRs: $\phi[acc]$, $\alpha[nom]$*

($T[*c:\{nom\}^*]$	\longrightarrow	$\alpha[*c:_*]:$	$T[c:\{nom\}]$
Step I	$\alpha[*c:_*]$	\longrightarrow	$T[c:\{nom\}]:$	$\alpha[*c:\{nom\}^*]$
Step II	$\alpha[*c:\{nom\}^*]$	\longrightarrow	$\phi[c:_]:$	$\alpha[c:\{nom\}]$
Step III	$\phi[c:_]$	\longrightarrow	$v[c:\{acc,nom\}]: \phi[c:\{acc,nom\}]$	
($v[*c:\{acc,nom\}^*]$	\longrightarrow	$\phi[c:acc,nom]:$	$v[c:\{acc,nom\}] (\phi \subseteq v)$

If Agree between functional heads and arguments is symmetric, the embedded T head first checks its case feature against the unvalued case feature of the wh-phrase. Next, the wh-phrase is valued first by T with nominative and then checked by the D head. Both Agree operations are unproblematic since there is no conflict between goal and probe. Afterwards, the D head receives accusative case from the matrix v. And even, if v checks its case feature at the end, no conflict arises. Thus, case mismatches are allowed in Greek free relatives. Note that this derivation resembles the derivation of parasitic gaps in Polish 1 (106).

The derivations in (113) and (114) raise the question why the wh-phrase overtly realizes the case of the matrix clause. Following Spyropoulos (2011) (see also Alexiadou and Varlokosta (1997, 2007)), I assume that (113) is the right structure for free relatives in Greek (but see Daskalaki (2011) for a different analysis). Consequently,

5. Complicating the Pattern

the concrete case realization in Greek free relatives must be due to a different operation that applies later in the post-syntactic derivation.

One possibility to account for the unexpected case realization would be context-sensitive vocabulary insertion rules. In the system of Arregi and Nevins (2012) these rules apply after Agree-Copy, thus, they should not affect case matching. The schema in (115) illustrates how these rules could work.

(115) [wh, {nom}, ...] \leftrightarrow *opjus* / [D, {acc,nom}] _

Due to this rule, the wh-phrase bearing nominative case is realized with an accusative form if it linearly follows a D head bearing accusative case.

There are two arguments that speak for this solution. First, the decision whether the matrix case or the embedded case is realized is determined by further factors in Greek: As noted by Spyropoulos (2011), the realization of the case depends on whether the case is inherent or not. This is illustrated in the contrast between (116-a) and (116-b) ((Spyropoulos 2011: 31f.)).

- (116) a. * θ a trayuðísun ópji / ópjus kerásame sfinákia
fut sing-3PL_{nom} who_{nom} / who_{acc} treat-PST.1PL_{acc} shots
'Those we treated (with) shots will sing.'
- b. to vrváo θ a to pári ópjós / *ópjón epiléksume
the price FUT CL take_{nom} who_{nom} / who_{acc} choose-1PL_{acc}
'The price will be awarded to whoever we choose.'

In both examples, the case assigned by the matrix clause is nominative and the case assigned by the embedded clause is accusative. However, in (116-a) the case conflict

cannot be resolved, while in (116-b), the structure is grammatical with nominative case on the *wh*-phrase. In the present system, this difference can be reduced to vocabulary insertion rules that distinguish between inherent and non-inherent cases, such as (117).

- (117) [wh, {acc,nom}_{noninh}, sg, ...] ↔ *opjos* / [D, {nom}] _

Independent of the concrete formulation, these rules would not be available to inherent cases.

A second argument for context-specific vocabulary insertion rules comes from linearization. In Greek, it can be observed that the realization of the case also depends on whether the free relative is in its base position or whether it has been moved to the left periphery. The difference can be seen in (118-a) and (118-b).

- (118) a. to vrvío θa to pári ópjós / *ópjon epiléksume
 the price FUT CL take3SG_{nom} who_{nom} / who_{acc} choose-1PL_{acc}
 ‘The price will be awarded to whoever we choose.’
- b. ópjós / ópjon epiléksume θa to pári to vrvío
 who_{nom} / who_{acc} choose-1PL_{acc} FUT take-3SG_{nom} the price
 ‘Whoever we may choose, he will get the price.’

While in (118-a), only nominative case is possible, both nominative and accusative case are possible for the sentence-initial *wh*-phrase in (118-b). Assuming that movement of the free relative can optionally strand the D head, the linear adjacency between the *wh*-phrase and the D head can be disturbed by movement. In these cases, the context-sensitive vocabulary insertion rule in (117) does not apply and the

5. Complicating the Pattern

wh-phrase realizes the accusative case of the embedded verb. If the D head is moved along, the rule in (117) applies and nominative case shows up overtly.

- (119) a. $[_{CP} \acute{o}pjon_{acc} \text{ epiléksume }] \theta a \text{ to pári } [_{DP} \emptyset_D t_{CP}] \text{ to vravío}$
 b. $[_{DP} \emptyset_D \acute{o}pjos_{nom} \text{ epiléksume }] \theta a \text{ to pári } t_{DP} \text{ to vravío}$

In sum, the analysis so far assumes that the realization of the matrix case on the wh-phrase is a language-specific peculiarity of Greek and not caused by a different structure for free relatives.

Turning to parasitic gaps in Greek, the prediction that the theory makes is that case mismatches in Greek are ungrammatical. This is so because case agreement between α and \emptyset is downward Agree. Again, the symmetry of Agree between functional heads and their arguments does not affect this outcome. This prediction seems to be true according to data such as (120).

- (120) a. $pion \text{ andhra}_{acc} \text{ pandreftike}_{acc} \text{ horis } na \text{ agapa}_{acc}$
 which man married without to love.acc
 ‘Which man did she marry without loving?’ Iatridou (1995)
 b. $*pion \text{ andra}_{acc} \text{ voithise}_{acc} \text{ horis } na \text{ dosi}_{gen} \text{ hrimata}$
 which man helped without to give money
 ‘Which man did she help without giving money to?’

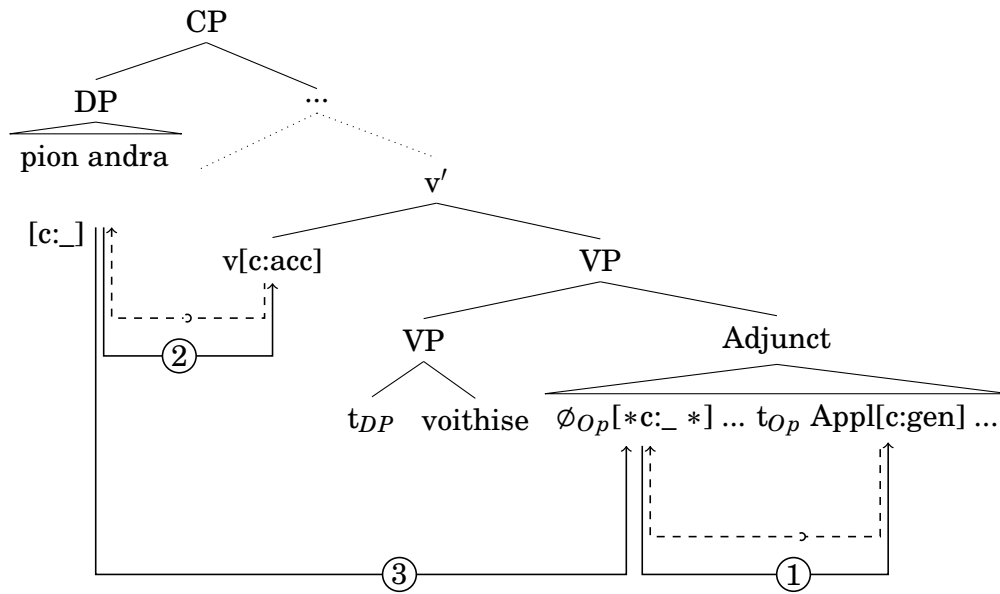
Artemis Alexiadou, p.c.

In the example in (120-a), the cases assigned by the embedded verb and the matrix verb are identical and the antecedent of the parasitic gap *pion andhra* (“which man”) receives accusative. Even though this construction is marked for some speakers (Androulakis (2001)), it seems to be fine for, at least some speakers. The example

in (120-b) shows a case mismatch. The embedded verb assigns genitive to the antecedent, while the matrix verb assigns accusative.¹⁹

The derivation in (121) and (122) shows why case mismatches in Greek parasitic gaps are not grammatical.

(121) *Syntactic Agree-Link*



¹⁹Note that inserting a clitic in the position of the parasitic gap remedies the violation of the case mismatch, as shown in (i). Assuming that the clitic construction is different from the parasitic gap construction, this difference is expected.

- (i) pion andra voithise horis na tu dosi hrimata
 which man helped without to CL-gen give money
 'Which man did she help without giving money to?'

Artemis Alexiadou, p.c.

A comparison of the parasitic gap construction to the free relative constructions above shows that in free relatives, the mismatch between accusative and genitive also requires a mismatch, see for example ((109-c)). However, the clitic in ((109-c)) does not occur in the position of the covert D head but inside the embedded clause, assuming that the wh-phrase is also part of the clause. The situation is different in (i). Here, the clitic *tu* is inside the parasitic gap clause, where it can replace the gap.

5. Complicating the Pattern

The structure in (121) is reminiscent of the structure of parasitic gaps in Polish 1 (106), the only difference being that Agree between the antecedent of the parasitic gap and the operator is downward Agree. As in (113), the dashed lines indicate case Agree triggered by functional heads. The post-syntactic derivation is summarized in (122).

(122) *PG: $\phi[gen]$, $\alpha[acc]$*

	$Appl[*c:\{gen,acc,nom\}^*]$	\longrightarrow	$\phi[c:_]:$	$Appl[c:\{gen,acc,nom\}]$
Step I	$\phi[c:_]$	\longrightarrow	$Appl[c:\{gen,acc,nom\}]:$	$\phi[c:\{gen,acc,nom\}]$
	$v[*c:\{acc,nom\}^*]$	\longrightarrow	$\alpha[*c:_]:$	$v[c:\{acc,nom\}]$
Step II	$\alpha[*c:_]$	\longrightarrow	$v[c:\{acc,nom\}]:$	$\alpha[*c:\{acc,nom\}^*]$
Step III	$\alpha[*c:\{acc,nom\}^*]$	\longrightarrow	$\phi[c:\{gen,acc,nom\}]:$	$\nmid (\phi \neq \alpha)$

Independent of whether functional heads are case probes, the derivation crashes because of the final downward Agree relation in Step III. At this point, both the wh-phrase and the operator have received a case value. If these values are not identical, the matching condition relevant for Agree in sharing constructions is violated. This rules out case mismatches in parasitic gaps in Greek.

5.3.4. The Unknown Pattern

The combination of parameters also predict a fifth possible pattern, the existence which is yet to be confirmed by empirical data. In this pattern, only the covert elements of sharing constructions, but not functional heads and not overtly shared elements trigger additional case agreement relations. In this pattern, case mismatches

should be possible in free relatives, as long as they follow the case hierarchy, and case mismatches should be also be possible in parasitic gaps without any restrictions. In this way, this pattern also combines a property of German 1 (for free relatives) and Polish 1 (for parasitic gaps. Therefore, this pattern is called German 2' or Polish 2' respectively. (123) and (124) illustrates this pattern for German examples with fictitious judgments.

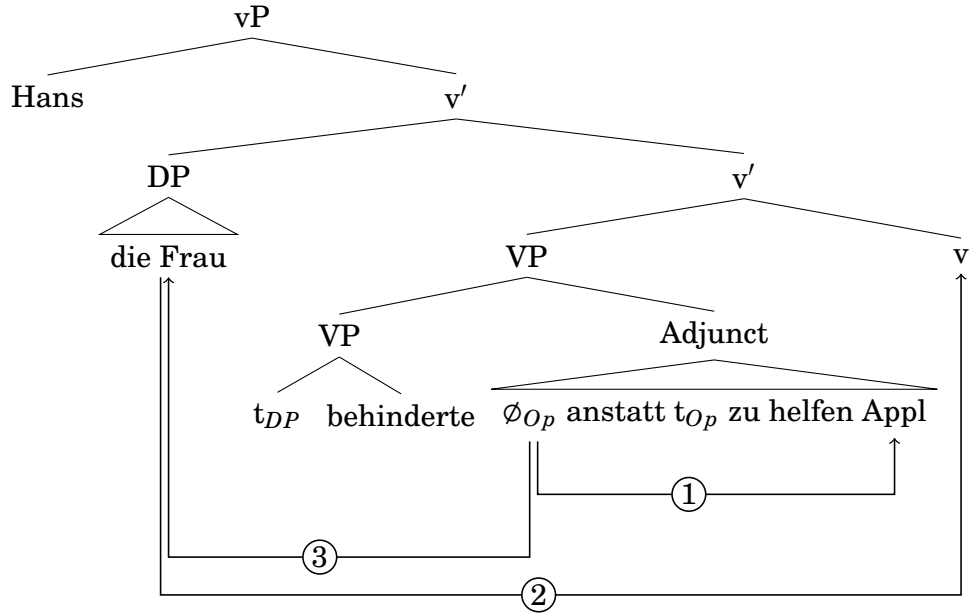
- (123) a. weil Hans *der_{dat}/die_{acc} Frau [anstatt zu helfen_{dat}]
 because Hans the woman instead.of to help
 behinderte_{acc}
 hampered
 'because Hans hampered the woman instead of helping her'
- b. weil Hans der_{dat}/*die_{acc} Frau [anstatt zu behindern_{acc}]
 because Hans the woman instead.of to hamper
 half_{dat}
 helped
 'because Hans helped the woman instead of hampering her'
- (124) a. Hans mag_{acc} [*wen_{acc}/wem_{dat} (auch immer) Maria vertraut_{dat}].
 Hans likes who ever Maria trusts
 'Hans likes whoever Maria trusts.'
- b. Hans vertraut_{dat} [*wen_{acc}/*wem_{dat} (auch immer) Maria mag_{acc}].
 Hans trusts who ever Maria likes
 'Hans trusts whoever Maria likes.'

In (123), case mismatches in parasitic gap structures are fine, as long as the overt phrase bears the case of the matrix clause (cf. the Polish data in (23)). In (124), the pattern of German 1 for free relatives is repeated: If the case assigned by the matrix clause is higher on the hierarchy than the case assigned within the embedded clause, the wh-phrase in the free relative may bear the case assigned to it inside the embedded clause.

5. Complicating the Pattern

Now, assuming the distribution of probes for this pattern as in table (95), the derivation of a parasitic gap construction in German 2'/Polish 2' would proceed as in (125) and (126).

(125) *Syntactic Agree-Link*



This derivation has only three Agree-Links: one between the operator and the embedded applicative head, one between the direct object *die Frau* (“the woman”) in the matrix clause and the matrix *v* head, and one between the operator and *die Frau*. This leads to the order of Agree-Copy operations in (126).

(126) *PG: ϕ [dat], α [acc]*

Step I	$\phi[*c:_*]$	\rightarrow	Appl[c:{dat,acc,nom}]:	$\phi[*c:{dat,acc,nom}*]$
Step II	$\phi[*c:{dat,acc,nom}*]$	\rightarrow	$\alpha[c:_]$:	$\phi[c:{dat,acc,nom}]$
Step III	$\alpha[c:_]$	\rightarrow	$v[c:{acc,nom}]$:	$\alpha[c:{acc,nom}]$

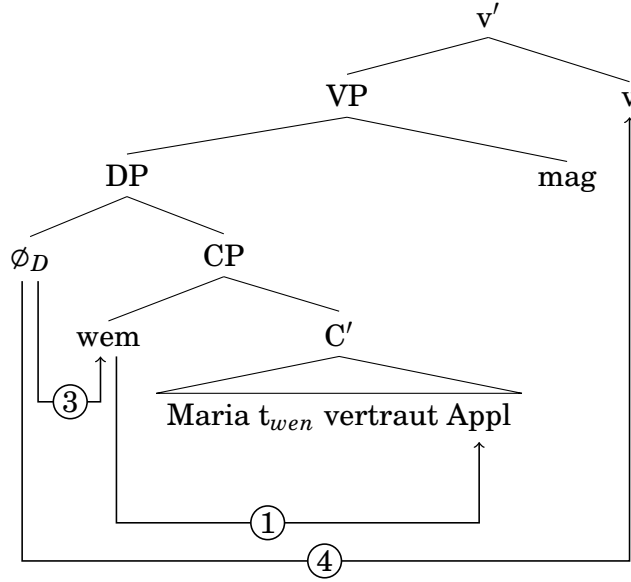
As shown in (126), all Agree relations trivially fulfill the matching condition since either the probe or the goal is unvalued. Therefore, case mismatches are fine in German 2'/Polish 2' parasitic gap constructions independent of the concrete cases that α and ϕ receive (see (127)).

(127) *PG: $\phi[acc]$, $\alpha[dat]$*

Step I	$\phi[*c:_*]$	\rightarrow	$v[c:\{acc,nom\}]:$	$\phi[*c:\{acc,nom\}^*]$
Step II	$\phi[*c:\{acc,nom\}^*]$	\rightarrow	$\alpha[c:_]:$	$\phi[c:\{acc,nom\}]$
Step III	$\alpha[c:_]$	\rightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\alpha[c:\{dat,acc,nom\}]$

As for free relatives, the derivation in (128) and (129) shows, why case hierarchy effects are expected in this pattern.

(128) *Syntactic Agree-Link*



(129) *FR: $\phi[acc]$, $\alpha[dat]$*

Step I	$\alpha[c:_]$	\rightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\alpha[c:\{dat,acc,nom\}]$
Step II	$\phi[*c:_*]$	\rightarrow	$\alpha[c:\{dat,acc,nom\}]:$	$\phi[*c:\{dat,acc,nom\}^*]$
Step III	$\phi[*c:\{dat,acc,nom\}^*]$	\rightarrow	$v[c:\{acc,nom\}]:$	$\phi[c:\{dat,acc,nom\}]$ ($v \subseteq \phi$)

5. Complicating the Pattern

Crucially, the case hierarchy plays a role because Agree with functional heads is asymmetric. Symmetric Agree would lead to an identity requirement. And due to downward Agree between the D head and the *wh*-phrase, the D head bears the dative case of the embedded applicative head (Step II). Thus, the constellation of the embedded case and the matrix case has to follow the case hierarchy restrictions for free relatives (Step III). Based on this, a configuration where the cases are reversed is expected to be ungrammatical. This is shown in (130).

(130) *FR: $\emptyset[dat]$, $\alpha[acc]$*

Step I	$\alpha[c:_]$	\longrightarrow	$v[c:\{acc,nom\}]:$	$\alpha[c:\{acc,nom\}]$
Step II	$\emptyset[*c:_*]$	\longrightarrow	$\alpha[c:\{acc,nom\}]:$	$\emptyset[*c:\{acc,nom\}^*]$
Step III	$\emptyset[*c:\{acc,nom\}^*]$	\longrightarrow	$Appl[c:\{dat,acc,nom\}]:$	$\nmid (Appl \not\subseteq \emptyset)$

5.3.5. Interim Summary

In this chapter, I have reviewed and discussed a number of different case matching patterns, extending the basic system of chapter 4 further. There have been five adjustments: First, syncretism rules were included to derive the syncretism effects of case matching (section 5.1). Second, the case values were decomposed in order to represent the case hierarchy (section 5.2.2.1). Third, the number of possible probes has been increased, including case assigning functional heads (section 5.2.2.2). Fourth, the matching condition of Agree has been refined, defining the issue of feature conflict between probe and goal exactly (section 5.2.2.2, (84)). And finally, the optionality concerning the order of Agree-Copy operations was eliminated (section 5.2.2.2). Leaving cross-linguistic variation to the question, whether a case feature

is a probe or just the goal of Agree, this extended system makes the prediction that there are five language types with respect to case matching in free relatives and parasitic gaps. So far, I was able to find examples for four of these types. The summary of the patterns is repeated in (131).

(131) *Typology of Case Matching*

F [*c*]?	α [*c*]?	\emptyset [*c*]?	mismatch FR?	mismatch PG?	Pattern
✓	✓	✓	*	*	German 2, Polish 2 (sec. 5.3.1)
		*	✓	*	Greek (sec. 5.3.3)
	*	✓	*	✓	Polish 1 (sec. 5.3.2)
		*	—	—	—
*	✓	✓	hierarchy	*	German 1 (sec. 5.2)
		*	✓	*	Greek (sec. 5.3.3)
	*	✓	hierarchy	✓	??? (sec. 5.3.4)
		*	—	—	—

Even though the system developed so far is intricate and based on very specific conditions for Agree, it is empirically superior insofar as it is able to derive the fact that case hierarchy effects in case matching are virtually absent in parasitic gap constructions cross-linguistically and that the similarities in case matching between free relatives and parasitic gaps as well as the cross-linguistic variation is not a coincidence but is due to the fact that both constructions follow the same mechanisms of case assignment and case agreement.

In the second part of this thesis, I will provide further evidence that parasitic

5. Complicating the Pattern

gaps and free relatives can be analyzed alike and I will provide a closer look at the construction specific agreement relation between the overt and the covert phrase.

Before turning to these points, I will discuss possible theoretical alternatives to the agreement-based approach developed in this part of the thesis.

6. Alternatives

The final question about the adequacy of the present approach is whether there are any alternatives accounts. The number of possibilities to analyze sharing constructions is limited. In principle, there are three strategies: First, there are agreement approaches like the present account: The main assumption these analyses are based on is that the structure of sharing constructions contains an covert category additional to the overt shared category and that the two categories communicate in some way – usually by some form of agreement (Chomsky (1982); Engdahl (1983); Chomsky (1986); Cinque (1990); Nissenbaum (2000); Kuroda (1968); Bresnan and Grimshaw (1978); Groos and Riemsdijk (1981); Hirschbühler and Rivero (1981); Harbert (1983); Suñer (1984); Grosu and Landman (1998); Grosu (2003); Caponigro (2002); Gračanin-Yuksek (2008)).

The second type of approach can be summarized as identity approach: Here, the shared category is the only category and the additional syntactic dependency between the shared category and the matrix clause (in case of free relatives) or the embedded clause (in case of parasitic gaps) is modeled differently. Such approaches can be multidominance accounts such as Riemsdijk (2006); Kasai (2008); Citko (2005, 2013) or movement accounts (Huybregts and van Riemsdijk (1985); Bennis and

6. *Alternatives*

Hoekstra (1985); Williams (1990); Nunes (2004); Rooryck (1994); Caponigro (2003); Donati and Cecchetto (2011); Ott (2011))).

Finally, the third possible type of approach is a reanalysis strategy. The core idea in this type of approach is to treat free relatives and parasitic gaps differently in different languages. For example, in some languages, an agreement approach is suitable for a certain sharing construction but not for another one.

The present analysis of case matching effects is an example for how agreement approaches can deal with the patterns discussed in this part. In this section, I will discuss how the two alternative types of approaches can handle the data.

6.1. Reanalysis

In what follows, the reanalysis strategy is discussed on the basis of two examples for this strategy. The concrete problems of this example do not translate to all analysis of this strategy but there is a general issue that all reanalysis accounts face.

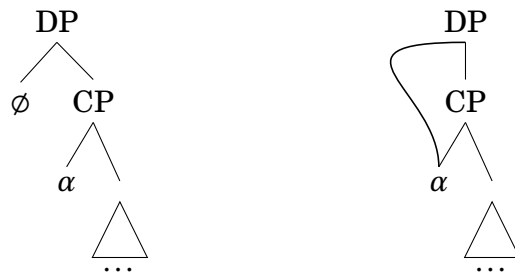
6.1.1. A Hybrid Account

The first example for reanalysis is a possible analysis based on the discussion above. In this approach, strict matching is due to identity: there is only one element that has to satisfy the case requirements of two verbs. These requirements can only be met, if the cases assigned by the two verbs are identical. The absence of matching effects, on the other hand, arises in structures where a second, covert, category is involved. Such an approach uses both identity and non-identity and can hence be called a hybrid account. The structures in (132) sketch this idea for free relatives,

the structures in (133) show this for parasitic gaps.

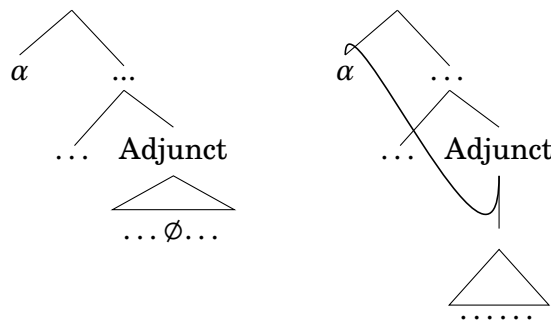
(132) *Free relatives*

- a. ✓*case mismatch* b. **case mismatch*



(133) *Parasitic Gaps*

- a. ✓*case mismatch* b. **case mismatch*



The trees in (132-a) and (133-a) show structures that allow case mismatches. Here, the overt α and the covert element \emptyset receive cases independently. Since in this approach they do not agree in case, case mismatches should be fine. In the trees in (132-b) and (133-b), there is only α that receives two cases by two different functional heads. Depending on how matching is defined, this should either result in strict case matching or in case hierarchy effects.

This solution to variation is very simple but requires additional evidence that the parasitic gaps and free relatives really have different derivations in different

6. Alternatives

languages. As for Polish and German, there seem to be no major differences concerning parasitic gaps and free relatives. First, comparing the property of island sensitivity (cf. Kayne (1983); Chomsky (1986)) in parasitic gaps in (134) and (135), both languages behave alike.²⁰

(134) German

- a. Welchen Artikel hast du [isl ohne pg zu lesen] *t* abgeheftet?
which article have you without to read filed
'Which article did you file without reading?'
- b. *Welchen Artikel hast du [isl ohne den Autor zu kennen [isl der
which article have you without the author to know who
pg geschrieben hat]] *t* abgeheftet?
written has filed
'Which article did you file without knowing the author who wrote?'

The German example in (134-b) is ungrammatical because the parasitic gap and its antecedent are separated by an additional relative clause. Assuming some sort of locality requirement between the antecedent and the parasitic gap (or the operator respectively), the parasitic gap cannot be licensed due to a lack of locality.

Examples similar to (134-b) can be found in Polish as well (Bondaruk 1996: 112).

- (135) *Kogo polubiłeś *t* [isl po tym jak rozmawiałeś z mężczyzną [isl który
who like-2SG.PST after talk-2SG.PST to man who
znał pg]]
knew-3SG
'Who did you like after you were talking to the man who knew (him)?'

²⁰In chapter 8, I will go through all the properties of parasitic gaps discussed here, in detail. For now,

the main point is that German and Polish are similar in these respects.

Thus, island sensitivity is a property which parasitic gaps in German and Polish have in common.

The next crucial property of parasitic gaps to look at are categorial restrictions (Postal (1993)). In both German and Polish, adverbial phrases seem to be ruled out as antecedents for parasitic gaps.

- (136) a. *... dass Hans laut [anstatt *pg* ein Lied zu pfeifen] *t* ein Lied
 that Hans loudly instead.of a song to whistle a song
 gesungen hat
 sung has
 ‘that Hans loudly sang a song instead of (loudly) whistling a song’
- b. *Jak głośno Janek śpiewał *t* [zanim zagrał *pg*]?
 how loudly Janek sang before played
 ‘How loudly did Janek sing before playing (loudly)?’

(Bondaruk 1996: 114)

In both the German (136-a) and the Polish (136-b), *loudly* can only be interpreted as an adverb to the matrix verb. A parasitic gap reading is excluded.

A third property that both languages have in common and that is a typical property of parasitic gaps is the anti-c-command condition (Engdahl (1983)). This condition bans subjects from licensing parasitic gaps. Illustrated by the examples in (137), this condition is active in both German and Polish.²¹

²¹Note that in German parasitic gaps can only occur in infinite adjunct clauses. In these clauses the subject is controlled by the matrix subject, represented as PRO in (137-a). If the subject were to license the object parasitic gap additionally to controlling the subject PRO, a reflexive reading would arise (cf. Kathol (2001)). As (137-a) shows, this reflexive reading is not possible, which means that a subject in German cannot license a parasitic gap. Even though this specific case might be reduced to a different constraint active in control structures, the empirical observation

6. Alternatives

- (137) a. *... dass Hans [anstatt PRO *pg* zu waschen] Parfüm versprüht
that Hans instead.of to wash perfume sprayed
hat
has
'that Hans has sprayed some perfume instead of washing (himself)'
- b. *Jakich materiałów *t* zabrakło [żeby użyć *pg* do pisania
which materials were.lacking in.order.to use to write
artykulu?]
paper
'Which materials were lacking to be used to write a paper?'

(Bondaruk 1996: 119)

These three comparisons of German and Polish parasitic gaps show that they both exhibit essential structural properties of parasitic gaps. Therefore, an approach that treats this construction differently in the two languages solely on the basis of a difference in the case matching property appears to be uncalled-for. The similarities seem to be strong and should not be treated as a coincidence.

As for free relatives, typical properties are less salient, but still, German and Polish seem to be strikingly similar.

A first major property of free relatives especially highlighted by Caponigro (2003) is that, in free relatives, the relative pronoun is replaced by a *wh*-phrase. As seen throughout this part of the thesis, this is the case for Polish and German as well. Two simple examples are given in (138).

- (138) a. Hans mag [_{FR} wen /*die Maria *t* hasst].
Hans likes who /*rel Maria hates
'Hans likes whoever Maria hates.'
- b. [_{FR} Kto /*którzy ciężko pracuje] czuje się zmęczony ale szczęśliwy.
who /*rel hard works feels self tired but happy
'He who works hard feels tired but happy.' (Caponigro 2003: 27)

is, nevertheless, the same for Polish and German.

These examples also illustrate a second major property of free relatives: the relative clause contains a gap ((Bresnan and Grimshaw 1978: 331)). In both (138-a) and (138-b), the base position of the direct object is not filled by a clitic or a resumptive or any other overt element.

Finally, it should be noted that free relatives should be replaceable by a truth-conditionally equivalent DP or PP (Donati and Cecchetto (2011)). This property can be seen in (139).

- (139) a. Hans mag [_{DP} die Personen, die Maria *t* hasst].
 Hans likes the people that Mary hates
 ‘Hans likes the people that Mary hates.’
- b. [_{DP} Ludzie którzy pracują ciężko] czują się zmęczeni ale szczęśliwi.
 people rel hard work feel self tired but happy
 ‘People who work hard feel tired but happy.’ (Caponigro 2003: 27)

Again, the two languages behave alike with respect to the basic properties of free relatives. Thus, a reanalysis strategy would have to capture these similarities differently.

The tables in (140) and (141) summarize the facts presented above.

(140) *Parasitic Gaps*

Property	<i>German</i>	<i>Polish</i>
island sensitivity	✓	✓
categorial restrictions	✓	✓
anti-c-command-condition	✓	✓
tensed environments	*	✓

6. Alternatives

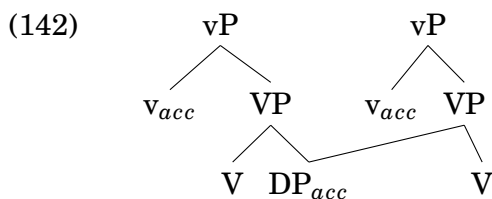
(141) *Free Relatives*

Property	<i>German</i>	<i>Polish</i>
wh-phrase instead of relative pronoun	✓	✓
clause with gap	✓	✓
truth-conditionally equivalent DP or PP	✓	✓

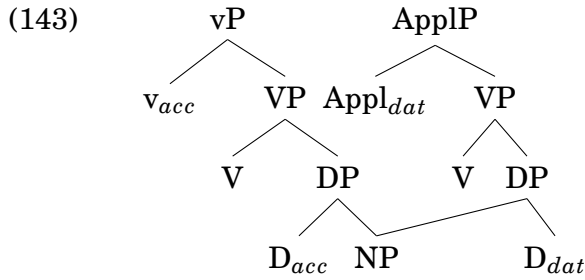
Judging from the similarities between German and Polish, there seems to be no motivation for analyzing the two constructions differently in the two languages.

6.1.2. Citko (2013)

A different kind of reanalysis approach is presented in Citko (2013) for Polish sharing constructions. The main idea of this account builds on multidominance: Strict matching occurs when a DP is subject to multidominance. The case feature, which is located on D, is shared between two verbs. Therefore, it has to match the requirements of both verbs. This configuration is sketched in (142).



Mismatching occurs if an NP is dominated by two DPs with one case feature each. Consequently, the two case features can receive two different values and the absence of matching effects is predicted. This is shown in (143).



The analysis faces the same problem that the hybrid account faces: German parasitic gaps have to have a different structure than Polish parasitic gaps and German free relatives have to have a different structure than Polish free relatives. In Polish parasitic gaps structures, only an NP is shared between two clauses, while in free relative contexts, the entire DP with only one case feature is shared. In German, parasitic gaps require sharing of a DP and free relatives sharing of an NP. Again, there is no independent evidence that confirms these major structural differences. The absence of a derivation of the cross-linguistic similarities of these constructions can thus be seen as a general problem of the reanalysis strategy.

A further, more specific problem to the account in Citko (2013) concerns case concord inside the DP. Morphologically, case is also realized on nouns. Thus, NPs must also bear a case feature. If the NP is shared between two D heads with conflicting case features, the single case feature on the NP cannot meet the requirements of both D heads. Consequently, the violation of matching is expected to occur on the NP level. Hence, the analysis does not solve the problem of matching, it just shifts the problem to a different area.

6.2. Identity

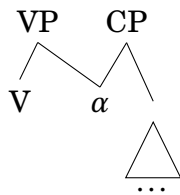
There are two kinds of identity approaches: multidominance approaches and movement approaches.

6. Alternatives

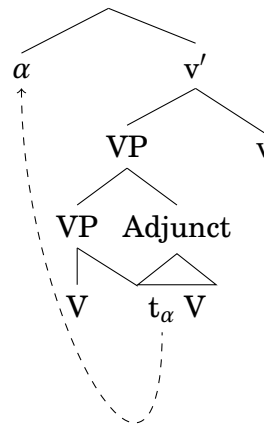
6.2.1. Multidominance

Multidominance can be depicted as in (144) and (145): The overt category is shared between two verbs and is, thus, dependent on the case requirements of both verbs.

(144) *Free Relatives*



(145) *Parasitic Gaps*



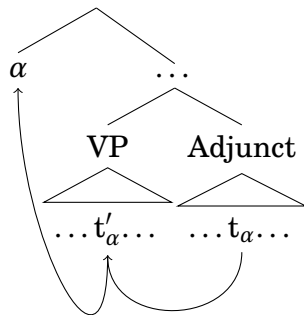
Such multidominance or *grafting* approaches have a problem explaining the variation in case matching illustrated in chapter 5. The cross-linguistic variation can be handled by assuming that in one language a certain case matching condition holds while in another language, there is no case matching condition or the case matching conditions are defined differently. However, taking that abstractly, free relatives and parasitic gaps are derived by the same theoretical means, namely multidominance, both constructions are predicted to behave alike in a certain language: They should either show case matching effects or not. The only way out would be to assume that case the matching condition varies from construction to construction – for some construction case matching is obligatory, for some it is subject to the case hierarchy, and for some constructions, no case matching condition holds.

There are two potential problems with this reasoning. First, this line of thoughts amounts to a reanalysis strategy and therefore might face the same general problem as the accounts discussed in section 6.1. Second, the assumption of construction-specific case matching conditions predicts that there should be languages which do not show matching effects in any of the two constructions. As far as the literature goes, this pattern does not exist. In any case however, the agreement approach argued for in this thesis and multidominance approaches do make different empirical predictions.

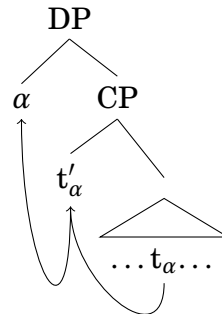
6.2.2. Movement

In movement approaches, the overt category is merged in the embedded clause, where it receives case. Afterwards, it moves to the respective argument position in the superordinate clause, before it reaches its target position. In the higher clause the case features received in the embedded clause have to match the new case requirements. The abstract derivation is shown in (146) and (147).

(146) *Parasitic Gaps*



(147) *Free Relatives*



The account faces the same problems as the multidominance account when it comes to the matching effects because the abstract derivation of the two constructions is

6. *Alternatives*

the same. Again, construction specific case conditions might have to be invoked in order to derive the observed variation.

6.3. Agreement

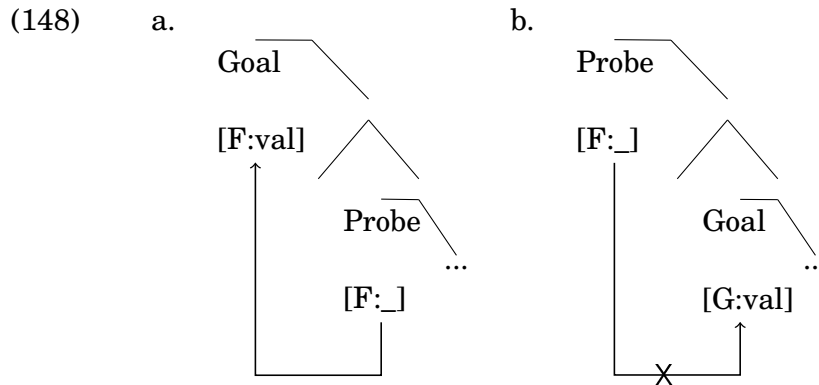
So far, we have seen that the reanalysis and the identity strategy show certain empirical problems and might also face conceptual issues. An agreement approach, therefore, seems to be the most promising strategy, since it is flexible enough to capture variation in case matching, but also rigid enough to derive structural similarities between languages.

Even though the present system of case matching follows the agreement strategy, it is based on a very specific concept of agreement, which is a bidirectional two-step Agree operation. It is therefore only fair to compare this specific understanding of agreement to alternative theories of agreement. In the following discussions, I will challenge the assumptions that Agree is bidirectional and that Agree applies in two steps, concluding that the present approach is superior to alternative approaches that do without these assumptions. Note that the alternatives are also based on the assumption that agreement results from an operation Agree. First, I will look at two alternative analyses that are based on a unidirectional Agree operation. Afterwards, I will look at an account based on a single Agree-operation.

6.3.1. Only Upward Agree

The first alternative to bidirectional Agree is an approach where Agree applies only upward with the goal c-commanding the probe. Downward Agree is impossible (see

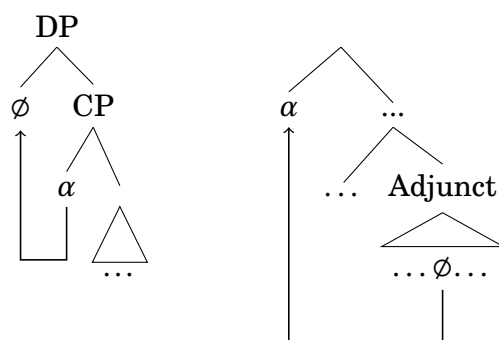
Zeijlstra (2012).) The approach is schematized in (148).



In such an approach, case assignment from functional heads would be analyzed as Agree between a probing argument and the case assigning head (see also Pesetsky and Torrego (2007)).

As for case agreement in parasitic gap and free relative constructions, it might not be possible to reduce this process to the operation Agree. If case agreement were Agree, it would always have to apply upward. But then, the difference in case matching between the constructions could not be derived. because both constructions in all languages would abstractly have the same derivation, as depicted in (149).

(149) a. *Free relatives* b. *Parasitic gaps*



6. Alternatives

Alternatively, case agreement in sharing constructions have to be derived by a different operation. A promising version of this idea is that case assignment is syntactic upward Agree and that case agreement is a postsyntactic rule. The difference between strict case matching, case hierarchy effects and the absence of case matching can then be derived as exemplified in (150) for a free relative, where the *wh*-phrase receives dative case and the D head receives accusative case.

(150) a. *Strict case matching*

Syntax	$\alpha[c: _]$	\longrightarrow	Appl[c:{dat,acc,nom}]:	$\alpha[c:\{\text{dat,acc,nom}\}]$
	$\emptyset[c: _]$	\longrightarrow	v[c:{acc,nom}]:	$\emptyset[c:\{\text{acc,nom}\}]$
Morphology	$\alpha[c:\{\text{dat,acc,nom}\}]$	\longleftrightarrow	$\emptyset[c:\{\text{acc,nom}\}]$:	$\nexists (\alpha \not\subseteq \emptyset)$
b. <i>Case hierarchy effect</i>				
Syntax	$\alpha[c: _]$	\longrightarrow	Appl[c:{dat,acc,nom}]:	$\alpha[c:\{\text{dat,acc,nom}\}]$
	$\emptyset[c: _]$	\longrightarrow	v[c:{acc,nom}]:	$\emptyset[c:\{\text{acc,nom}\}]$
Morphology	$\alpha[c:\{\text{dat,acc,nom}\}]$	\longrightarrow	$\emptyset[c:\{\text{acc,nom}\}]$:	$\checkmark (\emptyset \subseteq \alpha)$
c. <i>Absence of case matching</i>				
Syntax	$\alpha[c: _]$	\longrightarrow	Appl[c:{dat,acc,nom}]:	$\alpha[c:\{\text{dat,acc,nom}\}]$
	$\emptyset[c: _]$	\longrightarrow	v[c:{acc,nom}]:	$\emptyset[c:\{\text{acc,nom}\}]$
Morphology	$\alpha[c:\{\text{dat,acc,nom}\}]$		$\emptyset[c:\{\text{acc,nom}\}]$:	\checkmark

In (150), the matching condition for post-syntactic case agreement between α and \emptyset is the subset condition, where the case value of the goal of agreement has to be a subset of the probe. Strict case matching arises in (150-a) because this case agreement is symmetric, meaning that the subset condition has to hold both ways. In (150-b) the agreement is asymmetric, which results in case hierarchy effects. Finally in (150-c), the agreement relation is completely absent and case matching effects

do not result. In order to derive the different patterns of case matching effects in table (95), the post-syntactic agreement relations have to apply in one of the three possible ways according to the empirical facts.

So far, these assumptions in a unidirectional Agree approach could derive the empirical facts but there are two major issues that this account raises: First, similar to the identity strategy, this account predicts the existence of a language, where neither free relatives nor parasitic gaps show case matching or case hierarchy effects. Second, the assumption of a bidirectional agreement relation is shifted to another Agree-like process.²²

This second problem is aggravated by the fact that case agreement in sharing constructions shows some of the core properties of the Agree operation. First, c-command between α and \emptyset has to be given at some point of the derivation. This can be seen in parasitic gap constructions. If the antecedent of the parasitic gap stays in-situ, not creating c-command by movement, the construction is ungrammatical, as shown in (151) for English and German. This is also known as the ban on in-situ licensing of parasitic gap (Engdahl (1983)).

- (151) a. *I forget who filed which articles₁ [without reading pg_1]?

(Engdahl 1983: 114)

- b. *... dass Hans [anstatt pg_2 zu singen] ein Lied₂ gepfiffen hat.
 that Hans instead.of to sing a song whistled has
 ‘that Hans whistled a song instead of singing (it)’

²²Alternatively, case matching effects could be due to a genuinely different morphosyntactic process. However, having excluded movement and multidominance as alternatives, since these processes are too rigid to account for variation, it is unclear which morphosyntactic process could be responsible for case matching.

6. Alternatives

A second similarity between Agree and case agreement is locality: Agree relations usually do not cross certain absolute boundaries, such as finite clauses or islands (see Brattico (2009); Vainikka and Brattico (2014)). The Finnish examples in (152) ((Vainikka and Brattico 2014: 104,89)) show that temporal adjuncts are islands (152-a) and that accusative case assignment in a temporal adjunct is not influenced by properties of the matrix clause (152-b).

- (152) a. *?Kenet lähditte kotiin heti [Maijan löydettyä t
 who.ACC went.2PL home immediately Maija.GEN found.ESSA.PST
 pihalta]?
 yard.ABL
 [lit.] ‘Who did you went home after Maija found in the backyard?’
- b. [Maijan löydettyä sisko-n / *sisko] meidän
 Maija.GEN find.ESSA.PST sister-ACC *sister we.GEN
 täytyi lähteä kotiin.
 must.PST.3SG leave.A home
 ‘After Maija found you/the sister, we had to go home.’

This condition on absolute locality also applies to parasitic gap configurations in German. Examples, such as (153), where the empty operator and the antecedent are separated by a relative clause boundary are ungrammatical despite case matching. This follows if Agree cannot apply across a relative clause boundary.²³

- (153) *weil Hans die_{acc} Frau [anstatt einen Freund zu treffen [Op_{acc}
 because Hans the woman instead.of a friend to meet
 der t_{Op} unterstützen_{acc} könnte]] behinderte_{acc}
 who support could hampered
 ‘because Hans hampered the woman instead of meeting a friend who could
 help (her)’

²³Note that locality is also a property of other syntactic operations such as movement. Therefore, these data also follow from different assumptions. Still, the point is that locality does play a role in the relation between an operator and its antecedent in parasitic gap constructions.

The last crucial property of Agree that also plays a role in parasitic gap constructions is intervention. An example for intervention in case assignment is given in (154) from Japanese (Ura 2007: 9f).

- (154) a. Taro-wa [Hanako-o utsukushi-i to omot-ta.]
 Taro-TOP Hanako-ACC beautiful-be COMP consider-PST
 ‘Taro considered that Hanako was beautiful.’
 b. Boku-wa [_{CP} John-ni sono koto-ga / *koto-o deki-soo-da
 I-TOP John-DAT the task-NOM task-ACC able-likely-be.PRES
 to] omow-u.
 COMP think-PRES
 ‘I think that John is likely to be able to do the task.’

The accusative case on the embedded subject in (154-a) comes from the matrix verb *omota*. This accusative case assignment to the embedded object is blocked in (154-b) since the embedded T, which assigns nominative case, is also a case assigner and closer to the object. Due to the intervention property of Agree, the object must receive the closest case possible.

A similar intervention effect can also be observed in parasitic gap configurations in German, as shown in (155) (Heck and Himmelreich (2017)).²⁴

- (155) a. dass Hans der Maria_{dat} das Buch_{acc} [\emptyset _{Op_{acc}} ohne *t*_{Op}
 that Hans the Maria the book without
 durchzulesen] zurückgibt
 through.to.read back.gives
 ‘that Hans returns the book to Maria without reading (it) through’

²⁴Note that this is a case of opaque intervention where linear order is not responsible for the intervention effect. Opaque intervention arises, if intervention is given at the relevant point in the derivation, but disappears on the surface due to subsequent movement steps (see Heck and Himmelreich (2017) for details). The data in (155) show that intervention does play a role in these constructions. The full picture will be discussed in section 6.3.4.

6. Alternatives

- b. *wenn jemand der Anette_{dat} das Buch_{acc} [\emptyset _{Op_{dat}} ohne _{t_{Op}} zu
if someone the Anette the book without to
vertrauen] ausleiht
trust lends
'if someone lends Anette the book without trusting her'

In (155-a), the operator can have the same case as the direct object *das Buch* ("the book") since there is no other phrase that intervenes. In (155-b), on the other hand, *das Buch* intervenes between *der Anette* and the operator. Therefore, dative case agreement is blocked here.

In conclusion, the problems of an account based on unidirectional upward Agree, on the one hand, is that it shifts the bidirectionality of case agreement in sharing constructions to a different process and and, on the other hand, that it cannot capture the striking similarity between Agree and case agreement.

6.3.2. Only Downward Agree

In an approach where Agree may only apply downwards, case assignment must be a process different from Agree (see e.g. Chomsky (2001)), since arguments (the case probes) are in most cases c-commanded by their respective case assigners. Case agreement in free relatives and parasitic gap constructions are genuine cases of Agree in this approach. The empirically strongest version of this alternative account is very similar to the account proposed in this part: Agree is a two-step operation with Agree-Link and Agree-Copy working as described above. There are only two differences: First, arguments get their case features valued as a reflex of ϕ -Agree with a functional head and, second, Agree-Links can only be established in the syntax if the probe c-commands the goal. The three possibilities for case matching

are then derived as shown in (156) for a free relative with a dative-accusative case mismatch. Note that only the post-syntactic Agree-Copy operations are shown in (156).

(156) a. *Strict case matching*

Step I	$\alpha[c: _]$	\leftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\rightarrow	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\leftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\nexists (\alpha \neq v)$

b. *Case hierarchy effect*

Step I	$\alpha[c: _]$	\leftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\rightarrow	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\leftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[c:\{\text{dat}, \text{acc}, \text{nom}\}] (v \subseteq \emptyset)$

c. *Absence of case matching*

Step I	$\alpha[c: _]$	\leftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
Step II	$\emptyset[*c: _ *]$	\rightarrow	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\leftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
(no requirement)				

The derivations already show a major problem of this account: the difference between strict case matching and case hierarchy effects can only be derived if the matching condition of Agree is language-specific. In a language with the pattern of German 1, the D head can get a case value from the matrix v , if the subset condition is fulfilled, in a language with the Polish 1 pattern, the identity condition is imposed, and in a Greek-like language, no requirement needs to be met. This does derive all the variation for free relatives, but taking parasitic gaps into the picture, a problem shows: In order to rule out case hierarchy effects for parasitic gaps, the matching

6. Alternatives

condition for parasitic gaps can only be identity of case values. Strict case matching emerges if this condition is active in a language and case matching is missing if the identity condition is missing. All these assumptions, however, amount to a mere restatement of the empirical findings. The distribution of case matching effects in a language is coincidental and it is further predicted that a language should exist where no case matching is required at all (cf. the discussion in section 6.2.1).

In sum, both versions of a theory based on unidirectional Agree face problems that the bidirectional Agree approach argued for in this thesis does not have. Therefore, case matching can be seen as an argument for bidirectional Agree.

6.3.3. Only Syntactic Agree

The next two alternative agreement approaches I would like to discuss are based on the assumption that there is only one Agree operation. The first possibility is to assume that Agree is entirely syntactic. In section 4.3, I have already noted one problem for such an approach, namely the syncretism effect, illustrated by the Polish example in (157).

- (157) Jan lubi_{acc} kogokolwiek_{acc/gen} Maria nienawidzi_{gen}.
Jan likes whoever Maria hates
'Jan likes whoever Maria hates.'

The argument goes as follows: Assuming that the morphological form does not count for narrow syntax, the case matching condition cannot be a principle of narrow syntax. If case matching is reduced to the matching condition of Agree, the matching condition of Agree cannot be a condition in narrow syntax, and therefore

the operation Agree cannot be a strictly syntactic.

A second problem for such a syntactic Agree approach emerges from the order of Agree application. Assuming that Agree is syntactic, all Agree operations have to apply throughout the derivation and the order is different from the order in a two-step-Agree approach. (158) shows the derivations for all patterns of free relatives and (159) shows the derivations for parasitic gaps.

(158) a. *Strict case matching*

Step I	$\alpha[c: _]$	\longrightarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
--------	-----------------	-------------------	--	--

Step II	$\emptyset[*c: _ *]$	\longrightarrow	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
---------	-----------------------	-------------------	---	--

Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longleftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\nexists (\alpha \neq v)$
----------	--	------------------	------------------------------------	----------------------------

b. *Case hierarchy effect*

Step I	$\alpha[c: _]$	\longleftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
--------	-----------------	------------------	--	--

Step II	$\emptyset[*c: _ *]$	\longrightarrow	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
---------	-----------------------	-------------------	---	--

Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longleftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[c:\{\text{dat}, \text{acc}, \text{nom}\}] (v \subseteq \emptyset)$
----------	--	------------------	------------------------------------	---

c. *Absence of case matching*

Step I	$\alpha[c: _]$	\longleftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
--------	-----------------	------------------	--	--

Step II	$\emptyset[c: _]$	\longleftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[c:\{\text{acc}, \text{nom}\}]$ (no requirement)
---------	--------------------	------------------	------------------------------------	--

(159) a. *Strict case matching*

Step I	$\emptyset[*c: _ *]$	\longleftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$
--------	-----------------------	------------------	--	--

Step II	$\alpha[c: _]$	\longrightarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{acc}, \text{nom}\}]$
---------	-----------------	-------------------	------------------------------------	--

Step III	$\emptyset[*c:\{\text{dat}, \text{acc}, \text{nom}\} *]$	\longleftarrow	$\alpha[c:\{\text{acc}, \text{nom}\}]:$	$\nexists (\alpha \neq \emptyset)$
----------	--	------------------	---	------------------------------------

b. *Absence of case matching*

Step I	$\emptyset[c: _]$	\longleftarrow	$\text{Appl}[c:\{\text{dat}, \text{acc}, \text{nom}\}]:$	$\alpha[c:\{\text{dat}, \text{acc}, \text{nom}\}]$
--------	--------------------	------------------	--	--

Step II	$\alpha[c: _]$	\longleftarrow	$v[c:\{\text{acc}, \text{nom}\}]:$	$\emptyset[c:\{\text{acc}, \text{nom}\}]$ (no requirement)
---------	-----------------	------------------	------------------------------------	--

6. Alternatives

Similar to the identity approaches and the upward Agree approach, this approach can only derive the absence of case matching with the total absence of Agree. This would predict a language where case matching should be absent both in parasitic gap constructions and free relatives. Furthermore, the difference between strict case matching and case hierarchy effects must be due to a language specific matching condition of Agree.

Because of these two issues, a strictly syntactic definition of Agree seems to be problematic for deriving the case matching effects.

6.3.4. Only Post-Syntactic Agree

A different version of a single Agree approach would be to assume that Agree is only post-syntactic. This has the consequence that movement can bleed Agree relations and with them the case matching condition if it destroys the c-command relations. As argued in section 4.3, movement does not bleed case matching. Otherwise the case matching effects for German parasitic gaps should differ between base position and extraposed position, contrary to fact.

- (160) Hans hat vertraut_{dat} [DP D_∅ t_{CP}] [CP *wen_{acc}/*wem_{dat} (auch immer)]
Hans has trusted who ever
Maria mag_{acc}.
Maria likes
‘Hans trusted whoever Maria likes.’

Assuming that c-command is not important for post-syntactic Agree does not help either because this would make a wrong prediction for parasitic gaps: The ban on in-situ licensing should not exist (but see (151)).

Finally, a strictly post-syntactic Agree operation would also make wrong predictions for the intervention effects in German parasitic gaps. The full picture is given in (161).

- (161)
- a. dass Hans der Maria_{dat} das Buch_{acc} [\emptyset_{Opacc} ohne t_{Op}
 that Hans the Maria the book without
 durchzulesen] zurückgibt
 through.to.read back.gives
 ‘that Hans returns the book to Maria without reading (it) through’
 - b. dass Hans das Buch_{acc} der Maria_{dat} [\emptyset_{Opacc} ohne t_{Op}
 that Hans the book the Maria without
 durchzulesen] zurückgibt
 through.to.read back.gives
 ‘that Hans returns the book to Maria without reading (it) through’
 - c. *wenn jemand der Anette_{dat} das Buch_{acc} [\emptyset_{Opdat} ohne t_{Op} zu
 if someone the Anette the book without to
 vertrauen] ausleiht
 trust lends
 ‘if someone lends Anette the book without trusting her’
 - d. *wenn jemand das Buch_{acc} der Anette_{dat} [\emptyset_{Opdat} ohne t_{Op} zu
 if someone the Anette the book without to
 vertrauen] ausleiht
 trust lends
 ‘if someone lends Anette the book without trusting her’

As shown in (161), the intervention effect is not linear. The generalization is that if both the indirect and the direct object are scrambled in front of the adjunct clause, the operator can only Agree with the direct object. In Heck and Himmelreich (2017), this generalization is derived by assuming that at the point in the derivation where Agree with the operator applies, the direct object intervenes between the indirect object and the operator. Thus, Agree is blocked here. Assuming that this is the correct approach, Agree has to be, at least to some degree, syntactic.

6. *Alternatives*

Concluding from the arguments brought forward in this and the last section, Arregi and Nevins's (2012) idea of splitting Agree into a syntactic Agree-Link operation and a post-syntactic Agree-Copy operation is crucial to derive the full picture of case matching effects in free relatives and parasitic gaps.

7. Conclusion of Part I

In this part, I have provided an in-depth study of case matching effects that occur with free relatives and parasitic gaps. Starting from a simplified pattern that was used to illustrate the idea of how variation in this domain can be derived, I have developed a more fine-grained typology of case matching patterns.

The first observation is that Polish and German are mirror images of each other when it comes to case matching effects with free relatives and parasitic gaps: Polish free relatives and German parasitic gaps require strict matching, while German free relatives and Polish parasitic gaps allow case mismatches.

The account developed throughout this part builds on the order of Agree operations and the bidirectionality of Agree. Both in free relatives and parasitic gap constructions, there is an overt item α and a covert item \emptyset which have to Agree in case features additionally to their normal case relations with case assigning functional heads. The main idea to derive the first observation is that if Agree between α and \emptyset is upward agreement, it applies early and will not have an effect on other case assignment relations. If it applies late, it can potentially bleed other case assignment relations, leading to case matching effects.

The more fine-grained observation is that there are three types of case matching

7. *Conclusion of Part I*

for free relatives – strict case matching, case hierarchy effects and the absence of case matching – but only two types for parasitic gaps: strict case matching and absence of case matching. Possible combinations of these types are not free, but can be reduced to five patterns, four of which could be proven to exist.

This distribution follows from the interaction of Agree operations, the decomposition of case features, and the definition of two matching conditions: one for general case assignment (the subset condition) and one specific for sharing constructions, specifically free relatives and parasitic gaps (the identity condition). The three tables in (162)–(164) summarize the derivations for each pattern.

(162) *Patterns of case matching*

FR \ PG	Strict	Hierarchy	Absence
Strict	German 2/ Polish 2	German 1	Greek
Absence	Polish 1	???	—

(163) *Patterns of Agree-Link*

PG \ FR	Str.		Hier.		Abs.	
Str.	<div>SL</div>	<div>FR</div>	<div>SL</div>	<div>FR</div>	<div>SL</div>	<div>FR</div>
Abs.	<div>SL</div>	<div>FR</div>	<div>SL</div>	<div>FR</div>	<div>SL</div>	<div>FR</div>

(164) *Patterns of Agree-Copy*

FR \ PG	Str.	Hier.	Abs.
Str.	<div>SL FR</div> <div> $V_2 \rightarrow \phi$ $V_2 \rightarrow \alpha$ $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\phi \rightarrow \alpha$ $\alpha \rightarrow \phi$ $V_1 \rightarrow \alpha$ $\phi \rightarrow \alpha$ $\alpha \rightarrow V_1$ $\phi \rightarrow V_1$ $\alpha \rightarrow \phi$ $V_1 \rightarrow \phi$ </div>	<div>SL FR</div> <div> $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\phi \rightarrow \alpha$ $\alpha \rightarrow \phi$ $\alpha \rightarrow V_1$ $\phi \rightarrow \alpha$ $\alpha \rightarrow \phi$ $\phi \rightarrow V_1$ </div>	<div>SL FR</div> <div> $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\alpha \rightarrow V_1$ $\alpha \rightarrow \phi$ $\alpha \rightarrow \phi$ $\phi \rightarrow V_1$ </div>
Abs.	<div>SL FR</div> <div> $V_2 \rightarrow \phi$ $V_2 \rightarrow \alpha$ $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\phi \rightarrow \alpha$ $\phi \rightarrow \alpha$ $V_1 \rightarrow \alpha$ $\phi \rightarrow V_1$ $\alpha \rightarrow V_1$ $V_1 \rightarrow \phi$ </div>	<div>SL FR</div> <div> $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\phi \rightarrow \alpha$ $\phi \rightarrow \alpha$ $\alpha \rightarrow V_1$ $\phi \rightarrow V_1$ </div>	<div>SL FR</div> <div> $\phi \rightarrow V_2$ $\alpha \rightarrow V_2$ $\alpha \rightarrow V_1$ $\phi \rightarrow V_1$ </div>

7. Conclusion of Part I

A major success of this approach is that the absence of case hierarchy effects in parasitic gaps could be shown to fall back on the specific syntactic structure of this construction. There is no need to impose special constraints on Agree in this construction.

Finally, I have discussed in detail how alternative approaches to case matching in sharing constructions could deal with the empirical facts, concluding that the proposed system is the least problematic one. This conclusion also means that case matching can be seen as an argument for the bidirectionality of Agree and the split of Agree in Agree-Link and Agree-Copy.

Based on these conclusions, part II of this thesis further explores the question of how the additional Agree relation in free relatives and parasitic gaps arises and why it is special. In doing so, I will also review the two constructions in more detail showing that Agree is also able to derive general properties of them.

The last part, part III, will contain a more general look at Agree. I will deal with possible consequences that follow from this very specific understanding of Agree.

Part II.

The Emergence of Agree in Free Relatives and Parasitic Gap Constructions

8. Agree in Parasitic Gap Constructions

In part I, I proposed a structure for parasitic gaps based on the assumption that the shared category agrees with a covert category. This is one possible answer to the question as to how one dislocated syntactic category can be traced back to two different base positions in two different clauses.

In this chapter, I will argue that the agreement approach is essentially correct as it is able to capture the idiosyncratic properties of parasitic gap constructions. I will further show that the empty element in these structures is not an operator but is created from the antecedent phrase throughout the derivation due to a fission-like operation. This fission is ultimately also the reason why the overt phrase and the empty element have to Agree. Under this approach, no use of concepts like interarboreal movement (Nunes (1995, 2001, 2004)) or empty operator movement (e.g. Chomsky (1986); Nissenbaum (2000)) is necessary. Theories which make use of these syntactic means are shown to make wrong empirical predictions.

In order to show that this approach to parasitic gaps makes the right predictions, I will discuss four properties of parasitic gap constructions that are, for one thing, defining of the construction and, at the same time, challenging for derivational analysis. Concretely, I will respond to the following questions:

8. Agree in Parasitic Gap Constructions

- (165)
- a. Why are parasitic gaps only selectively sensitive to islands?
 - b. Why can complex antecedents only be reconstructed into the real gap but not the parasitic gap?
 - c. Why are certain properties of the antecedent only important for its relation to the real gap but not to the parasitic gap?
 - d. Why can parasitic gaps only be licensed from certain positions?

In order to answer these questions, I will start in section 8.1 by presenting a detailed derivation of the syntactic structure of parasitic gaps. Afterwards, in section 8.2, I will go through each of the questions separately. Finally, in section 8.3, I will review alternative accounts in the light of the data discussed before.

8.1. An Analysis of Parasitic Gaps

In order to answer the general questions what a parasitic gap is, I suggest a new approach that is based on a new operation *Create-Dep* that creates covert elements out of existing overt ones. The difference from a simple copy operation is that *Create-Dep* also creates a dependency between the two items. As a result, the covert and the overt element in parasitic gap constructions are merged separately and are targeted by different syntactic operations. The only syntactic dependency between the two items comes about by Agree. Thus, the overt and the covert item share certain properties but not others.

In this section, I will show how *Create-Dep* is integrated into a standard minimalist framework. I will begin by clarifying the assumptions the analysis is based on.

Afterwards, I will define the operation and go through a detailed derivation of a parasitic gap sentence.

8.1.1. Assumptions

8.1.1.1. Syntactic structure

I assume a standard clause structure like (166) with three functional heads above the VP in a transitive clause: v , T and C. The v head assigns accusative case to the complement of V, T assigns nominative case and also bears a ϕ -probe that is responsible for subject-verb-agreement. Nominal categories are headed by a functional head D (Abney (1987)). In clauses with dative objects (166-b), an applicative head Appl is part of the structure and assigns dative case.

- (166) a. $[_{CP} C [_{TP} DP_{Subj} T [_{vP} DP_{Subj} v [_{VP} V DP_{Obj}]]]]$
 b. $[_{CP} C [_{TP} DP_{Subj} T [_{vP} DP_{Subj} v [_{ApplP} Appl [_{VP} DP_{indObj} V DP_{Obj}]]]]$

Finally, adverbial clauses that can host parasitic gaps are taken to be adjuncts to VP.

8.1.1.2. Assumptions about derivation

Following a standard Chomskyan system (Chomsky (1995, 2000, 2001, 2008)), I assume that syntax is strictly derivational and obeys the Strict Cycle Condition (Heck and Himmelreich (2017), adapted from (Chomsky 1973: 243)) in (167), and that derivations do not allow look-ahead.²⁵

²⁵For a discussion of whether derivations of parasitic gap structures require look-ahead, see section

8. Agree in Parasitic Gap Constructions

(167) *Strict Cycle Condition*

If Σ is the root of the current phrase marker, then no operation can take place exclusively within Ω , where Ω is dominated by Σ .

The structure-building operation is called Merge and is triggered by features [$\bullet F \bullet$]. The feature-checking operation is called Agree and triggered by features [$*F*$] (Sternefeld (2006); Heck and Müller (2007); Müller (2010)). Movement is understood as a special type of Merge. This means that features [$\bullet F \bullet$] can also be checked by “dislocating” subconstituents of the structure. If movement applies to check a feature [$\bullet F \bullet$], the category bearing [$\bullet F \bullet$] has to c-command the category that checks [$\bullet F \bullet$]. This means that there is only upward movement. As argued in part I, Agree is possible if a probe feature [$*F*$] c-commands a matching feature [F] or vice versa (see also Koopman (2006); Baker (2008*b*); Riedel (2009)).

Spell-out of syntactic structure is cyclic, proceeding in phases with designated phase heads v , C and D (Chomsky (2000, 2001); see Svenonius (2004), Heck and Zimmermann (2004) for D) and successive-cyclic movement is enabled by edge features [$\bullet X \bullet$] (Chomsky (2000)) which can be added to phase heads whenever the edge feature condition is met.

As for the edge feature condition, I adopt the version proposed in Müller (2010: 42), defined as in (168).

(168) *Edge Feature Condition*

An edge feature [$\bullet X \bullet$] can be assigned to the head γ of a phase only if (a) and (b) hold:

- a. γ has not yet discharged all its structure-building or probe features.
- b. [$\bullet X \bullet$] ends up on top of γ 's list of structure-building features.

Note that this definition is based on the assumption that all structure-building features on a head are organized in a stack. Adding an edge feature on top of the stack means that the edge feature has to be discharged before the final operation-triggering feature is checked. For derivations, this means that successive-cyclic movement always targets an inner specifier, never the outermost specifier. This property of successive cyclic movement is summarized in the Intermediate Step Corollary (ISC) in (169) (Heck and Himmelreich (2017), adapted from (Müller 2011: 176)).

(169) *Intermediate Step Corollary (ISC)*

Intermediate movement steps to specifiers of X (triggered by edge features) must take place before the final specifier is merged within XP.

A final point to note about derivations is that the process of argument scrambling in languages like German is taken to be edge feature movement as well (see Heck and Himmelreich (2017) for a discussion). Thus, scrambling of objects to the vP always tucks-in below the subject in Spec-vP.

8.1.2. Create-Dep

The background of the operation Create-Dep is that lexical items consist of syntactic, phonological and possibly semantic features. As for the syntactic features, two main groups can be distinguished: features that engage in argument licensing (arg features) and operator features (op features) that are not involved in the predicate-argument relation. Abstractly, a lexical item LI can be represented as a

8. Agree in Parasitic Gap Constructions

set of features, as done in (170).

- (170) $LI = \{arg_1, arg_2, arg_3, \dots, op_1, op_2, op_3, \dots, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\}$

Assuming that syntactic structures can also be represented as sets (Chomsky (1995) et seq.), it is reasonable to assume that parts of lexical items can be subject to syntactic operations in the same way as parts of syntactic structures are targeted by syntactic operations such as movement and Agree. However, given that, at some point, some constraint on lexical integrity holds, a part of a lexical item cannot be subject to operations after a certain point. Again, this assumption is reasonable since such a principle also exist for parts of syntactic structures, namely the *Phase Impenetrability Condition* (PIC), as defined in (171) (adapted from Chomsky (2001)).

- (171) *Phase Impenetrability Condition*

In phase XP with a head X, the domain of X is not accessible to operations outside of XP, but only X and its edge.

I would like to suggest that lexical items constitute a phase as soon as they are merged with another lexical item. That is, lexical integrity only holds for the syntactic operations Move and Agree that build on preceding applications of Merge. Therefore, operations that affect a part of a lexical item LI have to apply before any application of Merge that involves LI.

Now, the main challenge of deriving sharing constructions in general, and of para-

sitic gap constructions in particular, is that, on the surface, two heads are licensing one and the same argument, which should not be possible assuming a standard look on the properties of argument licensing, such as agreement, subcategorization, and case assignment. This challenge can be overcome, assuming that the head of the argument that carries the relevant argument licensing features can (temporarily) “borrow” these features, thereby creating a dependency between the affected LI and the set of borrowed features. This process is reminiscent of movement in the copy theory of movement. The resulting dependency with the set of borrowed features intuitively resembles the dependency that we find in standard Agree relations, where certain features of a lexical item also depend on the features of another lexical item. Keeping in mind the variation concerning case matching effects, the locus of the dependent features can vary between languages.

With this background in mind, the operation Create-Dep can be defined as in (172).

(172) *Create-Dep(endent)*

- a. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{*f_1*, \dots, *f_m*, f_{m+1}, \dots, f_n\}, \{f_1, \dots, f_m\}$
- b. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{f_1, \dots, f_m, f_{m+1}, f_{m+1}, \dots, f_n\}, \{*f_1*, \dots, *f_m*\}$
- c. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{*f_1*, \dots, *f_m*, f_{m+1}, \dots, f_n\}, \{*f_1*, \dots, *f_m*\}$

8. Agree in Parasitic Gap Constructions

For parasitic gap constructions (and free relative constructions), the element to share is the head of a phrase with movement-triggering features such as the *wh*-movement feature [*wh*]. The features that are needed to be shared in order to fill two argument positions are the categorial feature for Merge, e.g. [*D*], the ϕ -features for ϕ -agreement [ϕ], and the case feature for case assignment [*c*]. According to (172), Create-Dep leads to one of the three pairs of items in (173).

- (173) a. $\{D, \phi:3sg, c:_, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\} \rightarrow$
 $\{*D*, *\phi:3sg*, *c:_*, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\}, \{D, \phi:3sg, c:_\}$
- b. $\{D, \phi:3sg, c:_, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\} \rightarrow$
 $\{D, \phi:3sg, c:__, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\},$
 $\{*D*, *\phi:3sg*, *c:_*\}$
- c. $\{D, \phi:3sg, c:_, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\} \rightarrow$
 $\{*D*, *\phi:3sg*, *c:__, wh, phon_1, phon_2, phon_3, \dots, sem_1, sem_2, sem_3, \dots\},$
 $\{*D*, *\phi:3sg*, *c:_*\}$

In (173-a), Create-Dep creates an additional set of features consisting of [*D*], [ϕ], and [*c*]. The probe features that create the dependency between the two items remain on the original item. In (173-b), the same happens as in (173-a), the only difference being the location of probe features on the new item. Finally, in (173-c), both items carry probe features. What (173) also shows is that the true reason for variation lies in the operation Create-Dep. Languages can vary as to which of the three options is chosen.

Adding the operation Create-Dep to the inventory of syntactic operations, the following facts about parasitic gap constructions can be captured: First, the reason why covert categories in parasitic gap constructions are actually covert is that phonological features are not affected by the operation Create-Dep, simply because they are not relevant for an item's status as a potential argument. Second, it provides a reason for the observed intimate relation between the empty item in parasitic gaps and the antecedent of the gap: They are tied together by various Agree relations. Finally, letting Agree be a result from Create-Dep also captures the fact that the relation between the covert item and its antecedent shows crucial properties of Agree relations, such as locality, intervention, and structural requirements.

Finally, note that in order for the two items created in (173-a-c) to engage in argument licensing, the probe features themselves must be able to satisfy the licensing features on the verbal functional heads.

To sum up, the sharing in parasitic gap constructions comes about by agreement between an covert item that is the argument of the embedded adjunct clause and a overt item that is an object in the matrix clause. The special assumption about parasitic gaps in this approach is that the covert item is not part of the lexicon, but emerges throughout the syntactic derivation.²⁶

²⁶I first proposed this idea in Assmann (2010), where a version of the present theory was presented.

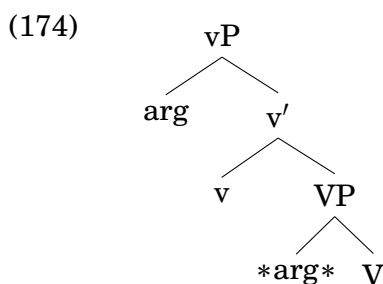
The approach outlined here differs from Assmann (2010) in two points. First, the prederivational operation that applies in parasitic gap constructions was called *duplication*. Second, *duplication* introduced a new type of feature [$\triangleright F \triangleleft$] into the grammar. This additional feature type has been dismissed and replaced by normal probe features [$*F^*$], thereby simplifying the theory. I would like to thank Fabian Heck and Gereon Müller for this suggestion. I would like to note further that this idea of creating a dependent item has also been explored in depth by Agbayani (1998). In

8. Agree in Parasitic Gap Constructions

This idea, however, also raises a number of questions as to how the application of the operation is constrained. So far, I only assumed that Create-Dep has to apply early in the derivation. But what triggers the application of the operation? Or, put differently, what happens if Create-Dep applies in a non-sharing construction?

As it turns out, Create-Dep can apply freely without a certain trigger. Thus, it might also apply in configurations other than parasitic gaps. But since Create-Dep creates an additional item that needs to be merged in the structure, it can only be successful in derivations that at some point have an open spot for this additional item, that is, a spot that is not filled by a lexical item.

This restriction, imposed by the need that all items in a structure are combined under a single root (Single Root Condition, see (Partee et al. 1990: 441f)), rules out all unnecessary applications of Create-Dep. Still, it is possible that Create-Dep applies, for example, in a simple transitive clause, so that one argument can fill both the object and the subject position. This is abstractly shown in (174).



his theory, however, the respective operation applies inside the derivation. Agbayani and Ochi (2007) discuss how this idea can be used to describe parasitic gaps. However, the operation I am proposing differs from the one of Agbayani and Ochi (2007) and also makes different predictions (cf. section 8.2).

There are two ways to deal with such cases: The structure is ruled out, because two arguments in one clause cannot Agree, presumably due to different θ -roles. (This can be encoded in a θ -feature, if necessary.) Second, such structures are allowed, which might give a new perspective on the derivation of reflexive structures (cf. Hornstein (1999)).

A second, related, question is how the features to engage in Create-Dep are chosen. Again, assuming that Create-Dep applies freely, unwanted derivations have to be filtered out. For example, operator features of the shared item in parasitic gap constructions should not be duplicated. This falls out, assuming that operator features, which trigger movement to operator positions, should be checked at some point. This assumption is not special to parasitic gaps or Create-Dep but is necessary to rule out operators in non-operator positions in general. For example, a relative pronoun with a operator feature [rel] should eventually occur in an operator position. Merging relative pronouns in derivations that do not involve a relative clause with such a position available to the relative pronoun have to be ruled out in general. Thus, this filter is not special to Create-Dep, but most likely can be reduced to general properties of movement. Finally, note that assuming that phonological features do not play a role in syntax, they are expected not to undergo Create-Dep. Thus, the covert category really has to be covert.

In the rest of this chapter, I will show how the special behavior of parasitic gaps²⁷ can be derived on the basis of Create-Dep.

²⁷See e.g. Postal (1993) about the differences between parasitic gap constructions and across-the-board movement. See also section 8.2.

8. Agree in Parasitic Gap Constructions

8.1.3. A Sample Derivation

In this section, I will integrate the operation Create-Dep introduced above into the general framework outlined in section 8.1.1. In doing so, I will go through a detailed derivation of the German example in (175).

- (175) (das Buch), das₁ ich [ohne *pg*₁ zu mögen] gekauft habe
 (the book) which I without to like bought have
 ‘(the book) which I bought without liking’

The shared item in (175) is the relative pronoun *das*. If the derivation of (175) should converge, *das* has to undergo Create-Dep. The derivation in (176) starts out with the two items *mögen* and *das* in the workspace.²⁸ This step is shown in (176) vs. (177).

- (176) WORKSPACE:

mögen	das
{V, •D•, ...}	{D, ϕ :3sg, c:–, rel, ...}

The relative pronoun *das* has all the argument licensing features listed above, and additionally, it has an operator feature [rel] for movement to Spec-CP. In (177), Create-Dep applies to *das*. Since this is the derivation of a German parasitic gap clause, the double probe option of Create-Dep (172-c) is chosen (cf. 95).

- (177) WORKSPACE: Create-Dep

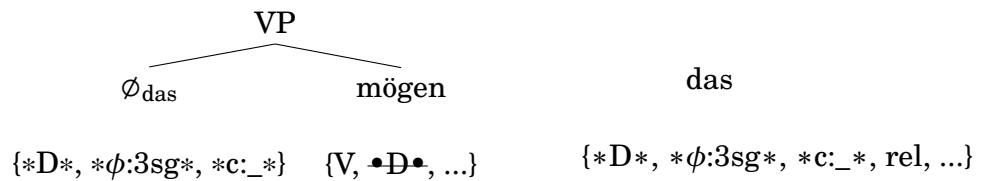
mögen	das	\emptyset_{das}
{V, •D•, ...}	{*D*, * ϕ :3sg*, *c:–*, rel, ...}	{*D*, * ϕ :3sg*, *c:–*}

²⁸I will not go into a detailed discussion of the concept of workspace. In general, if syntactic operations apply, the affected items have to be available for the operations. This availability is enabled by putting all items in a workspace. I will also not deal with the question of how lexical items are moved from the lexicon into the workspace. This is a question that every derivation in this Chomskyan system faces and is thus orthogonal to the question of how to deal with parasitic gaps.

In (177), the D feature, the ϕ -features and the case feature are copied and constitute a new item ϕ_{das} . The probe properties of the features follow from the parameters set for German. Importantly, the feature [rel] is not copied. (See section 8.1.2.)

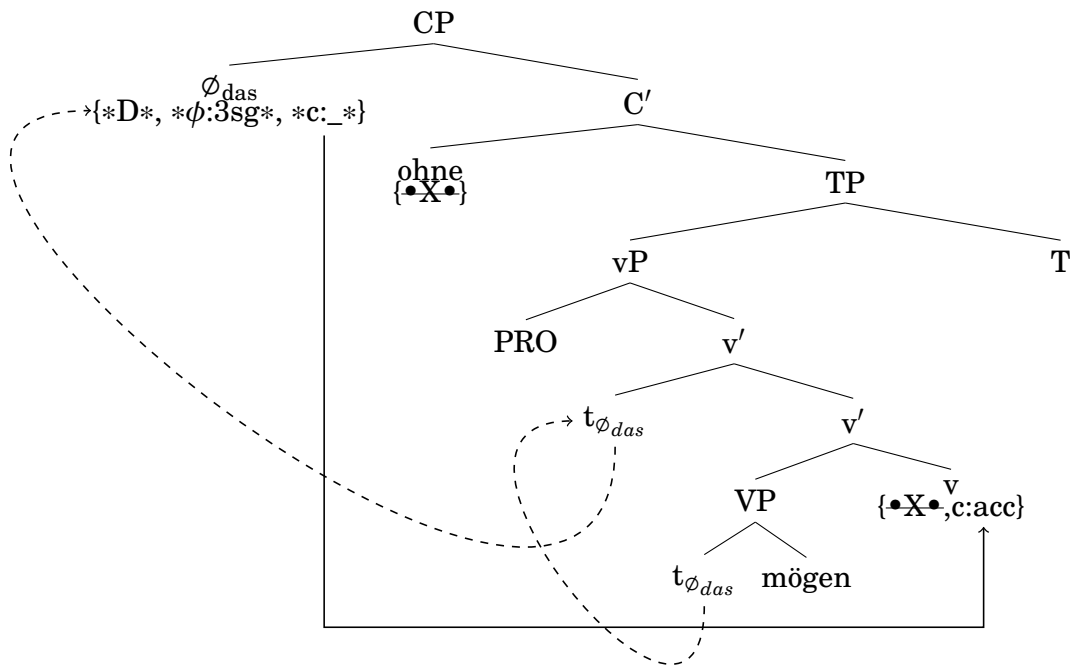
In the next step, the verb *mögen* and the newly created ϕ_{das} are merged. This step checks the selectional feature [$\bullet D \bullet$] of the verb. The relative pronoun *das* remains in the workspace until it can be merged in its target argument position.

(178) WORKSPACE: Merge



After this first application of Merge, several steps of Merge and Agree operations apply in order to build the embedded *ohne*-clause. These steps are summarized in the structure in (179).

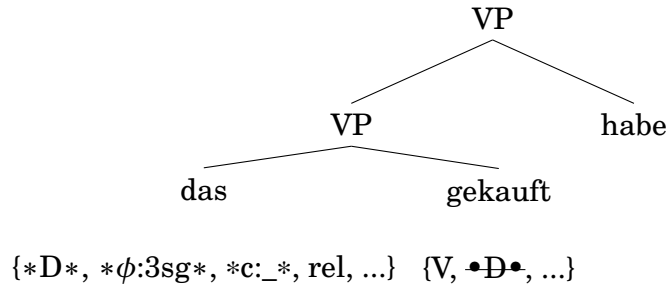
(179) WORKSPACE: Merge*/Agree*



8. Agree in Parasitic Gap Constructions

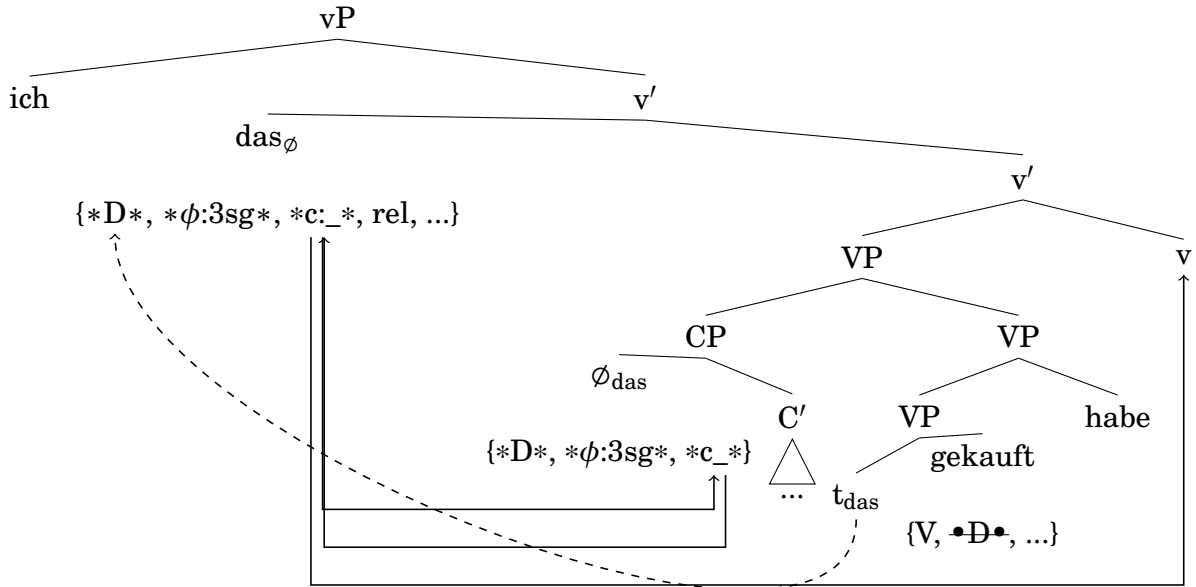
As soon as v merges with VP, ϕ_{das} and v establish an Agree-Link for case (and ϕ -features). In (180), ϕ_{das} scrambles to Spec-vP, under the subject PRO. For the sake of concreteness, I assume that infinitival adjunct clauses contain a PRO that is controlled by the matrix subject. PRO receives nominative case from T (not indicated in (179)). Finally, ϕ_{das} is moved to Spec-CP due to edge feature movement.²⁹ Next, the matrix VP is built, as shown in (180).

(180) WORKSPACE: Merge*



The final relevant part of the derivation of the example in (175) is shown in (181).

(181) WORKSPACE: Merge*/Agree*



²⁹I assume that scrambling to Spec-CP is an option in these adjunct clauses. The question about the restrictions on edge feature movement is an independent issue and not specific for parasitic gap constructions. Here, it suffices to assume that C in *ohne*-clauses allows movement to its specifier.

First, the *ohne*-clause adjoins to the VP. Next, the matrix *v* is merged and Agrees with *das* in case. Finally, *das* scrambles to Spec-vP. In this position, c-command between *das* and ϕ_{das} is given and the licensing of the parasitic gap applies via Agree between the two items. This establishes the final two Agree-Links relevant for the case matching derivations in section 5.2.1.³⁰

The derivation sketched above raises two further questions. First, what happens if the base-merge positions of *das* and ϕ_{das} are switched? Such a derivation would crash, because the relative pronoun has to be moved to the edge of the matrix clause. If it is not, the matrix C could not check its feature [\bullet rel \bullet]. If the relative pronoun were to be merged in the adjunct clause, it could not move out of the adjunct island into the matrix clause. One way to derive this concrete island effect, would be to assume that VP in German is a phase (cf. Müller (2010, 2011)) and that phase edges are not defined recursively. The reasoning is as follows: The adjunct clause is the last phrase to be merged in VP. Thus, there can be no successive-cyclic movement into Spec-VP. Since edges are not recursive, the element in the embedded Spec-CP position is not available for movement into the next phase, the matrix vP.

The next question is why *das* and ϕ_{das} have to Agree with each other. First, following the logic of the argument below (174), each of them cannot Agree with another DP in their respective clauses. But since parasitic gaps can also occur with ditransitive constructions, it might be possible that, for example, ϕ_{das} Agrees with the indirect instead of the direct object. Not filtering out such a case per se by an

³⁰Note that the derivation in (181) continues after this step. The matrix T and C are merged, and the relative pronoun is moved to Spec-CP. However, these steps do not influence the licensing of parasitic gaps.

8. Agree in Parasitic Gap Constructions

additional constraint, intervention effects are predicted to occur, which is the case in German (see section 6.3.4 for a detailed discussion).

If one wanted to rule out these cases, the following assumption could be added to the approach: Features encoding referential indices exist and are affected by Create-Dep as well. For example, following Browning (1987); Reuland (2001); Řezáč (2004a), the ϕ -features of a nominal category are linked to its referential index. Under this assumption, the referential index together with the ϕ -features would be present on both items, and thus, Agree can only be successful, if the two indices match. This assumption would ensure that there is only one possibility for ϕ_{das} to Agree.

Having shown, how parasitic gaps are derived in general, I will turn to properties of parasitic gaps and their derivation in this approach in the next section.

8.2. Deriving the Behavior of Parasitic Gaps

8.2.1. Question I: Why are Parasitic Gaps ‘Selectively’ Sensitive to Islands?

Parasitic gaps show a paradoxical property in that the antecedent of a parasitic gap can be separated from the gap by one island but not by more than one island (Kayne (1983, 1984); Bennis and Hoekstra (1985); Longobardi (1984); Koster (1986); Chomsky (1986); Manzini (1994); Nunes (1995, 2001, 2004)) This shown in (182) for German.³¹

³¹This property of parasitic gap constructions can be observed in other languages as well, most notably in English.

8.2. Deriving the Behavior of Parasitic Gaps

- (182) a. Welchen Artikel hast du [_{isl} ohne *pg* zu lesen] *t* abgeheftet?
 which article have you without to read filed
 ‘Which article did you file without reading?’
- b. *Welchen Artikel hast du [_{isl} ohne den Autor zu kennen [_{isl} der
 which article have you without the author to know who
pg geschrieben hat]] *t* abgeheftet?
 written has filed
 ‘Which article did you file without knowing the author who wrote?’

In (182-a), *welchen Artikel* (“which article”) is separated from the parasitic gap by one adjunct island while in (182-b) there are two islands between them. The difference in grammaticality falls out from the approach above: In order for a phrase to license a parasitic gap, Agree has to apply. In (182-b) ϕ_w and *welchen Artikel* are separated by a finite clause boundary. This is shown in (183). Assuming that Agree obeys absolute locality restrictions, such as clause boundaries or island constraints, probe and goal are too far away from each other in (182-b) to enter into an Agree relation.

- (183) [_{VP} du [_{DP} welchen Artikel] [*c*] [_{CP} ohne den Autor zu kennen [_{CP} ϕ_w [*c*]
 der t_{ϕ} geschrieben hat]] t_{DP} abgeheftet?

(183) shows the derivation at the point when *welchen Artikel* tries to Agree with ϕ_w . At this point, the two categories are separated by two CPs, which makes Agree impossible. In the derivation of (182-a) (cf. section 8.1.3), probe and goal are only separated by one CP boundary.³²

-
- (i) a. What did you read *t* [_{isl} before buying *pg*]?
 b. *What did you read *t* [_{isl} after expecting me to call the editor [_{isl} before buying *pg*]]?

³² Note that this only sketches the rough idea as to how island effects are derived in this approach.

The solution is the same as in empty operator approaches such as Chomsky (1986). Island effects are influenced by language specific constraints as well, which makes it difficult to propose a

8. Agree in Parasitic Gap Constructions

8.2.2. Question II: Why Can Complex Antecedents Only be Reconstructed

Into the Real Gap But Not the Parasitic Gap?

Binding data as in (184) suggest that the position of the parasitic gap cannot host a complex DP in contrast to the real gap.

- (184) a. [Which books about himself₁]₂ did John₁ file t_2 before Mary read pg_2 ?
b. *[Which books about herself₁]₂ did John file t_2 before Mary₁ read pg_2 ?

Kearney (1983)

In (184), the complex wh-phrase contains an anaphor which must be bound by an antecedent that agrees with the anaphor in gender. Having two potential base positions in two different clauses, the anaphor inside the wh-phrase has two potential antecedents, namely the subjects of the respective clauses, of which, however, only one agrees with the anaphor in gender. In this context, the following asymmetry can be noticed: in (184-a), binding of the anaphor *himself* is possible but in (184-b), binding of the anaphor *herself* fails.

The present approach offers a principled explanation for this binding asymmetry.

condition about locality that holds for all languages. For example, an structure of the type (182-b) would be grammatical in Korean (see (i)), which indicates that Korean has a different locality restriction (Lee 1988: 194).

- (i) [[ney-ka pg_1 mal-ul kelepoki ceney] ne-eykey cenwhahaci anhko] t_1 intebyuw hakiro
I-NOM speak to before you-DAT telephone not to interview
kyelcenghan salam₁
decide man
[lit.] ‘the man who I decided to interview without calling you before I spoke to’

In (184), Create-Dep would apply to the D head of the complex wh-phrase *which*. This creates the following two items.³³

- (185) *which* [D, ϕ , c:_, wh, •N•, SEM, PHON, ...]
- a. *which* [*D*, * ϕ :3sg*, *c:_*, wh, •N•, SEM, PHON, ...]
- b. ϕ_{which} [*D*, * ϕ :3sg*, *c:_*]

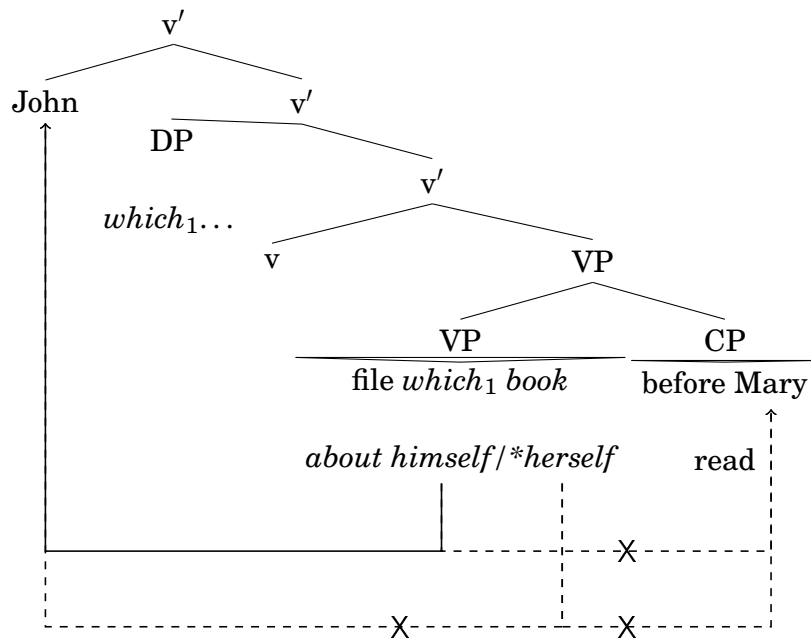
Note that only one part, namely *which*, has the feature [•N•] which enables it to merge with the NP *books about him / herself*.³⁴ The resulting DP *which books about him / herself* is merged in the matrix clause where the anaphor can be bound by the subject *John*. But the subject of the adjunct clause *Mary* cannot bind the anaphor since it does not c-command it from its position inside the adjunct clause. Thus, the only antecedent available for the anaphor is the matrix subject but not the embedded subject. The configuration is shown in (186).

³³Even though case differences are often invisible in English, English seems to be of the type German 2. Thus, it shows strict case matching (modulo syncretism effects) for parasitic gaps and free relatives (cf. Levine et al. (2001)). This is because both items, which are created by Create-Dep, bear probe features.

³⁴Applying Create-Dep to [•N•] is in principle possible. However, this would demand that at least the noun *book* undergoes Create-Dep as well. This would create Agree features which could not be checked later since both *book* and ϕ_{book} never get into a c-command configuration.

8. Agree in Parasitic Gap Constructions

(186)



The tree in (186) shows that the anaphor in the matrix clause can under no circumstances be bound by the subject of the adjunct clause. However, in case of gender agreement, it can be bound by the matrix subject. Since the matrix subject *John* is masculine, only the masculine anaphor *himself*, but not the feminine anaphor *herself*, can be bound. Hence, the anaphor *herself* is never bound and the sentence in (184-b) is expected to be ungrammatical.

8.2.3. Question III: Why are Certain Properties of the Antecedent Only

Important For Its Relation to the Real Gap But Not to the Parasitic Gap?

With respect to certain properties, the real and the parasitic gap have to behave alike. One example of this comes from categorial restrictions about parasitic gaps. Standardly, it is assumed that parasitic gaps can only be licensed by DPs (Cinque (1990)). (187), where the parasitic gap licenser is not a DP, is a standard example

that this generalization holds. However, examples as (188) can be found that show that, for example PPs can be antecedents of parasitic gaps.

- (187) *How did Deborah cook the pork *t* after cooking the chicken *pg*?

(Postal 1993: 736)

- (188) [PP Till himlen] är det inte säkert att [NP alla [S' som längtar [PP
To heaven it is not certain that everyone who longs
pg/dit]] kommer [PP *t*].
--/there get
'It is not certain that everyone who longs to (go to) heaven gets to go there.'

(Engdahl 1983: 17)

In any case, the categorial requirements of the real gap and the parasitic gap have to be identical. This follows in the present approach, since both the covert element emerges from the antecedent and has to bear the same categorial feature.

With respect to other properties, asymmetries between the real and the parasitic gap can be observed. Such an asymmetry between the real and the parasitic gap shows up, for example, in contexts where gap and antecedent are separated by weak islands such as *wh*-islands.³⁵ In German, for example, *wh*-movement is known to be sensitive to *wh*-islands, in contrast to other types of movement such as (argument) topicalization (Fanselow (1987); Müller and Sternefeld (1993)).

- (189) a. *Welche Radios weißt du nicht [CP wie man *t* repariert]?
which radios know you not how one repairs
'Which radios don't you know how to repair?'

³⁵I would like to thank Gereon Müller for bringing these data to my attention.

8. Agree in Parasitic Gap Constructions

- b. ?Radios weiß ich nicht [_{CP} wie man *t* repariert].
 radios know I not how one repairs
 ‘As for radios, I don’t know how to repair them.’

In (189), the wh-phrase *wie* creates a weak island from which only non-wh-phrases can be extracted. Thus, the wh-phrase *welche Radios* cannot move out of the embedded clause, but the topicalized non-wh-phrase *Radios* can.

If wh-islands are combined with parasitic gap constructions, we can observe that sentences are strongly deviant if the wh-island intervenes between the antecedent and the real gap (190-a), but not if it intervenes between the antecedent and the parasitic gap (190-b).³⁶

- (190) a. *Welche Radios weißt du [_{CP} wie man [ohne *pg* zu reparieren] *t*
 which radios know you how one without to repair
 verkauft]?
 sells
 lit.: ‘Which radios do you know how to sell without repairing?’
- b. ?Welche Radios hast du [ohne zu wissen [_{CP} wie man *pg*
 which radios have you without to know how one
 repariert]] *t* verkauft?
 repairs sold
 lit.: ‘Which radios did you sell without knowing how to repair?’

Thus, it seems that the wh-property of the antecedent only matters for its relation to the real but not the parasitic gap.

Such asymmetries concerning operator features simply follow from the fact that Create-Dep does not duplicate operator features in parasitic gap constructions. The

³⁶Note that the grammaticality judgments here are independent of the presence or absence of the *ohne*-clause including the parasitic gap.

element \emptyset in the derivation of (190) does not have a wh-position to move to, so it cannot bear a wh-feature. And since it doesn't bear a wh-feature, it is not sensitive to weak islands such as wh-islands in German. Hence, we predict that sentences such as (190-b) should be possible since there is no wh-phrase to cross a wh-island. On the other hand, sentences such as (190-a) must be ungrammatical because the wh-phrase cannot move out of the wh-island.³⁷

³⁷There is a second type of this asymmetry in German: It involves scrambling. Scrambling in German cannot cross a finite clause boundary. However, \emptyset licensed by a scrambled phrase can move across a finite clause boundary, as shown in (i-b).

- (i) a. *dass ich das Radio glaube [_{CP} dass man t_{dR} [\emptyset_{das} ohne t_{\emptyset} zu reparieren] t
 that I the radio believe that one without to repair
 verkaufen kann]
 sell can
 'that I believe that one can sell the radio without repairing it'
- b. ?dass ich das Radio [\emptyset_{das} ohne zu sagen [_{CP} dass man noch t_{\emptyset} reparieren muss]] t
 that I the radio without to say that one still repair must
 verkauft habe
 sold have
 'that I have sold the radio without saying that one still has to repair it'

This is unexpected if scrambling is simply edge feature movement because edge features are checked by categorial features and these features are affected by Create-Dep. So the features responsible for scrambling can be found on both \emptyset and the antecedent phrase *das Radio* in (i). The issue can be solved if one adopts a special scrambling feature afterall. This assumption, however, might cause a problem for the analysis in Heck and Himmelreich (2017), where scrambling and successive-cyclic movement are assumed to be the same process in order to derive intervention effects with floating quantifiers in German. Independent of how this conflict can eventually be resolved, the data are not a problem for the account of parasitic gaps, but rather a problem for the specific account of scrambling.

8. Agree in Parasitic Gap Constructions

8.2.4. Question IV: Why Can Parasitic Gaps only be Licensed from Certain Positions?

The last question subsumes three different properties of parasitic gaps: First, the antecedent of parasitic gap has to c-command the gap, which is known in the literature as ban on in-situ-licensing (Engdahl (1983)). Second, parasitic gaps cannot be licensed by a phrase which c-commands the parasitic gap from its base position. This is known as the anti-c-command condition Safir (1987) of parasitic gap and is mostly exemplified by cases where a subject tries to license a parasitic gap. Finally, it is known that parasitic gaps can be licensed by movement types such as wh-movement, relativization, topicalization or scrambling, which are all \bar{A} -movement types, but not by movement types such as passive movement or raising, which are A-movement types (Engdahl 1983: 13).

- (191) a. *John was killed t by a tree falling on pg .
b. *Mary seemed t to disapprove of John's talking to pg .

Interestingly, parasitic gaps cannot be licensed by a passive subject, even if it moves further to an \bar{A} -position.

- (192) *Which house was sold t [before we could demolish pg]?
based on Legate (2003: 511)

The ban on in-situ licensing follows without further ado: the antecedent has to c-command the parasitic gap in order for Agree to apply. C-command only arises

throughout the derivation if the antecedent moves.

The anti-c-command condition and the ban on A-movement are both instances of a more general observation:

- (193) After parasitic gap licensing, the antecedent cannot engage in further argument licensing relations.

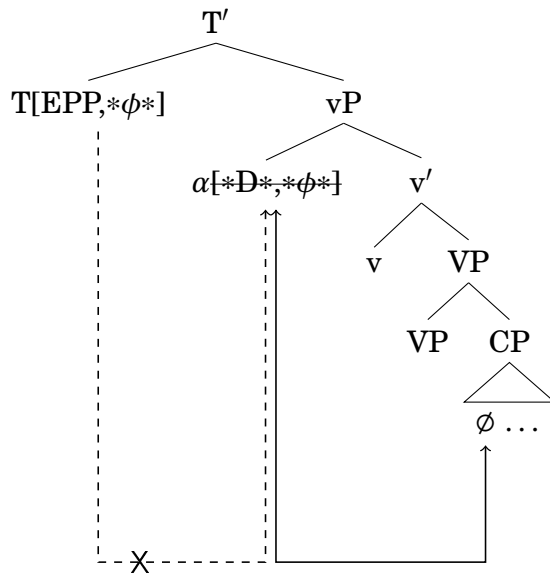
This observation is true for anti-c-command cases, where a subject licenses a parasitic gap in the vP and then enters into ϕ -agreement, and possibly EPP-movement, in the TP domain. The observation is also true for A-movement cases, where the licenser is moved into another argument position. It seems that after licensing the parasitic gap, the antecedent loses its argument properties.

The present approach offers an interesting solution to these puzzling data. Argument properties are due to categorial, ϕ and case features. Licensing of parasitic gaps is Agree in categorial, ϕ and case features. The fact that case features need to be valued throughout the derivation makes it necessary for case features to be able to play a role both in argument licensing and in parasitic gap licensing (see part I). Thus, this property of parasitic gaps cannot be reduced to case. However, both categorial and ϕ -features do not need to be valued on arguments. Thus, both can be used to derive the observation in (193): Concretely, I would like to suggest that categorial and ϕ -features are no longer available as goals for Merge and Agree after they have acted as probes themselves.³⁸ The idea is illustrated with the configuration in (194).

³⁸Note that this implies that symmetric Agree-Link between α and ϕ has to apply in parallel.

8. Agree in Parasitic Gap Constructions

(194)



After the subject α in (194) has entered into Agree with \emptyset in the vP phase, its D and ϕ features are no longer available as goals for the features on T. In other words, the derivation in (194) crashes due to a timing problem. Thus, we expect that if argument licensing of subjects happens before parasitic gap licensing, subjects should be able to license parasitic gaps. In fact, this prediction is confirmed by data as in (195-a).

- (195) a. a note which [unless we send back pg] t will ruin our relationship
 b. *a note which t will ruin our relationship [unless we send back pg]

Haegeman (1984)

Assuming that the adjunct clause in (195-a) precedes the base position of the subject because it is merged higher in the tree, e.g., as an adjunct to the TP, it follows that the relative pronoun *which* moves to the subject position Spec-TP before the adjunct clause is even merged and thus before *which* licenses the parasitic gap. In (195-b),

however, we have the same configuration as in (194), with the result that *which* cannot license the parasitic gap in this case.

In sum, the present agreement approach to parasitic gaps based on the operation Create-Dep provides solutions to the four empirical questions raised above without invoking to many additional stipulations.

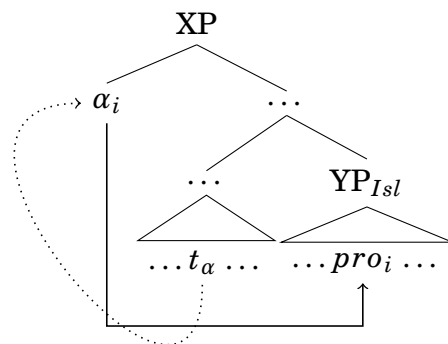
8.3. Alternative Approaches

So far, I have introduced a new approach to parasitic gaps and I have shown how this approach deals with important questions concerning parasitic gaps and their properties. I will finish this chapter by comparing the present approach to alternative approaches to parasitic gaps with respect to the four questions discussed above.

8.3.1. Binding

The binding approach is based on the assumption that parasitic gaps are empty pronouns which are bound by the antecedent of the real gap (Chomsky (1982); Engdahl (1983, 1985); Cinque (1990); Postal (1993, 1998); Ouhalla (2001); Munn (2001)). The configuration which shows this type of analysis is given in (196).

(196)



8. Agree in Parasitic Gap Constructions

The binding approach is able to explain the reconstruction asymmetries (section 8.2.2) as well as the weak island asymmetries (section 8.2.3): There is only one movement dependency between the antecedent and the real gap. Hence, reconstruction can only be possible into the real gap and weak islands can only be crossed by the antecedent moving from the real gap position to its final position.

Nevertheless, the binding approach cannot offer a principled solution to the other two puzzles discussed in section 8.2. First, there is no reason why an antecedent could not be able to bind a pronoun from an A-position, i.e., we expect that A-movement should license parasitic gaps, contrary to fact. Even if such configurational issues could be solved by stipulating that binding can only take place from an \bar{A} -position (e.g. Ouhalla (2001)), data as in (197), repeated from (192), where the passivized subject moves further to an \bar{A} -position cannot be captured.

(197) *Which house was sold *t* [before we could demolish *pg*]?

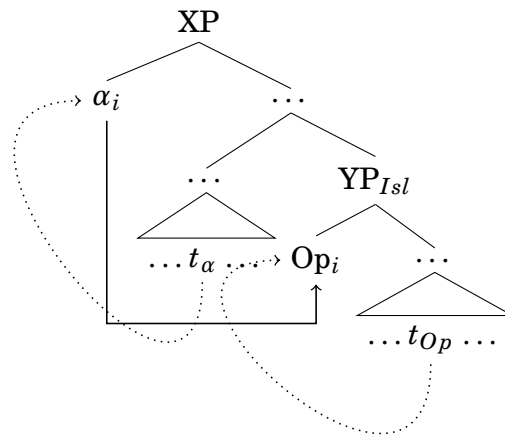
Furthermore, the binding approach cannot solve the island paradox with parasitic gaps (cf. section 8.2.1) because binding is insensitive to islands in general (Chomsky (1977); Zaenen et al. (1981); Reuland (2001)). For these reasons, the binding approach can be considered empirically inferior to the fission approach.

8.3.2. Operator Movement

Another type of approach (see e.g. Contreras (1984); Kiss (1985); Chomsky (1986); Lee (1998); Nissenbaum (2000)) assumes that an empty operator is merged in the position of the parasitic gap which moves to a higher position where it is identified

with the antecedent of the real gap; see (198). This approach structurally resembles the present approach to parasitic gaps, but does not say anything about the nature of the covert item. In this sense, the present approach can be seen as an advancement of the empty operator approach.

(198)



Since this analysis involves actual movement of the operator from the parasitic gap position to a higher position, the selective island sensitivity of parasitic gaps can be explained in the same way as in the present approach: The highest island does not intervene between the antecedent and the parasitic gap since there is no movement out of the island. Rather, the operator lands in the left periphery of this island. Nevertheless, all other islands would have to be crossed by movement.

Furthermore, the asymmetries between the real and the parasitic gap may in principle be captured since both gaps have different antecedents. For example, if the empty operator does not bear a feature [wh], it is not sensitive to weak islands, predicting that wh-islands should not fatally intervene between the antecedent of the real gap and the parasitic gap. And since the position of the parasitic gap is occupied

8. Agree in Parasitic Gap Constructions

by an empty operator, the antecedent of the real gap cannot be reconstructed into the parasitic gap, which provides an explanation for the reconstruction asymmetries in section 8.2.2. Note however, that there is no general filter to rule out a feature [wh] on the empty operator.

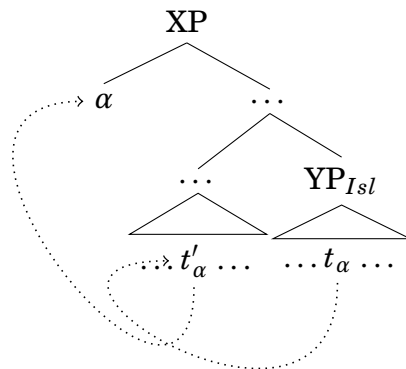
As for the answer to the final data question, it depends on the way licensing of the empty operator happens: In principle, the approach could involve additional configurational conditions that prohibit empty operators from being identified with categories in A-positions, for example by assuming a constraint on the timing of operator licensing and argument licensing (see section 13.3).

The empty operator approach, however, raises the question why the operator has to be covert. Unlike in the present approach, there is no general reason for this.

8.3.3. Sideward Movement

Finally, the sideward movement approach (Nunes (1995, 2001, 2004)) suggests that the antecedent of a parasitic gap construction is base-generated in the position of the parasitic gap and then moves interarborally to the position of the real gap, from where it moves on to its final position.

(199)



Within the framework Nunes (2001) assumes, the sideward movement approach can capture the selective island sensitivity by imposing conditions on the timing of operations in the derivation such that sideward movement itself can only apply in very restricted contexts: it may cross exactly one island but never more than one island. The fact that subjects cannot license parasitic gaps falls out from invoking additional stipulations about the licensing of traces by heads which define intervention effects whenever a subject licenses a parasitic gap.

The major problem for the sideward movement approach, however, concerns asymmetries between the real and the parasitic gap. Since the sideward movement approach claims that there is no difference between these two gaps, no asymmetries of any kind should exist. In particular, as the antecedent with all its properties and possible complexity is generated in both gaps, the antecedent should (i) be able to reconstruct in both positions to the same extent³⁹ and (ii) be sensitive to weak islands no matter where they occur. Hence, the data discussed in section 8.2.2 and 8.2.3 strongly argue against the sideward movement approach.

In conclusion, we have seen that the four properties of parasitic gaps which are

³⁹Nunes (2001) discusses data from Munn (1994) which show that reconstruction into the parasitic gap is possible when the category containing the parasitic gap precedes the real gap, see (i). Thus, he claims that reconstruction asymmetries do not pose a problem for his account. Note, however, that the sideward movement approach predicts that there should be *no* reconstruction asymmetries at all between the two gaps, contrary to fact.

- (i) a. *Which picture of herself did every boy who saw *pg* say Mary liked *t*?
- b. Which picture of himself did every boy who saw *pg* say Mary liked *t*? Munn (1994: 407)

8. *Agree in Parasitic Gap Constructions*

directly predicted by the present approach are potentially problematic for other analyses of parasitic gaps. For each of the three types of accounts discussed here, there is at least one property which cannot be explained. Thus, on empirical grounds, the present approach is superior to other theories of parasitic gaps.

8.4. Interim Summary

The aim of this chapter was to introduce an operation Create-Dep that is responsible for the emergence of Agree in parasitic gap constructions: It creates the covert item in parasitic gap constructions and it introduces probe features which trigger Agree between the covert and the overt item in this construction. The approach based on this operation is able to provide answers to the major questions concerning parasitic gaps. I finished this chapter by evaluating competing approaches and have shown that the introduction of a new syntactic operation can derive a bundle of different data.

9. Agree in Free Relatives

In this chapter I will show how the operation Create-Dep can also be used to derive free relatives. Free relatives, just as parasitic gaps belong to the class of sharing constructions, the shared item being the wh-phrase. The aim of this chapter is to show that this operation is also capable of deriving a number of puzzling properties of free relatives, which result from the ambivalent status of the wh-phrase. The chapter is structured as follows: In section 9.1, I will go through a derivation of a free relative clause in detail. Afterwards, in section 9.2, I will review some interesting observations about free relatives and show how they follow under the present approach. Finally, in section 9.3, I will compare the approach to previous analyses of free relatives.

9.1. An Analysis of Free Relatives

The aim of this section is to show that the derivation of a free relative is very similar to the derivation of a parasitic gap. In both derivations, the missing element is created by application of Create-Dep. In both derivations, the two elements enter the derivation independently of each other, and in both derivations, the dependency is due to Agree. Superficially, the constructions differ because of the position of the two

9. Agree in Free Relatives

elements. I will start by going through a derivation of a free relative. Afterwards, I will add a few remarks on the semantics of free relatives within the present approach.

9.1.1. The Syntax of Free Relatives

In the following, I go through the derivation of the German free relative in (200).

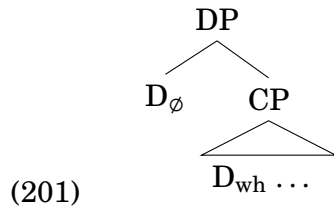
- (200) dass alle [_{FR} was ich tue] mögen
 that everyone what I do like
 ‘that everyone likes what I do’

The resulting structure of the derivation will be very similar to the analysis first proposed by Groos and Riemsdijk (1981). (Other versions of this analysis can be found in Grosu (1996, 2003); Citko (2004).) In this type of analysis, free relatives are assumed to have a structure as in (201), where a CP is adjoined to a covert D head. This assumption guarantees that the entire category is a DP which can be merged in a DP position.⁴⁰

⁴⁰Note that there are also free relatives of other categories, e.g., adjectival free relatives (cf. Grosu (1996, 2003)). A German example is given in (i).

- (i) Ich zahle [_{FR} wie viel du bezahlst hast].
 I pay how much you have paid.

In the present account, the external head of the free relative results from Create-Dep applying to the wh-phrase. In this step, the categorial feature of the wh-phrase is also copied. Thus, the external head of an adjectival free relative is expected to have a feature [A], just like the wh-phrase in (i) (compare section 8.2.3).



The main question here concerns the covert D head. In what follow, I assume that the covert head emerges from the wh-phrase. The feature structure of the wh-pronoun *was* in (201) is given in (202).

(202) $\text{was} = \{\text{D}, \phi:3\text{sg}, \text{c:}_-, \text{wh}, \dots\}$

Given the structure in (202), there are various possibilities as to how Create-Dep can apply to *was*. There is, however, only one possibility that will lead to a converging derivation: Creating a new item out of the argument features.

Similar to the derivation of parasitic gaps, the first operation to apply is Create-Dep. This is illustrated in (204).

(203) WORKSPACE:

tue	was
$\{\text{V}, \bullet\text{D}\bullet, \dots\}$	$\{\text{D}, \phi:3\text{sg}, \text{c:}_-, \text{wh}, \dots\}$

(204) WORKSPACE: Create-Dep

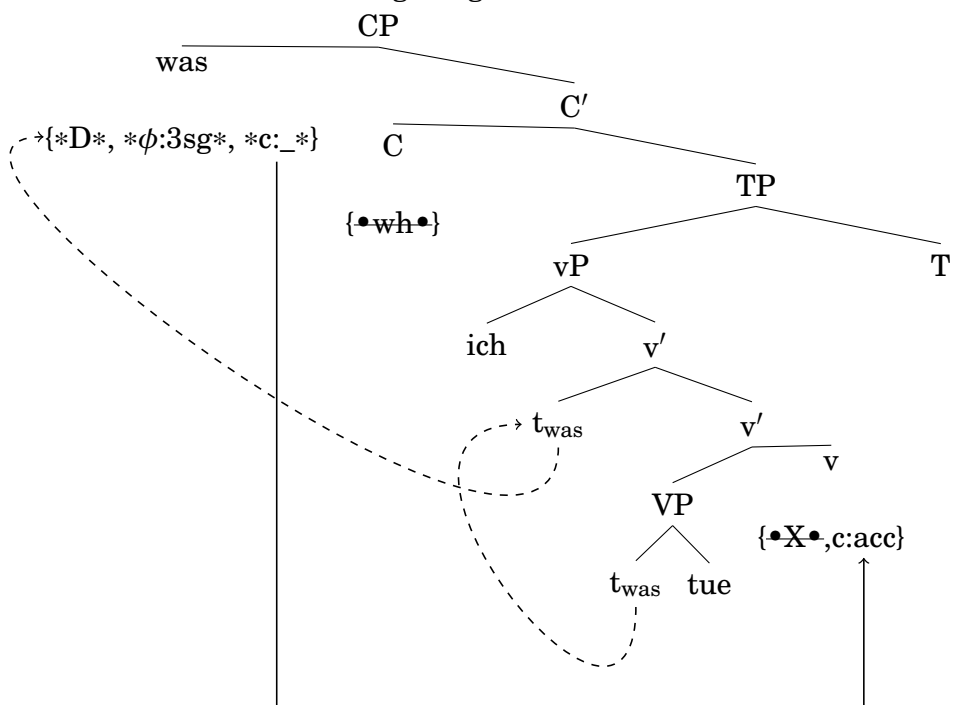
tue	was	\emptyset_{was}
$\{\text{V}, \bullet\text{D}\bullet, \dots\}$	$\{\ast\text{D}\ast, \ast\phi:3\text{sg}\ast, \ast\text{c:}_-\ast, \text{wh}, \dots\}$	$\{\ast\text{D}\ast, \ast\phi:3\text{sg}\ast, \ast\text{c:}_-\ast\}$

Again, deriving a German example, the double probe option of Create-Dep (172-c) is

9. Agree in Free Relatives

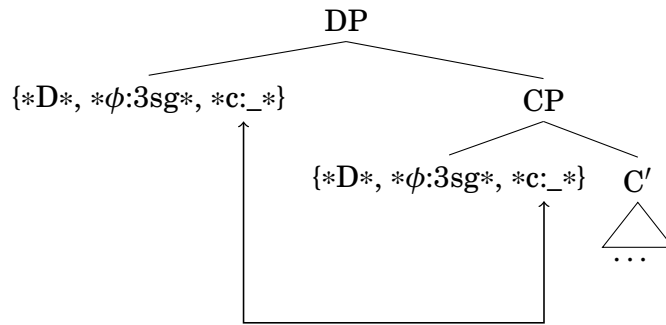
chosen (cf. (95)). When the syntactic derivation starts, the embedded clause is built first. The pronoun *was* that contains the *wh*-feature must be merged in this clause in order for C to check its *wh*-feature. In its base position, *was* receives accusative case from the embedded *v*. The *wh*-pronoun moves to Spec-CP through Spec-vP, just as the relative pronoun *das* did in the derivation in section 8.1.3.

(205) WORKSPACE: Merge*/Agree*



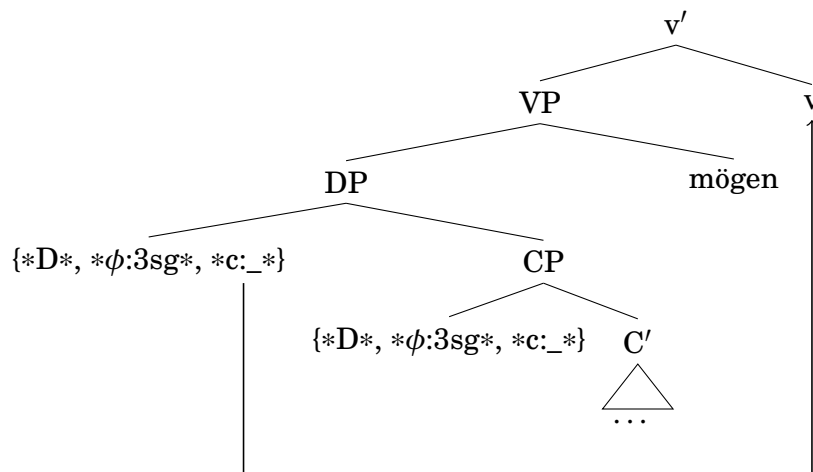
The element ϕ_{was} can now be merged with this CP, obtaining the structure in (201), see (206). Note that the D head does not contain any phonological features and is thus covert.

(206) WORKSPACE: Merge*/Agree*



In this configuration, a symmetric Agree-Link between the D head and *was* can be established. Afterwards, the entire DP is merged as the object of the matrix clause and the covert D head Agrees in case with the matrix v, see (207).

(207) WORKSPACE: Merge*/Agree*



To sum up, Create-Dep creates the covert D head out of the wh-phrase. Agree between the D head and the wh-phrase is established at the point when the DP is built.

This approach to free relatives can provide answers to three main questions raised by this construction: First, the question why a clause can occur in a position reserved for non-clausal arguments is answered by assuming that the clause is the

9. Agree in Free Relatives

complement of a covert D head. The entire structure is a DP and can therefore occur in DP positions.

Second, the head of the FR must be covert because it cannot contain any phonological features due to Create-Dep affecting only syntactic features.

Third, the close link between the wh-phrase and the covert head arises from the fact, that the covert head is made up of features from the wh-head and that the two have enter into Agree.

Before I proceed with the analysis of some of the properties of free relatives, one question remains to be answered: why must Create-Dep affect the wh-phrase? The reasons are the following: First, since it is a D item that is actually missing, involving a non-D category would not bring the required result. Furthermore, choosing some D item that would later be part of the matrix clause is not an option since the D head would then have to agree with another argument in the same clause (see the discussion about (174)). Now, only two potential candidates are left: *was* and the subject pronoun *ich*.⁴¹ The candidate *ich* can be excluded because *was* would eventually intervene for Agree between *ich* and the D head. If further locality restrictions are imposed on Agree, Agree between D and *ich* might also fail because the two are not close enough.

Having excluded all other options, the only possible candidate for a converging derivation is the wh-item that moves to Spec-CP of the embedded clause for independent reasons and is thus close enough for Agree to apply successfully.

In the next part of this section, I will show that the syntactic structure created so

⁴¹For the sake of concreteness, I assume that pronouns are simple D heads.

far is compatible with a standard semantics for free relatives.

9.1.2. Semantics of Free Relatives

The syntactic structure of FRs derived above is compatible with standard semantic analyses of free relatives. Following Caponigro (2003); Jacobson (1995) (cf. also Grosu and Landman (1998); Grosu (2003)), I assume that free relatives are semantically like DPs in that both denote the maximal entity (Link (1983)) described by a predicate. The maximal entity of a predicate is an entity that is composed of all single entities that fulfill the property P.

Furthermore, I assume that the semantic type of a lexical item depends on its features: wh-phrases, e.g., have a wh-feature and are therefore of type $\langle\langle e,t \rangle, \langle e,t \rangle\rangle$ (Caponigro (2003)). At this point, the question arises as to what extent the semantics of a lexical item depends on its semantic features and what semantic features actually are. Following Chomsky (1995: 230), semantic features are to be understood as descriptive features that constitute the result of a semantic decomposition of the meaning of the lexical item. Thus, only non-functional elements have true semantic features. The semantic type and the denotation of functional items, on the other hand, depend on the formal, i.e., the syntactic features. For the derivation of a free relative, this means that even though the D head originates from the wh-phrase, it syntactically behaves like a D head and therefore has the semantics of a D head.

The denotation of the wh-item *was* is given in (208) (cf. Caponigro (2003)).

$$(208) \quad \llbracket \{D, \phi:3sg, c:_, wh, \} \rrbracket = \lambda P : P \in D_{\langle e,t \rangle} [\lambda x : x \in D_e [\neg anim'(x) \wedge P(x)]]$$

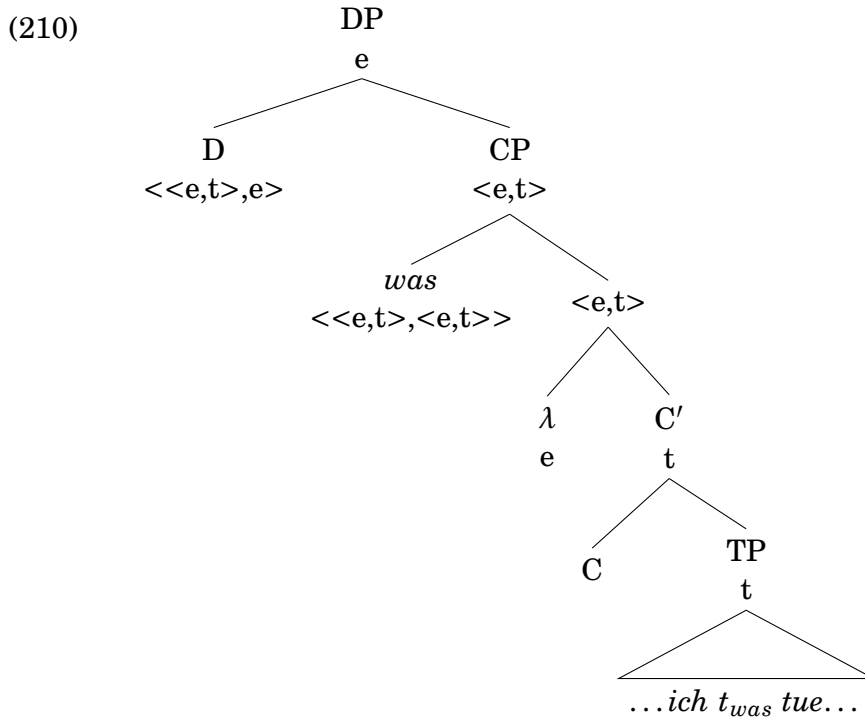
9. Agree in Free Relatives

Following Caponigro (2003), questions are predicate abstracts. He suggests that the *wh*-item *was* “applies to a set of entities to give back all and only the entities of that set that are inanimate” ((Caponigro 2003: 58)).

The semantic operator σ that returns the maximal entity of this set is of type $\langle\langle e,t\rangle,e\rangle$, which is the same type of a D head. Such a D head is indeed available in the theory above due to Create-Dep: it is the head of the DP dominating the embedded CP. The denotation of this head is given in (209).

$$(209) \quad \llbracket [D, \phi:3sg, c_]\rrbracket = \lambda P : P \in D_{\langle e,t \rangle} [\sigma x[P(x)]]$$

The denotation in (209) says that the meaning of the D head is the σ -operator that takes a set $P(x)$ and returns its maximal entity. With these assumptions in mind, the meaning of free relatives can be computed compositionally on the basis of the structure in (210).



- a. $\llbracket \text{CP} \rrbracket = \llbracket \text{was} \rrbracket ([\lambda y : y \in D_e [\text{speaker does } y]])$
- b. $\llbracket \text{CP} \rrbracket = \lambda P : P \in D_{\langle e, t \rangle} [\lambda x : x \in D_e [-\text{anim}'(x) \wedge P(x)] ([\lambda y : y \in D_e [\text{speaker does } y]])$
- c. $\llbracket \text{CP} \rrbracket = \lambda x : x \in D_e [-\text{anim}'(x) \wedge \text{speaker does } x]$
- d. $\llbracket \text{DP} \rrbracket = \llbracket \text{D} \rrbracket (\lambda x : x \in D_e [-\text{anim}'(x) \wedge \text{speaker does } x])$
- e. $\llbracket \text{DP} \rrbracket = \lambda P : P \in D_{\langle e, t \rangle} [\sigma z[P(z)]](\lambda x : x \in D_e [-\text{anim}'(x) \wedge \text{speaker does } x])$
- f. $\llbracket \text{DP} \rrbracket = \sigma x [-\text{anim}'(x) \wedge \text{speaker does } x]$

The crucial point is what happens in the CP and the DP. Movement of *was* to Spec-CP has created a λ -abstract $[\lambda y : y \in D_e [\text{speaker does } y]]$. Now, *was* applies to the set described by the λ -abstract (210-c). Finally the D head applies to (210-c) and returns its maximal entity. The maximal entity is of type e and can be merged in an argument position.

As shown above, the analysis of Caponigro (2003) is fully compatible with the syntactic structure of free relatives as proposed here. Furthermore, the emergence of the σ -operator – a point not explained in Caponigro (2003) – is captured: it is the result of Create-Dep.⁴²

⁴²Note that the analysis of Caponigro (2003) was assumed here for the sake of concreteness. Since the resulting syntactic structure of free relatives is a standard one, any other semantic analysis that is based upon the structure in (201) should be compatible with the syntactic analysis of the present approach.

9.2. Deriving the Behavior of the wh-Phrase

The aim of this section is to show that the wh-phrase in free relatives has to meet the requirements of two clauses at the same time and how this can be derived within the system developed above. This is done by outlining two types of facts: the first type suggests that the wh-phrase is part of the matrix clause while the second type suggests that it is part of the embedded clause.

9.2.1. The wh-Phrase is Part of the Matrix Clause

9.2.1.1. Number Agreement

The first fact that suggests that the wh-phrase is part of the matrix clause involves number agreement. Bresnan and Grimshaw (1978) observed for English free relatives that a plural wh-phrase in a free relative induces plural number agreement in the matrix clause. This observation also holds for other languages, e.g., German.

The German example in (211) shows that a plural wh-subject induces plural number agreement in the clause where it occurs.

(211) *Agreement: matrix clause*

[Welche Bücher] haben/ *hat dir gefallen?
which book.PL have.PL/ have.SG you liked

‘Which books did you like?’

Usually plural wh-phrases that occur in embedded clauses do not affect the number agreement in the matrix clause. This is exemplified in (212) for an indirect interrogative clause in subject position.

9.2. Deriving the Behavior of the *wh*-Phrase

(212) *Agreement: embedded clause*

[_{SClause} Welche Bücher ihm gefallen], ist/ *sind unklar.
which book.PL him like, be.SG/ be.PL unclear

‘It is unclear which books he likes.’

However, in the case of free relatives, the *wh*-phrase induces plural agreement, in contrast to indirect interrogative clauses. Note that, in the sentences in (213), the entire free relative is the subject of the matrix clause.

(213) *Agreement: free relative*

?[_{FR} [_{Rel} welche Bücher] ich auch immer gelesen habe], haben/ *hat
which book.PL I ever read have have.PL/ have.SG
mir gefallen.
me liked

‘I liked whatever books I read.’

If it is true that number agreement in a matrix clause cannot be induced by a constituent in the embedded clause, the *wh*-phrase in (213) must be part of the matrix clause.

Note that this property can only be shown with complex *wh*-phrases in German since simple *wh*-phrases always induce singular number agreement. But the use of complex *wh*-phrases in FRs in German is often considered to be marginal and it requires the use of the particle *auch immer* (“ever”), the status of which is not clear. But as shown in (214) for Spanish, the number agreement property of free relatives also occurs in languages with simple plural *wh*-phrases.

9. Agree in Free Relatives

- (214) [FR Quienes son del sur] son en gran parte bajos.
 who.PL be.PL of.the south be.PL in great part short.PL
 ‘Most people from the South are short.’ (Caponigro 2003: 169)

Again, the plural wh-pronoun *quienes* induces plural agreement in the matrix clause.

Within the present account, the data can be derived as follows: A plural number feature is part of the ϕ -features. As such, it is affected by Create-Dep and is present on both the wh-phrase and the D head. This is shown in (215) for *quienes*.

- (215) WORKSPACE: Create-Dep
- | | | |
|-----|--------------------------------------|----------------------------------|
| ... | quienes | $\emptyset_{quienes}$ |
| ... | {*D*, * ϕ :3pl*, *c:*, wh, ...} | {*D*, * ϕ :3pl*, *c:*, ...} |

If the entire DP headed by the plural $\emptyset_{quienes}$ is the subject of the matrix clause, plural number agreement on the matrix verb is expected.

- (216) [TP [DP { ϕ :3pl, ...} [CP {{ ϕ :3pl, ...}, {wh}, ...} ... T[3pl] ...]] T[3pl] ...]
-

Things are more complicated with the complex wh-phrase in German: I would like to suggest for these cases that the head of the wh-phrase still bears the ϕ -features and that number on nouns is a phenomenon of morphological concord, which is a process different from Agree. This means that the number agreement problem in complex wh-phrases is treated as an instance of the more general concord problem, namely that the interaction of agreement within DPs and agreement with DPs is difficult to derive.

9.2. Deriving the Behavior of the *wh*-Phrase

This problem is usually observed with case. Case values have to be assigned to entire DPs first, before the case value can be shared among the different constituents inside the DP, such as adjectives and nouns. This order, however, would violate the Strict Cycle Condition. But obeying the Strict Cycle Condition leads to the lack of case concord since no case values are available inside of the DP at first.

The case concord problem can be solved by invoking a new concept such as feature sharing (see e.g. Frampton and Gutman (2000); Pesetsky and Torrego (2007)) or by assuming that it is a late process (see e.g. Arregi and Nevins (2012)). Independent of what solution is the right one, concord phenomena pose a general problem for all theories of agreement.

9.2.1.2. *Extraction*

The second fact that suggests that the *wh*-phrase is part of the matrix clause comes from extraction data. (Rooryck 1994: 197) observed for English that nothing can be extracted out of a free relative unless it is part of the *wh*-phrase.

- (217)
- a. I will eat [_{FR} whatever the chef recommends to that person]
 - b. *This is the person [_{RC} to whom_{*i*} I will eat [_{FR} whatever the chef recommends *t_i*]]
 - c. This is the author [_{RC} of whom_{*i*} I buy [_{FR} [whatever books *t_i*]] the NYT recommends to its readers]

In (217-b) movement of a relative pronoun out of a free relative is shown to be impossible. However, movement becomes better if the relative pronoun is extracted from the *wh*-phrase, as shown in (217-c).

9. Agree in Free Relatives

This observation also holds for German (see Ott (2011)). The extraction property is illustrated for topicalization in German. In general, topicalization in German may cross a *wh*-island (Fanselow (1987); Müller and Sternefeld (1993)), see (218-b).⁴³

(218) *Extraction: wh-complement clause*

- a. Ich weiß [_{OC} welche Bücher *Der Spiegel* diesen Leuten
I know which books *Der Spiegel* these people
empfiehlt]
recommends
'I know which books *Der Spiegel* recommends to these people.'
- b. ?Diesen Leuten_i weiß ich [_{OC} welche Bücher *Der Spiegel* *t_i*
these people know I which books *Der Spiegel*
empfiehlt].
recommends
'As for these people, I know which books *Der Spiegel* recommends to
them.'
- (based on (Ott 2011: 188f))

However, topicalization out of a free relative results in strong ungrammaticality if the category is extracted from within the free relative, see (219-b). In contrast, if the category is a part of the *wh*-phrase, topicalization is possible, see (219-c).

(219) *Extraction: free relative*

- a. Ich lese [_{FR} welche Bücher von Jostein Gaarder auch immer
I read which books by Jostein Gaarder ever
Der Spiegel diesen Leuten empfiehlt]
Der Spiegel these people recommends
'I read whatever books by Jostein Gaarder *Der Spiegel* recommends to
these people.'

⁴³Note that topicalization of a PP that originates inside the *wh*-phrase ((218-b)) is slightly worse.

This might be due to parsing problems since there is a second reading where the PP is interpreted as a constituent of the matrix clause: *Jostein Gaarder* is the source of the information of which books *Der Spiegel* recommends.

9.2. Deriving the Behavior of the wh-Phrase

- b. *Diesen Leuten_i lese ich [_{FR} welche Bücher von Jostein Gaarder
these people read I which books by Jostein Gaarder
auch immer *Der Spiegel* *t_i* empfiehlt].
ever *Der Spiegel* recommends
'As for these people, I read whatever books by Jostein Gaarder *Der Spiegel* recommends to them.' *based on (Ott 2011: 188f)*
- c. ?Von Jostein Gaarder_i lese ich [_{FR} [welche Bücher *t_i*] auch immer
by Jostein Gaarder read I welche books ever
Der Spiegel empfiehlt].
Der Spiegel recommends
'As for Jostein Gaarder, I read whatever books by him *Der Spiegel* recommends.'

Again, these data suggest that the wh-phrase is actually part of the matrix clause. Under this assumption, the contrast between (219-b) and (219-c) follows immediately: in (219-b), topicalization is excluded since it involves extraction out of a relative clause island. In (219-c), however, the topicalized PP is extracted out of an object DP in the matrix clause which is possible in German as can be seen in (220).

- (220) Von Jostein Gaarder_i habe ich schon [viele Bücher *t_i*] gelesen.
by Jostein Gaarder have I already many books read
'I have already read many books by Jostein Gaarder.'

In those cases where extraction out of a free relative seems to be possible, I claim that extraction actually proceeds from outside the free relative. This is given, if the PP complement also has the option of being merged to the covert head outside the free relative. If it is merged outside the free relative, it can be topicalized.

This option is given under the following two assumptions: First, the PP complement above is optional. If it is present, the PP is merged with a head H, if H bears a feature [\bullet P \bullet]. Thus, some Merge features must be optional. If a feature is optional,

9. Agree in Free Relatives

it can be added to an item before this item undergoes Merge.⁴⁴ This means that there are two rules that can apply to a lexical item before it is merged with another item and has to obey lexical integrity: It can “borrow” features (Create-Dep) and it can receive (optional) features. If these two rules interact, there are three possible results shown in (221).⁴⁵

(221) a. WORKSPACE: Adding [$\bullet P \bullet$]

wh	
{D, ϕ :3pl, c:_, wh, $\bullet P \bullet$, ...}	
WORKSPACE: Create-Dep	
wh	\emptyset_{wh}
{*D*, ϕ :3pl*, c:_*, $\bullet P \bullet$ *, wh, ...}	{*D*, ϕ :3pl*, c:_*, $\bullet P \bullet$ *}

b. WORKSPACE: Create-Dep

wh	\emptyset_{wh}
{*D*, ϕ :3pl*, c:_*, wh, ...}	{*D*, ϕ :3pl*, c:_*}
WORKSPACE: Add [$\bullet P \bullet$]	
wh	\emptyset_{wh}
{*D*, ϕ :3pl*, c:_*, $\bullet P \bullet$ *, wh, ...}	{*D*, ϕ :3pl*, c:_*}

⁴⁴This is a slight divergence from Chomsky (1995) who claims that optional features are “added as the LI enters the numeration”[p.231].

⁴⁵Note that I assume that [$\bullet P \bullet$] is added to D heads, not to N heads. The latter option might be possible as well for other derivations, but D heads must in principle be able to host such a feature. This is crucial for the derivation. Evidence for this assumption comes from examples such as (i) where an overt D head *das* is combined with a *von*-PP.

- (i) Ich habe [_{DP} das [_{PP} von ihm]] gelesen.
 I have that by him read.
 ‘I have read that one by him.’

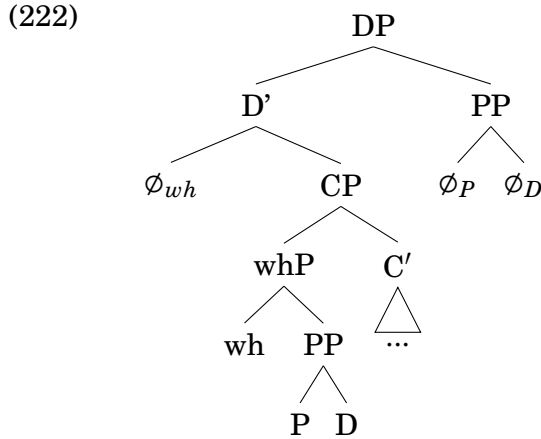
c. WORKSPACE: Create-Dep

$$\begin{array}{cc} \text{wh} & \emptyset_{\text{wh}} \\ \{ *D*, *\phi:3\text{pl}*, *c: *_*, \text{wh}, \dots \} & \{ *D*, *\phi:3\text{pl}*, *c: *_* \} \end{array}$$

WORKSPACE: Add [$\bullet P \bullet$]

$$\begin{array}{cc} \text{wh} & \emptyset_{\text{wh}} \\ \{ *D*, *\phi:3\text{pl}*, *c: *_*, \text{wh}, \dots \} & \{ *D*, *\phi:3\text{pl}*, *c: *_*, \bullet P \bullet \} \end{array}$$

In (221-a), feature adding applies before Create-Dep. Then, the feature [$\bullet P \bullet$] is present on both items. This is problematic since it requires creating dependents of the preposition and the D head inside the PP to undergo Create-Dep as well. The probe features resulting from these additional applications of Create-Dep cannot be checked in the derivation because at least the two D heads inside the PP will never c-command each other. This is shown in (222).⁴⁶



⁴⁶In light of the discussion around the structure in (174), other potential combinations of probe and goal, such as *wh* and \emptyset_D are ruled out as well. Note further that the structure shown in (222) assumes that the CP is closer to the D head than the PP.

9. Agree in Free Relatives

In the options (221-b) and (221-c), feature adding applies after Create-Dep. Both these options lead to converging derivations, with the result that the PP can be merged either in the free relative (option (221-b)) or in the matrix clause (option (221-c)).

9.2.2. The wh-Phrase is Part of the Embedded Clause

Besides the matching effects –such as case matching (see part I), number matching (see section 9.2.1.1 and categorial matching (see footnote 40) – extraposition shows that the wh-phrase has to be part of the embedded clause. The observation made by Groos and Riemsdijk (1981) is that if the free relative is extraposed, the wh-phrase is extraposed as well, see (223). This is unexpected under the assumption that the wh-phrase is outside the embedded clause (as in the analysis of Bresnan and Grimshaw (1978)).

In German, there is a ban on extraposing DPs. (Finite) CPs, on the other hand, are preferably extraposed. The data in (223-b) and (223-c) suggest that the wh-phrase *was* is part of a CP and is not the external D head of the free relative. Otherwise, (223-c) should be grammatical and (223-b) ungrammatical.

(223) *Extraposition: free relative*

- a. Ich denke, dass ich [_{FR} was ich mag] essen kann.
I think that I what I like eat can
'I think that I can eat what I like.'
- b. Ich denke, dass ich *t*_{FR} essen kann, [_{FR} was ich mag].
I think that I eat can what I like
'I think that I can eat what I like.'

- c. *Ich denke, dass ich [was t_{FR}] essen kann, [_{FR} ich mag].
 I think that I what eat can I like
 ‘I think that I can eat what I like.’

In (224), it can be seen that indirect interrogative clauses in subject position, where the *wh*-phrase clearly belongs to the embedded CP, show the same behavior as free relatives when it comes to extraposition.⁴⁷

(224) *Extraposition: indirect interrogative clause*

- a. [_{SClause} Was ich mag] ist unklar.
 what I like is unclear
 ‘It is unclear what I like.’
- b. Es ist unklar [_{SClause} was ich mag].
 it is unclear what I like
 ‘It is unclear what I like.’
- c. *[_{SClause} Was t_i] ist unklar [ich mag]_i.
 what is unclear I like
 ‘It is unclear what I like.’

On the other hand, as shown in (225), if the *wh*-phrase is the external head of a relative clause CP, it cannot be extraposed along with the relative clause.⁴⁸

⁴⁷Note that the clause in ((224-a)) can also be understood as a free relative. Under this reading, the matrix predicate *be unclear* would predicate over the things I like. The intended reading in ((224-a)) is, however, the reading of the indirect interrogative clause, i.e., the question of what I like cannot be answered clearly.

⁴⁸Note that relative clauses headed by *wh*-phrases are generally considered to be a bit quirky in German. Nevertheless, (225-a) and (225-c) are grammatical and stand in sharp contrast to (225-b).

9. Agree in Free Relatives

(225) *Extraposition: what-headed relative clause*

- a. ?Wer hat [DP was [R_{Clause} das ich gern gegessen hätte]]
who has what that I gladly eaten would have
weggeworfen?
thrown away
- b. *Wer hat weggeworfen [DP was [R_{Clause} das ich gern gegessen
who has thrown away what that I gladly eaten
hätte]]?
would have
- c. ?Wer hat [DP was t_i] weggeworfen [R_{Clause} das ich gern gegessen
who has what thrown away that I gladly eaten
hätte]_i?
would have

Thus, this argument strongly suggests that the wh-phrase in FR is actually part of the embedded clause.

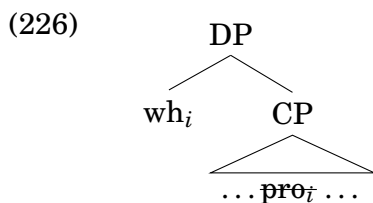
In the present account, the data follow since Create-Dep does not affect phonological features. The result of this restriction is that the newly created D head does not possess any phonological features. The phonological features are only available on the wh-item *was*, that is, all the phonological features are part of the embedded CP. This in turn means that the wh-phrase must be pronounced within the CP, i.e., if the CP is extraposed, the wh-phrase must be extraposed as well.

9.3. Alternative Approaches

In this section, I will summarize previous approaches to free relatives. All of them are deficient in that they cannot derive some of the properties of free relatives discussed in section 9.2.

9.3.1. External-Head Accounts

I start this overview with a discussion of one of the oldest formal approaches to free relatives. Due to facts like the ones presented in section 9.2.1, Bresnan and Grimshaw (1978) claimed that the *wh*-phrase is the external head of a relative clause which contains a pronoun that is bound by the *wh*-phrase and undergoes a process of *Controlled Pro-Deletion*. The rough structure is depicted in (226).⁴⁹



The main problem with this approach is certainly its conflict with the extraposition data from languages like German, as first pointed out by Groos and Riemsdijk (1981) and discussed in section 9.2.2. Within these approaches, it must be possible to extrapose DPs in order to account for the data. However, DP extraposition is not attested otherwise in German. Thus, the approach cannot derive the empirical facts correctly.

9.3.2. Comp Accounts

9.3.2.1. Groos and Riemsdijk (1981)

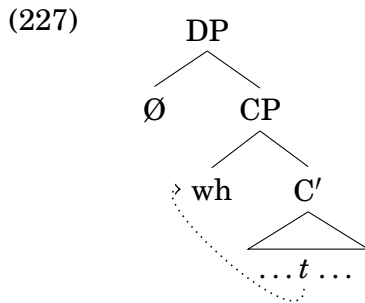
In order to derive the extraposition data, Groos and Riemsdijk (1981) propose that the *wh*-phrase is inside a relative clause that is headed by an empty category.⁵⁰ The

⁴⁹See also Bresnan (1973); Daalder (1977); Larson (1987) for other head-external approaches.

⁵⁰The earliest version of this approach was pursued by Kuroda (1968). Other variants can be found in Hirschbühler and Rivero (1981); Harbert (1983); Suñer (1984); Grosu and Landman (1998);

9. Agree in Free Relatives

analysis is sketched in (227).



Even though the structure in (227) is identical to the structure of free relatives in the present approach, former proposals as to how this structure arises lack an explanation for the special relationship between the covert head and the wh-phrase. In principle, there are two ways to model this relation: (i) excessive Agree which basically copies all the necessary features from the wh-phrase onto the covert head (even the categorial feature) and (ii) the approach developed above where the covert head emerges directly from the wh-phrase and has the necessary features to begin with.

In both options, all the data discussed above follow in the same way. However, if such categories are part of the lexicon, it is unclear why they are always covert and why they do not bear any features of their own, such as wh-features (see also section 8.3.2). In general, it also seems that leaving the existence of this covert D head to the lexicon would give rise to much more variation between languages and allow for various kinds of such covert heads.

As for the first problem, the difference between free relatives and light headed relatives, for example concerning case matching, does not follow independently. It

Grosu (2003); Caponigro (2002); Gračanin-Yuksek (2008) among others.

has to be assumed that the lack of phonological features is the reason why Agree with a *wh*-phrase is required.

- (228) a. *Ich folge Ø [_{FR} wem/wen ich bewundere]
 I follow_{dat} who_{dat}/who_{acc} I adore_{acc}
 ‘I follow who I adore.’
- b. Ich folge dem [_{Rel} den ich bewundere]
 I follow_{dat} that.one who_{dat}/who_{acc} I adore_{acc}
 ‘I follow the one who I adore.’

In the present account, this difference follows on a more systematic way: the derivation of a light-headed relative just involves Merge of a D head and a CP. No tight link between the two is expected. In free relatives, this tight link emerges from the fact that a free relative is a sharing construction, which requires the application of Create-Dep.

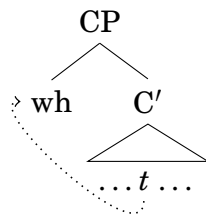
So, even though the present approach and the approach by Groos and Riemsdijk (1981) do not differ empirically, the latter approach does not respond to the question of how agreement in free relatives emerges.

9.3.2.2. *Rooryck (1994)*

The next approach, which I would like to discuss, was first proposed by Rooryck (1994) (see also Caponigro (2003)). As in Groos and Riemsdijk’s (1981) account, the *wh*-phrase is located inside the embedded CP. In contrast to this approach, however, the CP is directly merged in an argument position in the matrix clause. The structure is shown in (229).

9. Agree in Free Relatives

(229)

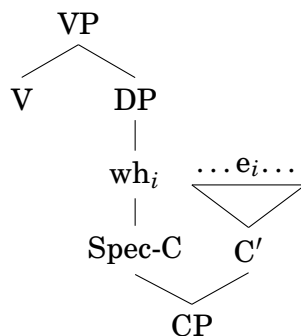


Even though, this approach may capture the extraction facts due to a lack of a higher DP shell, it suffers from an obvious problem: it is unclear why the CP can occur in positions that are entirely reserved for DPs. On these grounds, this theory does not constitute an ideal approach to free relatives.

9.3.3. Multidominance

In the grafting approach by Riemsdijk (2006), the *wh*-phrase is simultaneously part of both the embedded and the matrix clause. This comes about by grafting, which is a form of multidominance. The structure is shown in (230).

(230)



Even though Riemsdijk's (2006) analysis is intuitively close to the present approach in that both assume that the *wh*-phrase is part of two clauses, the grafting approach faces at least two kinds of problems: First, there are the conceptual problems related to grafting, and to multidominance more generally. Furthermore, it is not transparent how extraposition is handled in this account: if extraposition can apply,

because the free relative is a CP, the CP property should potentially also play a role for other syntactic properties, such as selectional constraints.

The most urgent problem, however, is that the case matching facts do not follow (see section 6.2.1).

9.3.4. Derived-Head Accounts

The following summarizes and discusses two recent derived-head accounts: Ott (2011) and Donati and Cecchetto (2011).⁵¹ In this type of theory, the *wh*-phrase is base-generated in the embedded CP and moves to a position where it becomes the head of the clause.

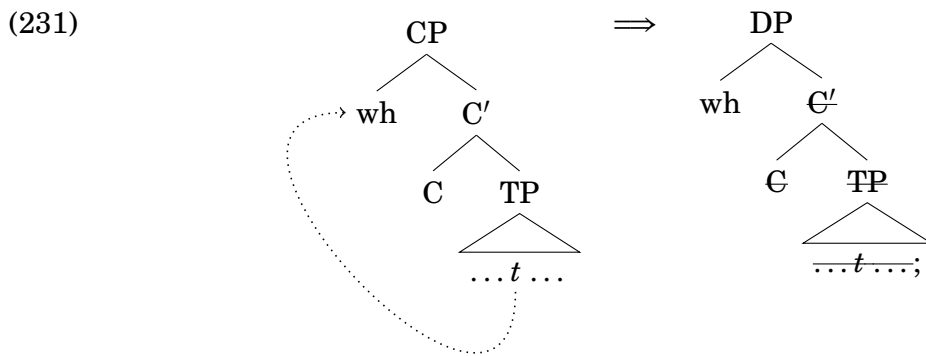
9.3.4.1. Ott (2011)

In Ott's (2011) analysis, the free relative starts out as a normal CP in which the *wh*-phrase is moved to Spec-C. Then, spell-out applies not only to the complement of C but to the C head as well because it does not bear any interpretable features (in contrast to, e.g., embedded questions). Since the head of the CP has been sent to Transfer, only the *wh*-phrase remains and becomes the head of the phrase. The important steps of the derivation are shown in (231).

⁵¹Other approaches of this kind include, e.g., Hirschbühler (1976); Bury and Neeleman (1999);

Iatridou et al. (2002)

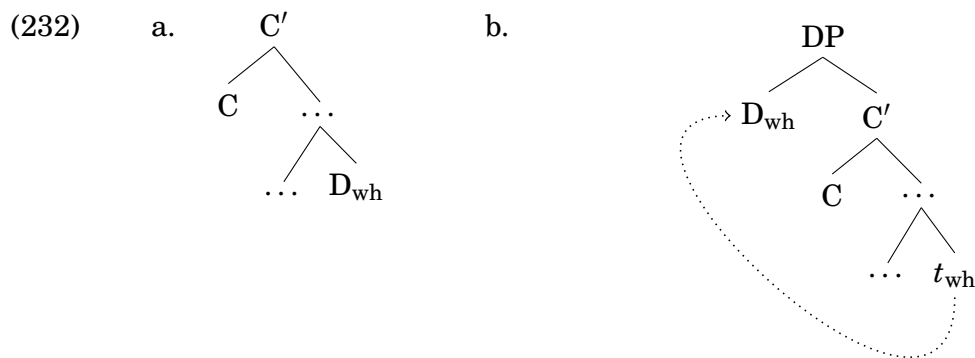
9. Agree in Free Relatives



Similarly to the account by Bresnan and Grimshaw (1978), Ott's account has a hard time dealing with the extraposition data, because in order to account for the facts, the DP should be able to extrapose, contrary to fact. To this end, Ott (2011: fn.5,p.186) proposes two solutions: The first solution is that extraposition applies post-syntactically after a process of intonational phrasing that assigns a clausal intonation to the free relative. The second solution is that CPs are base-generated in the right periphery and can undergo leftward-movement. Still, both solutions do not capture the data correctly. If only constituents with a clausal intonation could undergo extraposition, PPs are not supposed to be able to extrapose, contrary to fact. The second solution does not tell us anything about the situation in (231). At the point when the free relative is built, it is already a DP and should thus not be able to remain in the right periphery. As far as I can see, these problems cannot be overcome without referring to the syntactic labels "DP" and "CP", the latter not being available in this approach.

9.3.4.2. Donati and Cecchetto (2011)

Finally, there is the reprojection approach by Donati and Cecchetto (2011). The main idea here is that the *wh*-phrase is merged inside a CP and moved to Spec-CP. If it is a simple *wh*-phrase like *was* ('what') it may reproject as a head and turn the CP into a DP. The derivation is sketched in (232).



Similarly to Bresnan and Grimshaw (1978) and Ott (2011), the category dominating the *wh*-phrase is a DP. Therefore, the extraposition data do not follow straightforwardly from this analysis. Furthermore, the analysis bans complex *wh*-phrases from occurring in free relatives. Donati and Cecchetto (2011) explicitly discuss this issue, claiming that free relatives that contain a complex *wh*-phrase followed by *ever* are not free relatives. However, it also remains unclear how, for example, German free relatives are derived where the *wh*-phrase is contained in a PP or where it is a possessor (see also Grosu (1996, 2003)).

- (233) a. Ich lade ein [_{FR} [_{PP} auf wen] sich auch Maria freuen würde]
 I invite on who self also Maria be.happy would
 'I invite whoever Maria would also be happy to meet' (Vogel 2001: 904)

9. Agree in Free Relatives

- b. [_{FR} Wessen Birne noch halbwegs in der Fassung steckt] pflegt
 whose bulb still halfway in the socket sticks uses
 solcherlei Erloschene zu vermeiden
 such extinct to avoid
 ‘Whoever still has half of his wits tends to avoid such vacant characters’

(Müller 1999: 78)

In sum, the discussion of the previous approaches has shown that all of them lack an explanation for at least one crucial property of free relatives or lack an obvious way of deriving certain properties. Most of them even make empirically wrong predictions.

9.4. Interim Summary

The aim of this chapter was to provide a closer look at free relative constructions. I have shown that free relatives are indeed sharing constructions with the wh-phrase having to meet requirements of the matrix clause (number agreement, extraction) and of the embedded clause (extraposition).

I have further shown that these facts follow from invoking the operation Create-Dep into the derivation of free relatives. The present approach can be seen as a version of the theory by Groos and Riemsdijk (1981) that assumed that the CP is headed by a covert D head, which I argued to be the only empirically correct structure for free relatives. The advantage of the present approach, however, is that the close link between this covert D head and the wh-phrase can now be explained: the covert head is created out of the wh-item.

10. Conclusion of Part II

After having shown in part I of the thesis that parasitic gap and free relative constructions involve bidirectional Agree, part II has dealt with the origin of this Agree relation. I have argued that the reason for Agree is the same in both constructions: It is possible to create covert items out of existing overt ones with the condition that the two have to Agree with each other throughout the derivation. This essential step in the derivation of parasitic gaps and free relatives is an application of a new syntactic operation Create Dep. The definition is repeated in (234).

(234) *Create-Dep(endent)*

- a. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{*f_1*, \dots, *f_m*, f_{m+1}, \dots, f_n\}, \{f_1, \dots, f_m\}$
- b. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{f_1, \dots, f_m, f_{m+1}, f_{m+1}, \dots, f_n\}, \{*f_1*, \dots, *f_m*\}$
- c. $\{f_1, \dots, f_m, f_{m+1}, \dots, f_n\} \rightarrow$
 $\{*f_1*, \dots, *f_m*, f_{m+1}, \dots, f_n\}, \{*f_1*, \dots, *f_m*\}$

The existence of such an operation is not surprising considering that both lexical

10. Conclusion of Part II

items and syntactic structure share certain properties, such as integrity. Crucially, Create-Dep is also one origin of the cross-linguistic variation of case matching discussed in Part II.

With this background, the main achievement of this part is that the otherwise unexplained similarities between parasitic gap and free relative constructions, as well as each of their construction-specific properties could be derived coherently.

This is also a major advantage over previous approaches which have not made the connection between the two constructions.

The discussion in this part has also laid out the foundations for a new perspective on other sharing constructions, including across-the-board-movement, right-node-raising, and maybe cases of control. Even more generally, there is a new perspective on every phenomenon where one item seems to be placed in more than one position but where a certain asymmetry concerning the positions can be observed.

Part III.

Defining Agree

11. Introduction

In this last part of the thesis, I will look the assumptions about Agree made so far from a broader perspective. In doing so I will focus especially on the c-command condition of Agree, concluding that c-command is necessary, but that the directionality of Agree should not be restricted.

Since the introduction of Agree to the minimalist framework in Chomsky (2000, 2001) there have been discussions about the conditions of Agree. The definition of Agree as needed to derive all the data presented so far is given in (235).

- (235) a. Agree-Link($\alpha[*F*]$, $\beta[F]$) if
- (i) α c-commands β or β c-commands α
 - (ii) α and β are not separated by a (language-specific) barrier for Agree
- b. Agree-Copy($\alpha[F:val_1]$, $\beta[F:val_2]$) =
- (i) $\alpha[F:val_2]$ if (235-a) and $val_1 = \emptyset$
 - (ii) $\alpha[F:val_1]$ if (235-a) and $val_2 = \emptyset$
 - (iii) $\alpha[F:val_1]$ if (235-a), $val_2 \neq \emptyset$, $val_1 \neq \emptyset$ and Match($\alpha[F:val_1]$, $\beta[F:val_2]$)

11. Introduction

The definitions in (235) define two Agree operations Agree-Link and Agree-Copy, with Agree-Link being a precondition for Agree-Copy. Both operations can only apply, if certain conditions are met. For Agree-Link, c-command and locality is important. As for now, the term *barrier* includes absolute barriers such as islands and relative barriers such as intervening goals. For Agree-Copy, three different results are defined, depending on how the feature values of probe and goal look like.

Abstractly, the conditions on Agree in general are the existence of an Agree-triggering feature, c-command, locality, and matching. So, in its core Agree, is still the operation first defined in Chomsky (2000).

In the first chapter of this part (chapter 12), I will redefine the Agree as an algorithm that is part of the more general algorithm for derivations (see also work by Collins, Epstein, Seel, Kitahara, Graf, Kobele, Michaelis, and others for a related approach to syntactic operations). This has the advantage that all the conditions can be part of the algorithm and therefore part of the derivation. In this chapter, I will also discuss each of these conditions. Afterwards, in chapter 13, I will return to a more empirical discussion of Agree and show that bidirectional Agree is empirically superior to unidirectional Agree.

12. A Definition of Agree

12.1. A Definition of Agree-Link

An algorithm for Agree-Link is given in (236). This algorithm can be understood as a program that runs everytime it is triggered. Thus, Agree-Link is split into a bunch of single steps, each searching for a matching goal.

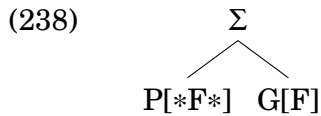
```
(236) Agree-Link (X[*F*], [A, B])
      //Upward
<1>  IF Contains(A,X)
<2>    IF Feat(B,F) RETURN (X,B);
<3>    ELSE RETURN; //no Goal found
<4>  ELSEIF Contains(B,X)
<5>    IF Feat(A,F) RETURN (X,A);
<6>    ELSE RETURN; //no Goal found
      //Sister
<7>  ELSEIF A = X
<8>    IF Feat(B,F) RETURN (X,B);
<9>    ELSE Agree(X[*F*], B);
<10> ELSEIF B = X
<11>  IF Feat(A,F) RETURN (X,A);
<12>  ELSE Agree(X[*F*], A);
      //Rest of C-Command Domain
<13> ELSE
<14>  IF Feat(A,F) RETURN (X,A);
<15>  ELSEIF Feat(B,F) RETURN (X,B);
<16>  ELSEIF A = [C,D] Agree (X[*F*], A);
<17>  ELSEIF B = [C,D] Agree (X[*F*], B);
<18>  ELSE RETURN; //no Goal found
```

12. A Definition of Agree

```
(237)  Contains([A,B], X)
        IF Dominates(A, X) RETURN true;
        ELSEIF Dominates(B, X) RETURN true;
        ELSEIF A = [C,D] Contains(A, X);
        ELSEIF B = [C,D] Contains(B, X);
        ELSE RETURN false;
```

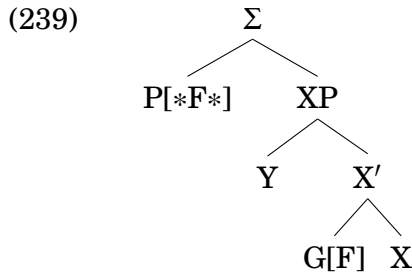
The algorithm in (236) describes the search of a probe for a goal in a structure $[A, B]$. I assume that this search is first initiated after the head bearing the probe feature is merged in the structure. At this point, it is able to look for a goal in its c-command domain. If no goal has been found in the c-command domain, Agree-Link has to be executed again until a goal is found. This will create cases of upward Agree. I will illustrate the workings of the algorithm by four abstract configurations between a probe P and a potential goal G .

We start with the most simple configuration – Agree under sisterhood – in (238).



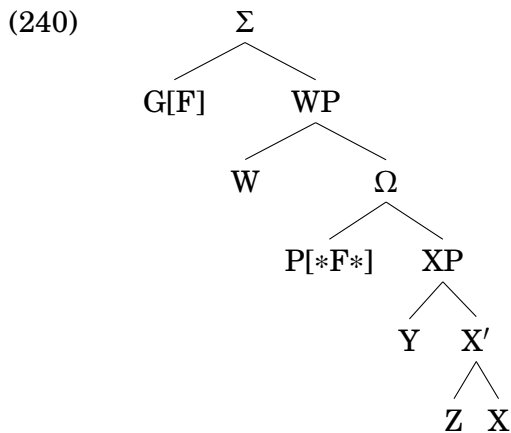
At the point when P is merged with G , the Agree-Link($P[*F*]$, Σ) is triggered. According to (236), the first condition to be checked is <1> Contains($[P,G]$, P). According to the definition of the function Contains in (238), this is not true assuming that Dominates is defined non-reflexively. Then, the equivalence condition <7> is checked. Since $P=P$, the sister of P , which is G , can be checked for a matching feature F . This is what the function Feat does. Since G bears F , the Agree-Link P, G is returned.

The next configuration is the c-command configuration in (239), where P c-commands G .



We start again with $\text{Agree-Link}(P[*F*], \Sigma)$. The first two conditions $\langle 1 \rangle / \langle 4 \rangle$ are not fulfilled. However, the condition $\langle 7 \rangle$ ($A = X$) is true and the sister of P , which is XP in (239), is checked for F . Since this is not true, Agree-Link applies to P and XP ($\langle 9 \rangle$). Now, the first conditions $\langle 1 \rangle, \langle 4 \rangle, \langle 7 \rangle, \langle 10 \rangle$ do not apply anymore, since P is not dominated by XP . The result of the operation $\text{Agree-Link}(P[*F*], [Y, X'])$ is that, because of $\langle 17 \rangle$, $\text{Agree-Link}(P[*F*], [G[F], X])$ applies: Neither Y , nor X' bears $[F]$ and X' is complex. Then, if $\text{Agree-Link}(P[*F*], [G[F], X])$ applies, the $\text{Agree-Link}(P, G)$ is established because of $\langle 14 \rangle$.

Next, we turn to a case of upward Agree. The configuration is given in (240).



At the point when P is merged, Agree-Link applies to P and Ω . Ω in (240) is the same structure as Σ in (239), the only difference being the lack of a goal for P . Going through all the steps as in (239), no condition is fulfilled and Agree-Link stops

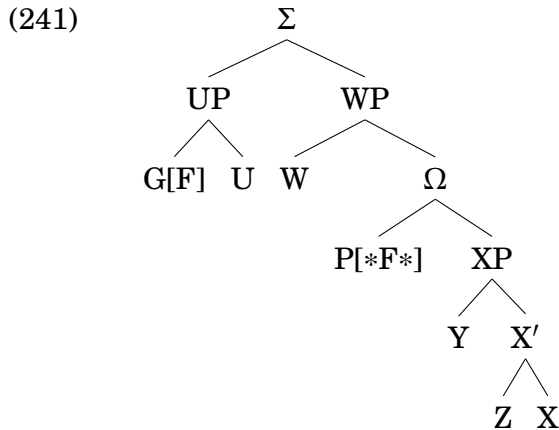
12. A Definition of Agree

eventually without having found a goal for P (<18>).

When W is merged, Agree-Link applies again to P and WP. What happens now, is that the condition <4> is fulfilled: Ω dominates P. Thus, the sister of Ω (W) is checked for a feature F. If this is not the case, Agree-Link stops, again, without a goal for P.

When G is merged, Agree-Link applies again to P and Σ . Now, conditions <4> and <5> are fulfilled and the Agree-Link (P, G) is established.

A final configuration, is one where neither P c-commands G nor G c-commands P. This is shown in (241).



Everything proceeds as in (240). But at the point when Agree-Link applies to P and Σ , no Agree-Link can be established. Here is why: UP does not bear F. At this point, the search algorithm stops, even though P has not found goal (<6>). Thus, the algorithm demands that at some point in the derivation, c-command holds between P and G.

The intuition behind this way of dealing with the c-command condition is the following: Some sort of search algorithm for finding goals is needed in any theory

that adopts Agree. Thus, although explicit, the general idea that Agree involves some sort of algorithm is not new. The algorithm defined in (236) can be summarized in three phases:

1. Look at your sister for a goal.
2. Look at everything your sister dominates.
3. Look at every new item merged into your tree.

The first two steps happen as soon as the probe is merged, simply due to earliness (Pesetsky (1989); Řezáč (2004a)). If the search for a goal hasn't been successful right away, the probe has to wait to find a goal. But at this point, the probe does not have to check the entire tree again but simply looks at every new item (simple head or complex phrase) that is merged to the root. Note that this solution presupposes that features project from heads to maximal projections.

In sum, an algorithmic definition of Agree-Link elegantly subsumes the search condition and offers an easy solution to the configurations where the goal c-commands the probe: C-command just falls out of the definition of the search algorithm. Note further that the complexity of this algorithm is linear: No structure is searched twice and the number of applications of Agree-Link is directly proportional to the number of nodes, by which probe and goal are separated. As such, this bidirectional Agree is computationally (and conceptually) not more complex than a unidirectional Agree operation (see (Chandra 2011: 12) for a similar conclusion). The main reason for this is that Agree-Link is not directly defined between probe and goal, but between a probe and a complex structure that might or might not contain the goal.

Before continuing with a definition for Agree-Copy, a note on locality is in order.

12. A Definition of Agree

First, intervention configurations are excluded by the algorithm without further assumptions: As the search algorithm goes top down and stops when it has found a goal, no potential goal can intervene between the probe and the intended goal. Second, absolute locality restrictions, which I take to be language-specific (cf. footnote 32), might be integrated by including a further variable depth: depth is a counter that is increased every time Agree-Link starts over. If depth has reached a number n , Agree-Link returns without a goal. If this solution is pursued, depth has to be set back at the point when upward Agree starts. The algorithmic approach to Agree offers an easy solution to include all these language-specific factors.

12.2. A Definition of Agree-Copy

The algorithm for Agree-Copy is a lot simpler and mainly consists of distinguishing various combinations of feature values.

```
(242)  Agree-Copy(P[\as{F:val1}], G[F:val2])
        IF      !val1 RETURN P[F:val2];
        ELSEIF !val2 RETURN P[F:val1];
        ELSEIF val1 = val2 RETURN P[F:val1];
        ELSEIF Feat(P,D) AND Feat(G,D) RETURN;
        ELSEIF (F = c) AND (val2 < val1) RETURN P[F:val1];
        ELSE RETURN;
```

The algorithm in (242) takes an Agree-Link consisting of probe and goal and returns a valued probe, that has its probe feature checked. The success of this valuation depends on various conditions. All these conditions have been established in part I of this thesis. Of course, the algorithm might be expanded if further construction- or language-specific requirements need to be fulfilled.

The first two cases in (242) are trivial: Either the probe or the goal is unvalued. If the probe is unvalued (!val1), it receives the value of the goal, which might be empty itself. If the goal is unvalued (!val2), the probe sticks with its value.

The next three cases are given, if both probe and goal are valued, which might arise in Agree relations in parasitic gap and free relative constructions. If the values are identical, the probe simply keeps its value. If they are not identical, the two different matching conditions in (84) are looked at. If probe and goal are both D elements ($\text{Feat}(P,D) \text{ AND } \text{Feat}(G,D)$), which happens in the sharing constructions, mismatches are not tolerated. The algorithm stops without valuing the probe feature. If one of them is not a D element and if the feature to be valued is a case feature, the subset condition is invoked ($\text{val2} < \text{val1}$). If it is met, Agree-Copy is successful.⁵² In all other cases, if the subset condition is not met or if the feature is not case, a mismatch is not tolerated.

This definition of Agree-Copy suffices for all the data analyzed so far. The main advantage of this approach to Agree-Copy is that specific matching conditions can be clearly defined and checked. If matching is not given, Agree-Copy fails.

In sum, an algorithmic definition of Agree-Link and Agree-Copy is elegant and solves a lot of issues resulting from an imprecise definition of Agree.

⁵²There are two options here: Either P simply keeps its value or the value of the goal is added on top, as suggested in the derivations of the syncretism effects in section 5.1.

13. The Direction of Agree

There are three different Agree types with respect to the direction of Agree: The first type is *Downward Agree*, which only allows Agree to apply in configurations where the probe c-commands the goal. This is the type most often found in the literature (Chomsky (2000, 2001), most recently Preminger (2015); Preminger and Polinsky (2015)). The second type is *Upward Agree*. Here the goal always has to c-command the probe. This idea has been argued for in Zeijlstra (2012) et seq. Finally, the third type is *Bidirectional Agree*. This is the type assumed in the present account of Agree and is also proposed in a lot of other work on Agree (see for example Merchant (2006); Baker (2008b); Carstens (2016); Abels (2012); Bjorkman (2014)). In this type the direction of Agree is underspecified. Depending on the concrete analysis, this underspecification can be given in general, only in some languages, only in some constructions, or only in some features.

The aim of this chapter is to show that leaving the direction of Agree completely underspecified is the most promising approach. In order to do so, three steps are necessary. First, it has to be shown that there are cases where bidirectional Agree is necessary. This step has already been done in part I of this thesis, where I have argued that bidirectional Agree is necessary in order to derive case matching effects.

13. *The Direction of Agree*

Next, it has to be shown that there are no conceptual reasons to assume that Agree is unidirectional. This, I have shown in chapter 12. Now, there is only one step left, namely to show that suggested counterarguments against bidirectional Agree are not very strong.

For this final step, it is useful to differentiate between four types of potential arguments against bidirectional Agree. The first type of argument tries to prove the existence of downward Agree. The second type tries to prove the existence of upward Agree. The third type tries to prove a ban on downward Agree. Finally, the last type tries to prove a ban on upward Agree. I will go through each of these arguments in turn.

13.1. **Yes to Downward Agree**

The goal of this type of argument is to show that there must be an Agree operation that applies downward. A standard argument of this type comes from long-distance agreement in Tsez (Polinsky and Potsdam (2001), further discussed in Preminger (2015); Preminger and Polinsky (2015))

- (243) a. Eni-r [uži ϕ-āy-ru-hi] ϕ-iy-xo.
 mother-DAT boy.I.ABS i-arrive-PST.PRT-NMZL I-know-PRES
 ‘The mother knows that as for the boy, he arrived.’
- b. Eni-r [už-ā magalu b-āc’-ru-hi]
 mother-DAT boy-ERG bread.III.ABS III-eat-PST.PRT-NMZL
 b-iy-xo.
 III-know-PRES
 ‘The mother knows that as for the bread, the boy ate it.’

(243) shows that the verb of the matrix clause agrees with the absolutive argument of the embedded clause in its noun class feature.

The argument for downward Agree goes as follows: If there is no downward Agree, Agreement in Tsez must be Upward Agree. For this to be true, the goal of agreement would have to be in the matrix clause. Polinsky and Potsdam (2001) argue explicitly that there is no process of prolepsis or (c)overt raising that provides such a goal. If there is no goal in the matrix clause, the matrix verb must agree with the argument in the embedded clause. This agreement must be downward. Thus, there is a need for downward Agree.

In order to prove such conclusions wrong, proponents of Upward Agree have to show that the assumptions of the analysis above are wrong. Not only would this require a reanalysis of the data in Tsez, but of every other case ever brought forward in the literature. This is in principle possible, but it appears to be highly unlikely that all these different reanalyses can be reduced to the same idea and that the alternative downward Agree approaches can be shown to be wrong.

13.2. Yes to Upward Agree

In this type of argument, it is shown that there must be an Agree operation that applies downward.

A standard argument of this type comes from concord phenomena, such as sequence of tense in Dutch or English (Zeijlstra (2012)).

(244) a. John said Mary was ill

13. *The Direction of Agree*

- b. Jan zei dat Marie ziek was
 John said that Mary ill was
 ‘John said Mary was ill’

In (244), the tense of the embedded verb depends on the tense of the matrix verb. Note that some kind of feature checking/matching process is needed. Therefore, it is plausible that this dependency is simply an Agree dependency.

The argument for upward Agree goes like this: If there is no upward Agree, sequence of Tense must be downward Agree. If sequence of tense were downward Agree, the tense of the matrix verb would depend on the tense of the embedded verb, which is not the case: it is the higher verb’s tense that makes a semantic contribution rather than the tense of the lower verb (Kratzer (1998)). Therefore, the direction of the dependency should not be downward, but upward.

Similar to the first type of argument, proponents of downward Agree would have to show that all these apparent cases of upward Agree can be reanalysed as downward or that they are not Agree. This possible, but very cumbersome given that a bidirectional Agree operation is plausible.

At this point, I would like to mention that there have been proposals as how probe and goal can be redefined in an Agree relation in order to match directionality requirements (see for example Bošković (2011); Heck and Himmelreich (2017)). Such redefining is always possible of course, but it also sacrifices the intuitive understanding of probe and goal, namely that the probe is dependent on the goal.

Finally, it should be noted that both arguments for the existence of downward Agree and arguments for the existence of upward Agree can be used to argue for

bidirectional Agree. Put differently, a bidirectional Agree operation can derive both cases like long-distance agreement in Tsez and cases like sequence of tense in Dutch.

13.3. No to Upward Agree

This type of argument is different from the two discussed above. The aim of such arguments is to show that there cannot be an Agree operation that applies upward. If there were an additional upward Agree operation, the respective data could not be derived.

A standard argument of this type comes from anti-agreement effects (AAE), as can be observed in Berber (Ouhalla (1993), recently discussed in Georgi (2014)).

- (245)
- a. zri-n imhdarn Mohand
saw-3PL students Mohand
'The students saw Mohand.'
 - b. man tamghart ay yzrin/*t-zra Mohand
which woman C see.PARTCP/3SG.FEM-saw Mohand
'Which woman saw Mohand?'
 - c. man tamghart ay nna-n qa t-zra Mohand
which woman C said-3PL that 3SG.FEM-saw Mohand
'Which woman did they say saw Mohand?'

In (245-a), the verb shows full agreement with an argument that it c-commands. In (245-b), the verb cannot agree with an argument that it is c-commanded by and that is part of the same clause. In (245-c), the verb shows full agreement with an argument that it is c-commanded by and that is not part of the same clause.

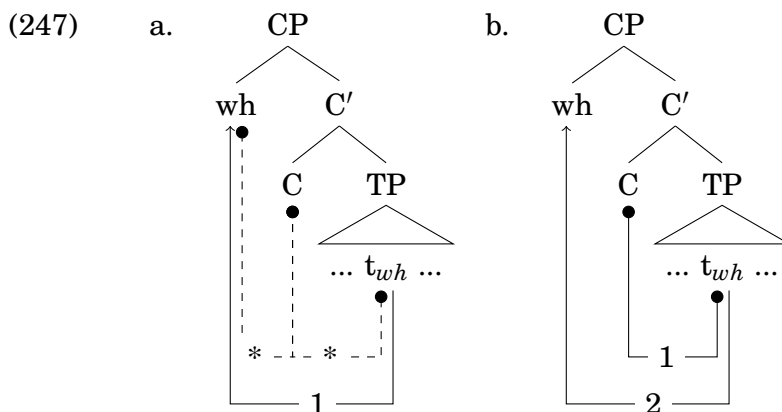
These data are analyzed in Georgi (2014) based on the following assumptions: The first assumption is that agreement on the verb is mediated by a ϕ -feature on

13. The Direction of Agree

C. Furthermore, agreement and movement features on C are ordered as in (246). Finally, there is a distinction between features triggering criterial movement and features triggering intermediate movement.

$$(246) \quad [\text{wh}] < [\phi] < [\text{EF}]$$

Based on these assumption, the argument against Upward Agree goes as follows: If criterial wh-movement applies in (245-b), the wh-phrase is moved to Spec-CP. At the time when Agree applies, the wh-phrase is not in the c-command domain of C. If upward Agree were to exist, agreement between C and the wh-phrase in Spec-CP would be possible and (245-b) should be grammatical. Since this is not the case, upward Agree cannot exist. In (245-c), Agree can apply before movement. Therefore, the wh-phrase is still c-commanded by C when Agree applies. Based on these assumptions, the conclusion has to be that there is no upward Agree (in Berber). The structure in (247) depict the difference between (245-b) and (245-c).



This argument potentially causes a problem for the bidirectional Agree approach. In order to show that the conclusion of the argument does not hold, I will reanalyze the

data. Note that in order to prove the conclusion wrong, I only have to provide one possible analysis that includes upward Agree.

As it stands, the analysis in Georgi (2014) does not provide an explanation for why subject extraction does not show any properties of movement, such as island effects (248), weak crossover effects (249) and reconstruction effects (250).

(248) *absence of island effects* (Elouazizi (2005: 126))

- a. sqssa-n [ma y-w]a Jamal lktab i
ask.PERF-3PL.MAASC whether 3SG.MASC-give.PERF J. book to
w-arba]
CS-boy
'They asked whether Jamal gave the book to the boy.'
- b. man lktab₂ ixef sqssa-n [ma D Jamal₁ i
which book about ask.PERF-3PL.MASC whether COP J. who
*(θ) y-w]i-n t₁ t₂ i w-arba]
it PRT-give.PERF-PRT to CS-boy
'Which book did they wonder whether it is Jamal who gave it to the
boy?'

(249) *absence of weak crossover effects* (Elouazizi (2005: 127))

(D) [ymas n kur arba]_j i g [vP i-texsse-n
COP mother-his of every boy who PRT-love.IMPERF.PRT
t_j memi-s_j]
son-his/her

'It is the mother of every boy who loves her son.'

(250) *absence of reconstruction* (Elouazizi (2005: 127))

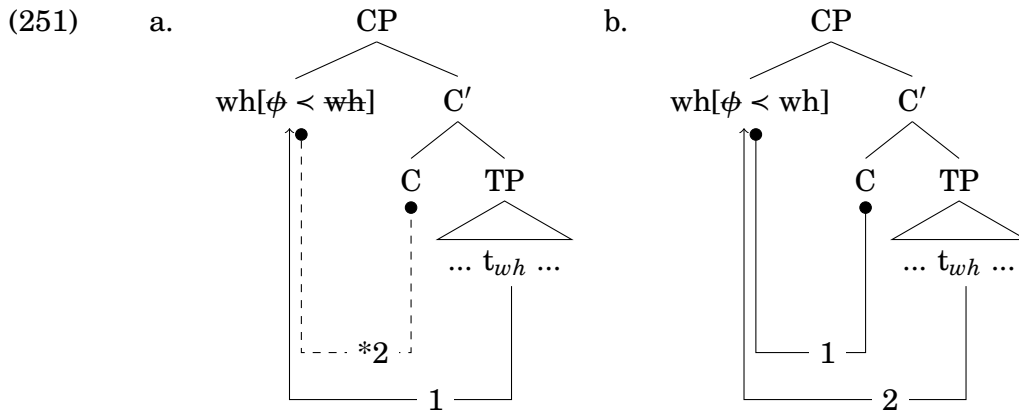
- a. ð ixefiness_{i/j} ixef y-siwrr Muhand_j ag Omar_i
COP himself about 3SG.MASC-talk.PERF Muhand with Omar
'It is himself that Muhand talked with Omar about.'

13. The Direction of Agree

- b. δ Muhand_j i g y-siwrr-n ag Omar_i x
 COP Muhand who PRT-talk.PERF-PRT with Omar about
 ixefiness_{i/*j}
 himself
 ‘It is Muhand who talked with Omar about himself.’

If these data are taken to be signs that the *wh*-phrase has not moved to Spec-CP, the anti-agreement effect might not be due to an interaction of movement and agreement. This in turn means that we cannot base an argument against upward Agree on the analysis by Georgi (2014).

If movement were to be involved, anti-agreement effects could also be reanalyzed by imposing featural restrictions on the goal of agreement: Once the *wh*-feature on a subject has been targeted, the ϕ -features are deactivated and are no longer available for Agree, as indicated in (251). The intuition behind this idea is that argument licensing has to be complete once operator features are targeted.



In both structures in (251), movement applies before ϕ -agreement. In (251-a), movement checks the *wh*-feature. Given the featural restriction, ϕ -agreement is not possible afterwards. This derives anti-agreement in an analysis based on Upward

Agree. (251-b) shows the case of long-distance *wh*-movement, where the embedded C does not check the *wh*-feature. Then, ϕ -agreement is predicted to be possible.

In conclusion, an analysis that allows upward Agree is possible for Berber. Thus, this argument against bidirectional Agree is not valid.

13.4. No to Downward Agree

The last type of argument, I want to review here is the reverse of the third type: Here, it is argued that there cannot be an Agree operation that applies downward.

A standard argument of this type comes from partial agreement, as it can be observed in Arabic (Aoun et al. (1994), discussed in Bjorkman (2014)).

- (252) a. ʔal-ʔawlaad-u naam-uu/*naam-a .
 the-children-NOM slept-3PL/*slept-MASC.3SG
 ‘The children slept.’
- b. naam-a/*naam-uu l-ʔawlaad-u .
 slept-MASC.3SG/*slept-3PL the-children-NOM
 ‘The children slept.’

The picture looks like the reverse of the anti-agreement effect in Berber: ϕ -agreement is only possible if the subject *c*-commands the ϕ -probe. The data are captured as long as there is no downward agreement (Bjorkman (2014); Zeijlstra (2012)).

As Preminger (2015) notes (judging from data in Benmamoun and Lorimor (2006)), the facts do not show reverse anti-agreement, but rather partial agreement because agreement in gender is possible. In (253), the verb is always marked for feminine gender independent of the *c*-command relations.

13. The Direction of Agree

- (253) a. t-taalibaat-u ?akal-na/*?akal-at
the-student.FEM.PL-NOM ate-FEM.3PL/*ate-FEM.3SG
'The students ate.'
- b. ?akal-at/*?akal-na t-taalibaat-u
ate-FEM.3SG/*ate-FEM.3PL the-student.FEM.PL-NOM
'The students ate.'

Based on this observation, Preminger (2015) argues that an analysis based on downward Agree is possible: First, it has to be assumed that partial agreement in verb-subject-structures such as (253-b) is due to the fact that there is an intervener or a locality boundary between the verb, which probes for ϕ -features and the subject, which is the goal of ϕ -agreement. Furthermore, ϕ -features do not Agree as a bundle, but separately in the order PERSON < NUMBER < GENDER. If it is the number feature's turn to Agree, number agreement does something to the structure to get rid of the intervention or the locality problem. Afterwards gender agreement can apply.

In subject-verb-structures, such as (253-a), the subject has moved passed the intervener/ locality boundary and can therefore fully agree with the verb. After Agree has applied, the subject moves to its target position. This analysis does not explicitly disallows upward Agree but rather shows that upward Agree is not necessary. Thus the argument against downward Agree does not hold.

Finally, it should be mentioned that an analysis based on bidirectional Agree is at least equally, if not even more plausible. All else being equal, the subject in a bidirectional Agree approach does not move to an intermediate position, but directly to its target position in which it c-commands the verb and enters into an Upward Agree relation with the verb.

13.5. Interim Summary

In this chapter, I have shown that there are four types of arguments for the direction of Agree. Arguments of the type that either of the two directions of Agree must exist are trivially reconcilable with a bidirectional definition of Agree and difficult to be proven wrong by proponents of unidirectional Agree, since it would have to be shown that a specific analysis is not possible which is in general a difficult thing to do.

Arguments of the type that a certain direction of Agree must not exist are potentially dangerous for bidirectional Agree. However, in order to falsify the validity of such arguments, it suffices to find one possible account to the respective data that includes the illicit direction of Agree. Thus, such argument are probably not as strong as they first seem to be. Usually, two strategies can be used to debunk such arguments: First, a closer look at the grammar of the specific language might reveal that the picture is more complicated than it seems to be at first sight. Second, a reanalysis based on bidirectional Agree can most likely be achieved with some modifications.

Thus, it seems that both directions of Agree exist. This leaves us with two possible interpretations of the data presented in this chapter: First, Agree can in principle apply in two directions but not within the same language or construction (Baker (2008b); Georgi (2014)). Second, Agree can in principle apply in two directions, but one of the two directions is ruled out by factors of general properties of the theory of syntactic derivation in combination with specific properties of certain constructions in certain languages structure, that is, the direction of Agree is restricted by inde-

13. The Direction of Agree

pendent factors in the derivation such as activation of features, earliness, matching, intervention, or locality. In the discussion of the case matching effects, we have seen that the first possibility is most likely not true or that the parameters that determine the Agree direction must be highly complex. Instead it seems simpler, if there is just no a priori assumption about the direction of Agree.

14. Conclusion

The topic of this thesis has been the syntactic operation Agree. In each of the three parts of this thesis, I have looked at a different aspect of Agree.

In the first part (chapter 3–7), I have laid out the foundations, by arguing that Agree needs to be bidirectional and need to apply in two steps. The argument for that comes from case matching effects, as they can be observed with free relatives and parasitic gaps. The new observation about this case matching effect is that languages and constructions vary in whether they allow case mismatches or not. Looking at various languages, a number of possible patterns emerge. The summary of the patterns is repeated in (254)-(256).

14. Conclusion

(254) *Patterns of case matching*

FR \ PG	Strict	Hierarchy	Absence
Strict	German 2/ Polish 2	German 1	Greek
Absence	Polish 1	???	—

(255) *Patterns of Agree-Link*

PG \ FR	Str.		Hier.		Abs.	
Str.	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>
Abs.	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>	<div><p>SL</p><p>Diagram description: A tree structure with root α. α branches to v_1 and \dots. v_1 branches to \emptyset and v_2. \emptyset branches to α and v_2. Arrows indicate links: $\alpha \rightarrow v_1$, $\alpha \rightarrow \emptyset$, $v_1 \rightarrow \emptyset$.</p></div>	<div><p>FR</p><p>Diagram description: A tree structure with root v_1. v_1 branches to \emptyset and \dots. \emptyset branches to α and v_2. α branches to \emptyset and v_2. Arrows indicate links: $v_1 \rightarrow \emptyset$, $\emptyset \rightarrow \alpha$, $\emptyset \rightarrow v_2$.</p></div>

(256) *Patterns of Agree-Copy*

PG \ FR	Str.		Hier.		Abs.	
Str.	SL	FR	SL	FR	SL	FR
	$V_2 \rightarrow \emptyset$	$V_2 \rightarrow \alpha$	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$
	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow \emptyset$	$\alpha \rightarrow V_1$	$\alpha \rightarrow \emptyset$
	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow \emptyset$	$\alpha \rightarrow V_1$	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow \emptyset$	$\emptyset \rightarrow V_1$
	$V_1 \rightarrow \alpha$	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow \emptyset$	$\emptyset \rightarrow V_1$		
	$\alpha \rightarrow V_1$	$\emptyset \rightarrow V_1$				
	$\alpha \rightarrow \emptyset$	$V_1 \rightarrow \emptyset$				
Abs.	SL	FR	SL	FR	SL	FR
	$V_2 \rightarrow \emptyset$	$V_2 \rightarrow \alpha$	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$
	$\emptyset \rightarrow V_2$	$\alpha \rightarrow V_2$	$\emptyset \rightarrow \alpha$	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow V_1$	$\emptyset \rightarrow V_1$
	$\emptyset \rightarrow \alpha$	$\emptyset \rightarrow \alpha$	$\alpha \rightarrow V_1$	$\emptyset \rightarrow V_1$		
	$V_1 \rightarrow \alpha$	$\emptyset \rightarrow V_1$				
	$\alpha \rightarrow V_1$	$V_1 \rightarrow \emptyset$				

Importantly, there are three different patterns for free relatives, but only two patterns for parasitic gaps.

The main achievement of the analytical work in this part is that the absence of case hierarchy effects follow from the specific properties of the parasitic gap construction and need not be stipulated on top of it. In order to derive the patterns, I have assumed that a central part of both the derivations of parasitic gaps and the derivations of free relatives is Agree between the overtly shared item and an additional covert item that actually fills the second position of the shared item. This Agree relation is eventually responsible for the matching effect. Thus, case matching is an instance of the general matching condition of Agree.

The constructions themselves have the same derivation in all the languages looked at. Variation arises by varying the location of the probe features that trigger case agreement. The main assumptions that derive the patterns are the following: Agree is bidirectional, Agree is split into syntactic Agree-Link and post-syntactic Agree-Copy (Arregi and Nevins (2012)), and the order of Agree operations is regulated by general constraints about derivations, especially the bottom-up nature of derivations and the Earliness principle. With these assumptions, the patterns follow without too many additional assumptions.

As for the case hierarchy, the hierarchy is implemented in the case values directly by means of decomposition. Syncretism effects follow if the Agree operations can interact with the syncretism rules. Overall, the interaction of different morphosyntactic rules gives rise to the idea that the patterns can also be described in terms of interaction types such as bleeding and counter-bleeding (see section 4.6).

14. Conclusion

Case matching was then shown to be a convincing argument for bidirectional Agree, since other approaches, which exclude this option, face severe difficulties in deriving the patterns. In sum, the main point of part I is that bidirectional two-step Agree must exist.

In the second part of this thesis (chapter 8–10), I have dealt with the question why there is Agree at all in these constructions, that is, how Agree emerges. I have argued for the existence of an operation Create-Dep, which creates the covert item needed for parasitic gaps and free relatives out of the shared item, and which introduces probe features to create a dependency between the overt and the covert item. This new operation gives rise to the derivation of the construction-specific properties of parasitic gaps and free relatives. I have looked at different empirical observations about these constructions that hold in various languages. The properties I have looked at are listed in (257).

- (257) a. *Parasitic gaps*
- (i) island sensitivity
 - (ii) reconstruction
 - (iii) asymmetries
 - (iv) positional restrictions
- b. *Free relatives*
- (i) number agreement
 - (ii) extraction
 - (iii) extraposition
 - (iv) semantics

Even though these properties are very specific to the respective construction, I have shown that they can be derived based on the same assumptions. Thus, the main theoretical achievement of part II is that free relatives and parasitic gaps can be traced back to the same principles.

In the final part of this thesis (chapter 11–13), I have looked at Agree outside of the parasitic gaps and free relatives. Having established the existence of bidirectional two-step Agree in part I for the derivation of case matching and having used it to derive further properties of free relatives and parasitic gaps in part II, part III provided a clear definition of Agree. I have used an algorithmic approach to define Agree-Link and Agree-Copy, thereby treating the various conditions on Agree as part of the algorithm, that is, as part of the derivation. The major achievement here is that the algorithm could be defined in a way that does not involve any additional complexity for Upward Agree.

I further continued to argue that bidirectional Agree is also superior to unidirectional Agree since it is compatible with most agreement phenomena. The argumentation was based on a review of different arguments for unidirectional Agree. I have shown that none of these arguments convincingly shows that bidirectional Agree must be ruled out.

In sum, in this thesis I have argued for a bidirectional two-step Agree operation by carrying out an in-depth study of case matching in free relatives and parasitic gap constructions.

Bibliography

- Abels, Klaus (2012): *Phases: an essay on cyclicity in syntax*. de Gruyter.
- Abney, Steven Paul (1987): The English Noun Phrase in its Sentential Aspect. PhD thesis, Indiana University.
- Agbayani, Brian (1998): Feature Attraction and Category Movement. PhD thesis, University of California, Irvine.
- Agbayani, Brian and Masao Ochi (2007): Split Lexical Insertion in Parasitic Gap Constructions. Ms. California State University, Fresno and Osaka University.
- Alexiadou, Artemis and Spyridoula Varlokosta (1997): 'The syntactic and semantic properties of free relatives in Modern Greek', *ZAS Papers in Linguistics* **5**, 1–31.
- Alexiadou, Artemis and Spyridoula Varlokosta (2007): On the structure and matching effects of free relatives in Greek. In: A. Alexiadou, ed., *Studies in the Morpho-Syntax of Greek*. Cambridge Scholar Press, Cambridge, pp. 222–250.
- Androulakis, Anna (2001): 'Clitics and Doublind in Greek', *Reading Working Papers in Linguistics* **5**, 85–111.

Bibliography

- Aoun, Joseph, Elabbas Benmamoun and Dominique Sportiche (1994): 'Agreement, Word Order, and Conjunction in Some Varieties of Arabic', *Linguistic Inquiry* **25**, 195–220.
- Arregi, Karlos and Andrew Nevins (2012): *Morphotactics: Basque Auxiliaries and the Structure of Spellout*. Springer, Dordrecht.
- Assmann, Anke (2010): Parasitic Gaps in Derivational Grammar. Master's thesis, Universität Leipzig, Leipzig.
- Assmann, Anke (2012): Deriving Parasitic Gaps by Fission and Fusion. In: E. Boone, K. Linke and M. Schulpen, eds, *Proceedings of ConSOLE XIX (2011, Groningen)*. pp. 49–75.
- Assmann, Anke (2013): Three Stages in the Derivation of Free Relatives. In: F. Heck and A. Assmann, eds, *Rule Interaction in Grammar (Linguistische Arbeitsberichte 90)*. Universität Leipzig, Leipzig, pp. 203–245.
- Baker, Mark C. (2008a): The macroparameter in a microparametric world. In: T. Biberauer, ed., *The Limits of Syntactic Variation*. John Benjamins, Amsterdam, pp. 351–373.
- Baker, Mark C. (2008b): *The Syntax of Agreement and Concord*. Cambridge University Press.
- Béjar, Susana and Milan Řezáč (2009): 'Cyclic Agree', *Linguistic Inquiry* **40**(1), 35–73.

- Benmamoun, Elabbas and Heidi Lorimor (2006): 'Featureless expressions: when morphophonological markers are absent.', *Linguistic Inquiry* **37**, 1–23.
- Bennis, Hans and Teun Hoekstra (1985): 'Gaps and Parasitic Gaps', *The Linguistic Review* **4**, 29–87.
- Bjorkman, Bronwyn and Hedde Zeijlstra (2014): Upward Agree is superior. Ms., [lingbuzz/002350](#).
- Bondaruk, Anna (1996): 'On parasitic gaps in Polish', *Papers and studies in contrastive linguistics* **31**, 111–125.
- Borer, Hagit (1984): *Parametric syntax: case studies in Semitic and Romance languages*. Foris, Dordrecht.
- Bošković, Željko (2011): On unvalued uninterpretable features. In: *Proceeding of NELS 39*. pp. 109–120.
- Brattico, Pauli (2009): 'Long-Distance Case Assignment and the Theory of Phases', *Biolinguistica Fennica Working Papers* **1**, 79–104.
- Bresnan, Joan (1973): 'Headless' Relatives. Ms., MIT.
- Bresnan, Joan and Jane Grimshaw (1978): 'The Syntax of Free Relatives in English', *Linguistic Inquiry* **9**(3), 331–391.
- Browning, Marguerite (1987): Null Operator Constructions. PhD thesis, MIT, Cambridge, Massachusetts.
- Bury, Dirk and Ad Neeleman (1999): Projection of Free Relatives. Ms., University College London.

Bibliography

- Caha, Pavel (2009): The Nanosyntax of Case. PhD thesis, University of Tromsø, Tromsø.
- Campos, Hector (1991): Silent objects and subjects in Spanish. *In*: H. Campos and F. Martinez-Gil, eds, *Current studies in Spanish linguistics*. Georgetown University Press, Washington D.C., pp. 117–141.
- Caponigro, Ivano (2002): Free Relatives as DPs with Silent D and a CP Complement. *In*: V. Samiian, ed., *Proceedings of WECOL 2000*. California State University, Department of Linguistics, Fresno, pp. 140–150.
- Caponigro, Ivano (2003): Free Not to Ask: On the Semantics of Free Relatives and Wh-Words Cross-linguistically. PhD thesis, University of California Los Angeles, Los Angeles, Californien.
- Carstens, Vicki (2016): ‘Delayed Valuation: a reanalysis of “upwards” complementizer agreement and the mechanics of case’, *Syntax* pp. 1–42.
- Chandra, Pritha (2011): *(Dis)Agree*. Cambridge Scholars Publishing.
- Chomsky, Noam (1973): Conditions on Transformations. *In*: S. Anderson and P. Kiparsky, eds, *A Festschrift for Morris Halle*. Academic Press, New York, pp. 232–286.
- Chomsky, Noam (1977): On Wh-Movement. *In*: P. W. Culicover, T. Wasow and A. Akmajian, eds, *Formal Syntax*. Academic Press, New York.
- Chomsky, Noam (1982): *Some Concepts and Consequences of the Theory of Government and Binding*. Vol. 6 of *Linguistic Inquiry Monographs*, MIT Press.

- Chomsky, Noam (1986): *Barriers*. Vol. 13 of *Linguistic Inquiry Monograph*, MIT Press, Cambridge, Massachusetts.
- Chomsky, Noam (1995): *The Minimalist Program*. MIT Press, Cambridge, Massachusetts.
- Chomsky, Noam (2000): Minimalist Inquiries: The Framework. In: R. Martin, D. Michaels and J. Uriagereka, eds, *Step by Step. Papers in Minimalist Syntax in Honor of Howard Lasnik*. MIT Press, Cambridge, chapter 3, pp. 89–155.
- Chomsky, Noam (2001): Derivation by Phase. In: M. Kenstowicz, ed., *Ken Hale: A Life in Language*. MIT Press, Cambridge, Massachusetts, pp. 1–52.
- Chomsky, Noam (2008): On Phases. In: R. Freidin, C. P. Otero and M. L. Zubizarreta, eds, *Foundational issues in linguistic theory: essays in honor of Jean-Roger Vergnaud*. MIT Press, Cambridge, chapter 6, pp. 133–166.
- Cinque, Guglielmo (1990): *Types of A'-dependencies*. MIT Press, Cambridge, Massachusetts.
- Citko, Barbara (2004): 'On Headed, Headless, and Light-Headed Relatives', *Natural Language and Linguistic Theory* **22**, 95–126.
- Citko, Barbara (2005): 'On the nature of Merge, External Merge, Internal Merge and Parallel Merge', *Linguistic Inquiry* **36**(4), 475–496.
- Citko, Barbara (2013): Size Matters: Multidominance and DP Structure in Polish. Talk at the 44th Poznan Linguistic Meeting.

Bibliography

- Collins, Chris (2002): Eliminating Labels. *In*: S. D. Epstein and T. D. Seely, eds, *Derivation and Explanation in the Minimalist Program*. Blackwell, Oxford, pp. 42–64.
- Contreras, Heles (1984): ‘A Note on Parasitic Gaps’, *Linguistic Inquiry* **15**(4), 698–701.
- Daalder, Saskia (1977): ‘Over relativie zinnen met ingesloten antecedent’, *Spektator* **7/8**, 401–407.
- Daskalaki, Evangelia (2011): ‘Case Mis-Matching as Case Stranding’, *University of Pennsylvania Working Papers in Linguistics* (Proceedings of the 34th Annual Penn Linguistics Colloquium) **17**(1), 77–86.
- Donati, Caterina and Carlo Cecchetto (2011): ‘Relabeling Heads: A Unified Account for Relativization Structures’, *Linguistic Inquiry* **42**(4), 519–560.
- Elouazizi, Noureddine (2005): Anti-Agreement Effects as (Anti-)connectivity. Proceedings of WCCFL 24.
- Engdahl, Elisabet (1983): ‘Parasitic Gaps’, *Linguistics and Philosophy* **6**, 5–34.
- Engdahl, Elisabet (1985): ‘Parasitic Gaps, Resumptive Pronouns, and Subject Extractions’, *Linguistics* **23**, 3–44.
- Epstein, Sam, Erich Groat, Ruriko Kawashima and Hiratsugu Kitahara (1998): *Derivational Approach to Syntactic Relations*. Oxford University Press, New York and Oxford.
- Fanselow, Gisbert (1987): *Konfigurationsalität*. Narr, Tübingen.

- Fanselow, Gisbert (1993): 'Die Rückkehr der Basisgenerierer', *Groninger Arbeiten zur germanistischen Linguistik* **36**, 1–74.
- Felix, Sascha (1985): Parasitic Gaps in German. In: W. Abraham, ed., *Erklärende Syntax des Deutschen*. Stauffenburg, Tübingen, pp. 173–200.
- Frampton, John and Sam Gutman (2000): Agreement in Feature Sharing. Ms., Northeastern University.
- Franks, Steven (1993): 'On Parallelism in Across-the-Board Dependencies', *Linguistic Inquiry* **24**, 509–529.
- Georgi, Doreen (2014): Opaque Interactions of Merge and Agree: On the Nature and Order of Elementary Operations. PhD thesis, Leipzig University, Leipzig.
- Gračanin-Yuksek, Martina (2008): 'Free Relatives in Croatian: An Argument for the Comp Account', *Linguistic Inquiry* **39**, 275–294.
- Graf, Thomas (2013): Local and Transderivational Constraints in Syntax and Semantics. PhD thesis, University of California, Los Angeles.
- Grewendorf, Günther and Erich Groat (2013): Labeling and Agree in Free Relatives. Ms., Goethe University Frankfurt.
- Grimshaw, Jane (1977): English Wh-Constructions and the Theory of Grammar. PhD thesis, University of Massachusetts, Amherst.
- Groos, Anneke and Henk van Riemsdijk (1981): Matching Effects with Free Relatives: A Parameter of Core Grammar. In: A. Belletti, L. Brandi and L. Rizzi, eds, *Theory*

Bibliography

- of Markedness in Generative Grammar: Proceedings of the 1979 GLOW Conference*.
Scuola Normale Superiore, Pisa, pp. 171–216.
- Grosu, Alexander (1996): ‘The Proper Analysis of “Missing-P” Free Relative Constructions’, *Linguistic Inquiry* **27**(2), 257–293.
- Grosu, Alexander (2003): ‘A Unified Theory of ‘Standard’ and ‘Transparent’ Free Relatives’, *Natural Language and Linguistic Theory* **21**(2), 247–331.
- Grosu, Alexander and Fred Landman (1998): ‘Strange Relatives of the Third Kind’, *Natural Language Semantics* **6**, 125–170.
- Haegeman, Liliane (1984): ‘Parasitic gaps and adverbial clauses’, *Journal of Linguistics* **20**, 229–232.
- Haegeman, Liliane (1994): *Introduction to Government and Binding Theory*. 2 edn, Blackwell Publishing, Oxford.
- Halle, Morris and Alec Marantz (1993): Distributed Morphology and the Pieces of Inflection. In: K. Hale and S. J. Keyser, eds, *The View from Building 20*. MIT Press, Cambridge Massachusetts, pp. 111–176.
- Harbert, Wayne (1983): ‘On the Nature of the Matching Parameter’, *The Linguistic Review* **2**, 237–284.
- Heck, Fabian and Anke Himmelreich (2017): ‘Opaque Intervention’, *Linguistic Inquiry* **48**, 47–97.
- Heck, Fabian and Gereon Müller (2007): Extremely Local Optimization. In: E. Brain-

- bridge and B. Agbayani, eds, *Proceedings of WECOL 26*. California State University, Fresno, pp. 170–183.
- Heck, Fabian and Malte Zimmermann (2004): DPs as Phases. Ms., Universität Leipzig and HU Berlin.
- Himmelreich, Anke (2016): Variation in Case Matching Effects: Evidence for Two-Step Bidirectional Agree. *In*: C. Hammerly and B. Prickett, eds, *Proceedings of NELS 46*. pp. 81–94.
- Hiraiwa, Ken (2001): Multiple Agree and the Defective Intervention Constraint in Japanese. *In*: O. Matushansky, A. Costa, J. Martin-Gonzalez, L. Nathan and A. Szczegielniak, eds, *Proceedings of the HUMIT 2000*. Vol. 40 of *UCLA Working Papers in Linguistics*, University of California, Los Angeles, pp. 67–80.
- Hirschbühler, Paul (1976): Two Analyses of Free Relatives in French. *In*: A. Ford, J. Reighard and R. Singh, eds, *North East Linguistic Society (NELS) 6*. Montreal Working Papers in Linguistics, Montreal, pp. 137–152.
- Hirschbühler, Paul and María-Luisa Rivero (1981): ‘Catalan Restrictive Relatives: Core and Periphery’, *Language* **57**, 591–625.
- Hornstein, Norbert (1999): ‘Movement and Control’, *Linguistic Inquiry* **30**(1), 69–96.
- Huybregts, Riny and Henk van Riemsdijk (1985): Parasitic Gaps and ATB. *In*: S. Berman, J.-W. Choe and J. McDonough, eds, *Proceedings of the Fifteenth Annual Meeting of the North East Linguistics Society*. University of Massachusetts at Amherst, pp. 168–187.

Bibliography

- Iatridou, Sabine (1995): 'Clitics and Island Effects', *Penn Working Papers* **2**, 11–31.
- Iatridou, Sabine, Elena Anagnostopoulou and Roumyana Izvorski (2002): Some Observations about the Form and Meaning of the Perfect. *In*: M. Kenstowicz, ed., *Ken Hale: A Life in Language*. MIT Press, Cambridge, Massachusetts, pp. 189–238.
- Jacobson, Pauline (1995): On the Quantificational Force of English Free Relatives. *In*: E. Bach, E. Jelinek, A. Kratzer and B. H. Partee, eds, *Quantification in Natural Languages*. Kluwer, Dordrecht, pp. 451–486.
- Kasai, Hironobu (2008): A Multiple Dominance Approach to Parasitic Gaps. Ms., University of Kitakyushu.
- Kathol, Andreas (2001): Nonexistence of Parasitic Gaps in German. *In*: P. Culicover and P. Postal, eds, *Parasitic Gaps*. MIT Press, Cambridge, Massachusetts, pp. 315–338.
- Kayne, Richard (1983): 'Connectedness', *Linguistic Inquiry* **14**, 223–250.
- Kayne, Richard (1984): *Connectedness and Binary Branching*. Foris, Dordrecht.
- Kearney, Kevin (1983): 'Governing Categories', Ms. University of Connecticut.
- Keine, Stefan (2010): *Case and Agreement from Fringe to Core: A Minimalist Approach*. Vol. 536 of *Linguistische Arbeiten*, De Gruyter, Berlin.
- Kiparsky, Paul (1973): Phonological Representations. *In*: O. Fujimura, ed., *Three Dimensions of Linguistic Theory*. TEC Co., Tokyo, pp. 1–135.
- Kiparsky, Paul (1976): Abstractness, Opacity, and Global Rules. *In*: A. Koutsoudas, ed., *The Ordering and Application of Rules*. Mouton.

- Kiss, Katalin (1985): 'Parasitic Chains', *The Linguistic Review* **5**, 41–74.
- Kobele, Gregory (2006): *Generating Copies: An investigation into structural identity in language and grammar*. PhD thesis, University of California, Los Angeles.
- Koopman, Hilda (2006): Agreement Configurations: In Defense of "Spec Head". In: C. Boeckx, ed., *Agreement Systems*. John Benjamins, Amsterdam, pp. 159–200.
- Koster, Jan (1986): *Domains and Dynasties*. Foris, Dordrecht.
- Kratzer, Angelika (1998): More Structural Analogies Between Pronouns and Tenses. In: *Proceedings of SALT VII*. Cornell University, Ithaca.
- Kuroda, Sige-Yuki (1968): 'English Relativization and Certain Related Problems', *Language* **44**, 244–266.
- Larson, Richard K. (1987): "Missing Prepositions" and the Analysis of English Free Relative Clauses', *Linguistic Inquiry* **18**(2), 239–266.
- Lee, Kap-Hee (1988): *A Theory of Parasitic Gaps*. PhD thesis, Hankuk University of Foreign Studies.
- Lee, Kap-Hee (1998): 'A Minimalist Approach to Parasitic Gap Constructions', *Studies in Generative Grammar* **8**, 343–363.
- Legate, Julie Anne (2003): 'Some interface properties of the phase', *Linguistic Inquiry* **34**, 506–515.
- Levine, Robert D., Thomas E. Hukari and Michael Calcagno (2001): Parasitic Gaps in English: Some Overlooked Cases and Their Theoretical Implications. In: P. D.

Bibliography

- Culicover and P. M. Postal, eds, *Parasitic Gaps*. MIT Press, Cambridge, Mass., pp. 181–222.
- Link, Godehard (1983): The Logical Analysis of Plural and Mass Terms: A Lattice-Theoretical Approach. In: R. Bäuerle, C. Schwarze and A. von Stechow, eds, *Meaning, Use and Interpretation of Language*. de Gruyter, Berlin, pp. 302–323.
- Longobardi, Giuseppe (1984): Connectedness and Island Constraint. In: J. Gueron, H.-G. Obenauer and J.-Y. Pollock, eds, *Grammatical Representation*. Foris, Dordrecht, pp. 169–185.
- Mahajan, Anoop (1990): The A/A-bar Distinction and Movement Theory. PhD thesis, MIT, Cambridge, Massachusetts.
- Malhotra, Shiti (2009): ‘Intervention Effects and wh-movement’, *University of Pennsylvania Working Papers in Linguistics* **15**(1), 134–144.
- Manzini, M. Rita (1994): ‘Locality, Minimalism and Parasitic Gaps’, *Linguistic Inquiry* **25**(3), 481–508.
- Merchant, Jason (2006): ‘Polyvalent case, geometric hierarchies, and split ergativity’, *Chicago Linguistics Society (CLS)* **42**, 47–67.
- Michaelis, Jens (2001): On Formal Properties of Minimalist Grammars. PhD thesis, Universität Potsdam.
- Müller, Gereon (2010): ‘On Deriving CED Effects from the PIC’, *Linguistic Inquiry* **41**(1), 35–82.

- Müller, Gereon (2011): *Constraints on Displacement: A Phase-Based Approach*. Vol. 7 of *Language Faculty and Beyond*, John Benjamins, Amsterdam.
- Müller, Gereon and Wolfgang Sternefeld (1993): 'Improper Movement and Unambiguous Binding', *Linguistic Inquiry* **24**, 461–507.
- Müller, Stefan (1999): 'An HPSG-Analysis for Free Relatives in German', *Grammars* **2**, 53–105.
- Munn, Alan (1994): A Minimalist Account of Reconstruction Asymmetries. In: M. Gonzalez, ed., *Proceedings of NELS 24*. GLSA, University of Massachusetts, Amherst, Massachusetts, pp. 397–410.
- Munn, Alan (2001): Explaining Parasitic Gap Restrictions. In: P. Culicover and P. Postal, eds, *Parasitic Gaps*. MIT Press, Cambridge, Massachusetts, chapter 11, pp. 369–392.
- Nissenbaum, Jonathan W. (2000): Investigations of covert phrase movement. PhD thesis, MIT.
- Noyer, Rolf (1992): Features, Positions and Affixes in Autonomous Morphological Structure. PhD thesis, MIT.
- Nunes, Jairo (1995): The Copy Theory of Movement and Linearization of Chains in the Minimalist Program. PhD thesis, University of Maryland, College Park.
- Nunes, Jairo (2001): 'Sideward Movement', *Linguistic Inquiry* **32**, 303–344.
- Nunes, Jairo (2004): *Linearization of Chains and Sideward Movement*. MIT Press, Cambridge, Massachusetts.

Bibliography

- Ott, Dennis (2011): 'A Note on Free Relative Clauses in the Theory of Phases', *Linguistic Inquiry* **42**(1), 183–192.
- Ouhalla, Jamal (1993): 'Subject-Extraction, Negation and the Antiagreement Effect', *Natural Language and Linguistic Theory* **11**, 477–518.
- Ouhalla, Jamal (2001): Parasitic Gaps and Resumptive Pronouns. In: P. D. Culicover and P. M. Postal, eds, *Parasitic Gaps*. MIT Press, Cambridge, Mass., chapter 5, pp. 147–180.
- Partee, Barbara, Alice ter Meulen and Robert Wall (1990): *Mathematical Methods in Linguistics*. Kluwer Academic Publishers, Dordrecht.
- Pesetsky, David (1989): Language-Paticular Processes and the Earliness Principle. Ms., MIT.
- Pesetsky, David and Esther Torrego (2007): The Syntax of Valuation and the Interpretability of Features. In: S. Karimi, V. Samiiian and W. K. Wilkins, eds, *Phrasal and Clausal Architecture: Syntactic Derivation and Interpretation*. Benjamins, Amsterdam, pp. 262–294.
- Pittner, Karin (1991): Freie Relativsätze und die Kasus-hierarchie. In: E. Feldbusch, ed., *Neue Fragen der Linguistik Vol. 1*. Niemeyer, Tübingen, pp. 341–347.
- Pittner, Karin (1995): 'Regeln für die Bildung von freien Relativsätzen', *Deutsch als Fremdsprache* **32**(4), 195–200.
- Polinsky, Maria and Eric Potsdam (2001): 'Long-Distance Agreement and Topic in Tsez', *Natural Language and Linguistic Theory* **19**, 583–646.

- Postal, Paul (1993): 'Parasitic Gaps and the Across-the-Board Phenomenon', *Linguistic Inquiry* **24**, 735–754.
- Postal, Paul (1998): *Three Investigations of Extraction*. MIT Press, Cambridge, Massachusetts.
- Preminger, Omer (2015): Upwards and onwards: On the direction of valuation in phi-feature agreement. Talk at New York University.
- Preminger, Omer and Maria Polinsky (2015): Agreement and semantic concord: a spurious unification. Ms., lingbuzz/002363.
- Reuland, Eric (2001): 'Primitives of Binding', *Linguistic Inquiry* **32**(3), 439–492.
- Řezáč, Milan (2004a): Agree and Merge. Ms. University of the Basque Country.
- Řezáč, Milan (2004b): Elements of Cyclic Syntax: Agree and Merge. PhD thesis, University of Toronto.
- Riedel, Kristina (2009): The Syntax of Object Marking in Sambaa: A comparative Bantu perspective. PhD thesis, University Leiden.
- Riemsdijk, Henk van (2006): Free Relatives. In: M. Everaert and H. v. Riemsdijk, eds, *The Blackwell Companion to Syntax*. Blackwell, Oxford, pp. 338–382.
- Rooryck, Johan (1994): 'Generalized Transformations and the wh-Cycle: Free Relatives as Bare wh-CPs', *Groninger Arbeiten zur germanistischen Linguistik* **37**, 195–208.
- Ross, John (1967): Constraints on Variables in Syntax. PhD thesis, MIT, Cambridge, Mass.

Bibliography

- Safir, Ken (1987): 'The Anti-C-Command Condition on Parasitic Gaps', *Linguistic Inquiry* **18**, 678–683.
- Spyropoulos, Vassilios (2011): Case Conflict in Greek Free relatives: Case in Syntax and Morphology. In: A. Galani, G. Hicks and G. Tsoulas, eds, *Morphology and its Interfaces*. John Benjamins, Amsterdam, pp. 21–56.
- Sternefeld, Wolfgang (2006): *Syntax. Eine morphologisch motivierte generative Beschreibung des Deutschen*. Stauffenburg Verlag, Tübingen. 2 Bände.
- Suñer, Margarita (1984): 'Free Relatives and the Matching Parameter', *The Linguistic Review* **3**, 363–387.
- Svenonius, Peter (2004): On the Edge. In: D. Adger, C. de Cat and G. Tsoulas, eds, *Peripheries: Syntactic Edges and their Effects*. Kluwer, Dordrecht, pp. 261–287.
- Taraldsen, Knut (1981): The Theoretical Interpretation of a Class of Marked Extractions. In: A. Belletti, L. Brandi and L. Rizzi, eds, *Theory of Markedness in Generative Grammar*. Scuola Normale Superiore, pp. 475–516.
- Toosarvandani, Maziar and Coppe van Urk (2014): The syntax of nominal concord: What *ezafe* in Zazaki shows us. In: H.-L. Huang, E. Poole and A. Rysling, eds, *Proceedings of NELS 43*. The City University of New York, pp. 221–234.
- Trommer, Jochen (2002): Modularity in OT-Morphosyntax. In: G. Fanselow and C. Féry, eds, *Resolving conflicts in grammars: optimality theory in syntax, morphology, and phonology*. Buske, Hamburg, pp. 83–118.

- Ura, Hiroyuki (2007): 'Long-Distance Case-Assignment in Japanese and Its Dialectal Variation', *Gengo Kenkyu* **131**, 1–43.
- Vainikka, Anne and Pauli Brattico (2014): 'The Finnish Accusative: Long Distance Case Assignment Under Agreement', *Linguistics* **52**, 73–124.
- Vogel, Ralf (2001): Case Conflict in German Free-Relative Constructions: An Optimality-Theoretic Treatment. In: G. Müller and W. Sternefeld, eds, *Competition in Syntax*. Mouton de Gruyter, Berlin, pp. 341–375.
- Webelhuth, Gert (1992): *Principles and Parameters of Syntactic Saturation*. Oxford University Press, Oxford.
- Williams, Edwin (1990): 'The ATB theory of parasitic gaps', *The Linguistic Review* **6**, 265–279.
- Zaenen, Annie, Elisabet Engdahl and Joan M. Maling (1981): 'Resumptive Pronouns can be Syntactically Bound', *Linguistic Inquiry* pp. 679–682.
- Zeijlstra, Hedde (2012): 'There is only one way to agree', *The Linguistic Review* **29**, 491–539.

Bibliography

Selbstständigkeitserklärung

und Erklärung über frühere Promotionsversuche

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Ich habe nicht die Hilfe eines Promotionsberaters in Anspruch genommen. Die Arbeit wurde zuvor weder im Inland noch im Ausland in gleicher oder ähnlicher Form einer anderen Institution vorgelegt. Sie ist weder Bestandteil eines ruhenden Verfahrens noch wurde sie in einem gleichartigen Promotionsverfahren als endgültig nicht bestanden erklärt. Die Arbeit ist vorher auch noch nicht veröffentlicht worden. Ich erkläre, dass ich bisher noch keine Promotionsversuche unternommen habe.

Leipzig, 10.07.2017

Anke Himmelreich