

Variation in Case Matching Effects: Evidence for Two-Step Bidirectional Agree

Anke Himmelreich
 Leipzig University
 anke.assmann@uni-leipzig.de

1 Introduction

The following study deals with case matching effects as they occur with free relatives and parasitic gaps. It is shown that the case matching condition for both constructions is subject to cross-linguistic variation. Interestingly, Polish is the mirror image of German: 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.

- (1) *Free relatives*
 I'll buy what you are selling.
- (2) *Parasitic gaps*
Which article did you file without reading?
- (3) *Case mismatches*

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

The cross-linguistic variation shows that each construction can in principle be subject to a matching condition. Thus, an analysis is required that allows case matching in both constructions, but can still account for the distribution of case matching effects as shown in (3).

Claim:

The variation between Polish and German as well as the variation between the two constructions can be accounted for if case matching is modeled as Agree and if Agree is bidirectional and subject to variation. It is shown that other possible approaches to matching including an account based on a unidirectional Agree operation fails to fully account for the pattern.

2 Data

2.1 German (Pittner (1995); Vogel (2001); Fanselow (1993); Kathol (2001))

(4) *Parasitic gaps*

Matr/PG	Acc	Dat	Gen
Acc	✓	*	*
Dat	*	✓	✓(syn)
Gen	*	✓(syn)	✓

(5) *Free relatives*

Matr/FR	Acc	Dat	Gen
Acc	✓	✓(dat)	✓(gen)
Dat	*	✓	✓(gen)
Gen	✓(acc)	✓(dat)	✓

(6) *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}] behinderte_{acc}
 because Hans the woman instead.of to help hampered
 “because Hans hampered the woman instead of helping her”
- d. weil Hans *der_{dat}/*die_{acc} Frau [anstatt zu behindern_{acc}] half_{dat}
 because Hans the woman instead.of to hamper helped
 “because Hans hampered the woman instead of helping her”
- e. weil Hans der_{gen/dat} Verstorbenen [anstatt ein Gedicht zu widmen_{dat}] in einer
 because Hans the dead.one instead.of a poem to dedicate in a
 Gradrede gedachte_{gen}
 eulogy commemorate
 “because Hans commemorated the dead one in a eulogy instead of dedicating a poem ot her”

(7) *Free relatives: absence of 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) Maria vertraut_{dat}].
 Hans helps who ever Maria trusts
 “Hans helps whoever Maria 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.”

2.2 Polish (Citko (2013))

(8) *Parasitic gaps*

Matr/PG	Acc	Dat	Gen
Acc	✓	✓(acc)	✓(acc)
Dat	✓(dat)	✓	✓(syn)
Gen	✓(gen)	✓(syn)	✓

(9) *Free relatives*

Matr/FR	Acc	Dat	Gen
Acc	✓	*	✓(syn)
Dat	*	✓	*
Gen	✓(syn)	*	✓

(10) *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 pomagać_{dat}].
 this is girl which Jan accompanied before started 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 pomagać_{dat}].
 this is girl which Jan liked before started help
 “This is the girl Jan liked before he started to help.”
- d. To jest dziewczyna, której_{dat}/*która_{acc} Jan ufał_{dat} [zanim polubił_{acc}].
 this is girl which Jan trusted before liked
 “This is the girl Jan trusted before he got to like.”

(11) *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.”

2.3 Observations

- German and Polish are mirror images of each other: FR that lack case matching effects in German, show them in Polish, while PGs that allow case mismatches in Polish show strict matching in German.
- Free relatives and parasitic gaps are mirror images of each: if one of the two configurations show case matching effects, the other does not.
- Syncretic forms can repair violations of the case matching condition. Thus, what seems to count for matching are not the abstract Case features but the morphological form.
- *Consequence*: Under the assumption that the morphological form does not count for narrow syntax, the case matching condition cannot be a constraint solely of narrow syntax (see Trommer (2002) for the same conclusion).

3 Analysis

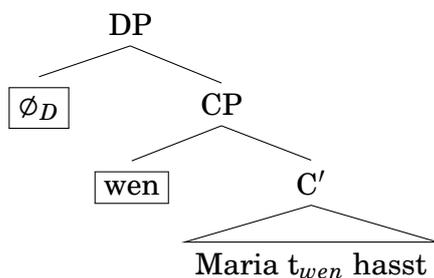
3.1 Assumptions

3.1.1 Structure

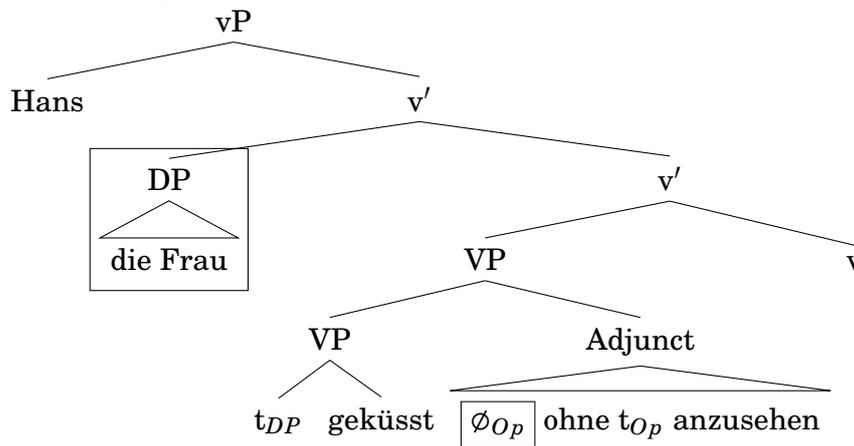
Standard structures for free relatives and parasitic gaps: In both cases there is not one item that occupies two case positions, but there is one overt item α and one covert item \emptyset (see Groos and Riemsdijk (1981) among others for FRs, Chomsky (1986) among others for PGs). For concreteness, \emptyset in FR structures is an empty D head, while PG configurations involve an empty operator.

(12) *Structures*

a. *Free relatives*



b. *Parasitic gaps*



3.1.2 Agreement between α and \emptyset

- The overt item α and \emptyset have a special syntactic bond that can be formalized as an agreement relation (cf. Assmann (2012) for PGs, Grosu (2003); Assmann (2013); Grewendorf and Groat (2013) among others for FRs). Among other features, the agreement relation also affects the case features of the two items.
- Importantly, agreement between α and \emptyset is asymmetric: One of the two acts as the probe.
- Otherwise, this special type of case agreement between the overt and the covert item works just like case assignment.

3.1.3 Case assignment

Case assignment is Agree Case assignment is copying of case features from a case assigning head onto a head that probes for case features. Due to the special agreement relation between α and \emptyset in PG and FR constructions, case features on α and \emptyset can in principle probe twice (once for the case assigning head and once for \emptyset or α respectively).

Two-Step 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 a probe and a goal. Post-syntactic Agree-Copy copies the case values from probe to goal.

Evidence for post-syntactic Agree: 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 (see (6-e), (11-e)).

Evidence for syntactic Agree: Evidence for syntactic Agree comes from the fact that the case matching pattern does not change if c-command between α and \emptyset is disrupted, e.g. by extraposition (13). If one doesn't want to give up the c-command condition of Agree entirely, and if extraposition is a syntactic process (but see Truckenbrodt (1995)), \emptyset and *wem* have to Agree prior to extraposition in syntax.

- (13) a. Hans hat [_{DP} \emptyset_D t_{CP}] gemocht_{acc} [_{CP} wen_{acc} (auch immer) Maria gehasst_{acc} hat].
 Hans has liked who ever Maria hated has
- b. Hans hat [_{DP} \emptyset_D t_{CP}] geholfen_{dat} [_{CP} wem_{dat} (auch immer) er vertraut_{dat} hat].
 Hans has helped who ever he trusted has

- c. Hans hat [_{DP} \emptyset_D t_{CP}] gemocht_{acc} [_{CP} *wen_{acc}/wem_{dat} (auch immer) Maria vertraut
 Hans has liked who ever Maria trusted
 hat_{dat}].
 has
- d. Hans hat [_{DP} \emptyset_D t_{CP}] vertraut_{dat} [_{CP} *wen_{acc}/*wem_{dat} (auch immer) Maria gemocht
 Hans has trusted who ever Maria liked
 hat_{acc}].
 has

Matching Case assignment is only successful if the case feature value of probe and goal do not conflict. By assumption, a conflict cannot arise if one of the two features is still unvalued.

Order Case probes that are lower in the structure receive their case features before higher probes, that is, post-syntactic Agree-Copy proceeds bottom-up. If a probe participates in more than one Agree-Link relation, which happens in FR and PG constructions due to the additional relation (see above), the order is free.

Directionality There is upward as well as downward agreement for case features (see Zeijlstra (2012) among others for upward Agree, Baker (2008b), Toosarvandani and van Urk (2014) among others for bidirectional Agree). The only configurational restriction is that the probe and goal are in a c-command relation with the probe c-commanding the goal or vice versa.

3.1.4 Variation

Following the Borer-Chomsky-Conjecture (Borer 1984; Chomsky 1995; Baker 2008a), the differences between Polish and German regarding case matching only concern lexical items: Concretely, Polish and German differ in whether the overt element α or the covert element \emptyset triggers case agreement in FRs and PGs.

- In German, case agreement is triggered by α : $\alpha[*\text{case:}_*]$.
- In Polish, case agreement is triggered by \emptyset : $\emptyset[*\text{case:}_*]$.

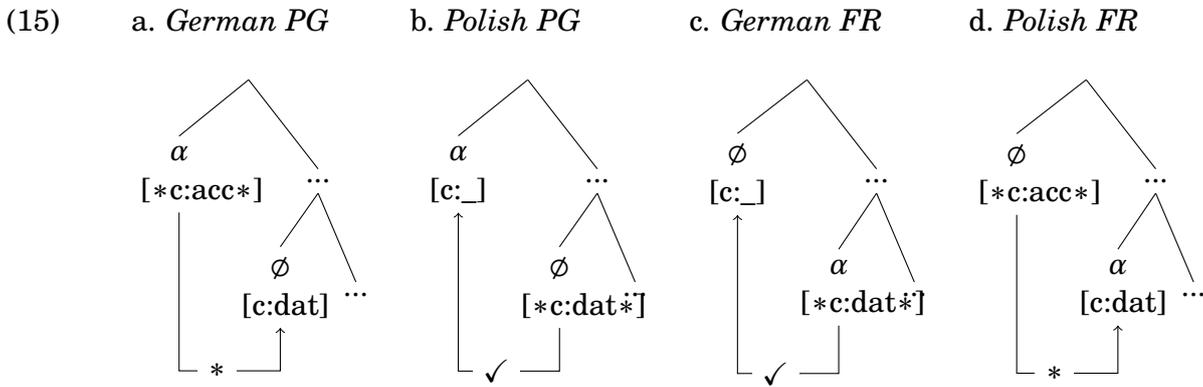
Note:

The double-probe property is depicted by two symbols: “_” stands for “I need a value”, while “* *” stands for “I want to probe a second time” (see Sternefeld (2006) for the notation * *). Thus sometimes case features need to be valued and additionally checked.

This leads to the four possible configurations shown in (15). Each configuration corresponds to one of the four patterns in (3) repeated as (14). (The configurations show the case features at the time when the case feature on α/\emptyset probes a second time.)

(14) *Case mismatches*

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



Idea in a nutshell:

- If the lower of the two elements is the probe, upward case agreement will always result 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 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.

3.2 Derivations

Note:

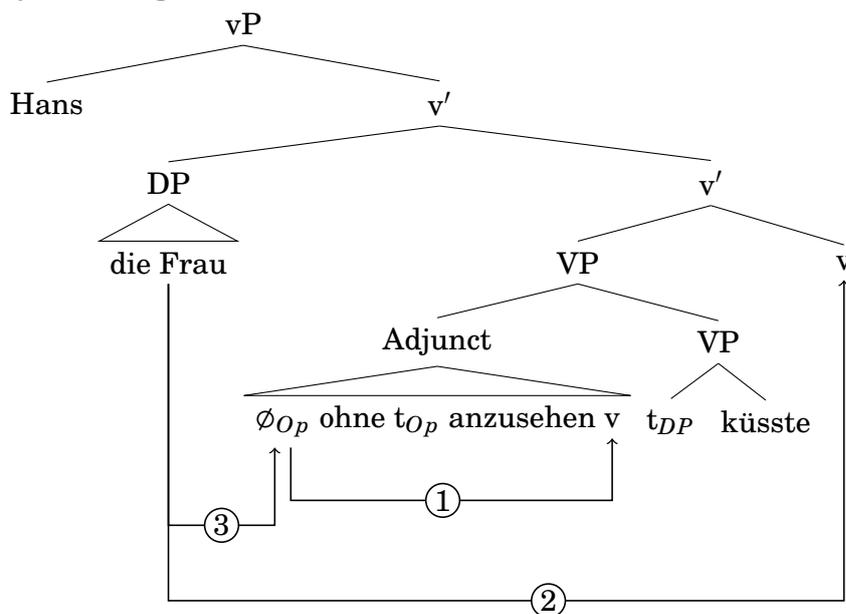
By assumption, dative case is assigned by an empty applicative head Appl. Alternatively, dative can be assigned by an empty preposition or some other functional head. Nothing hinges on that.

3.2.1 German: probe on α

Parasitic gaps: matching case

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

(17) *Syntactic Agree-Link*



Due to the bottom-up property of post-syntactic Agree-Copy, case feature valuation on \emptyset must precede valuation of the case features on *die Frau*.

(17) i. $\emptyset[c:_] \rightarrow v[c:acc]: \emptyset[c:acc]$

Next, the case feature of α (*die Frau*) is valued. Since the case feature on α is a double probe, there are two possible orders of case assignment. Both result in case matching.

(17) ii. $\alpha[*c:_*] \rightarrow v[c:acc]: \alpha[*c:acc*]$ ii. $\alpha[*c:_*] \rightarrow \emptyset[c:acc]: \alpha[*c:acc*]$
 iii. $\alpha[*c:acc*] \rightarrow \emptyset[c:acc]: \alpha[c:acc]$ iii. $\alpha[*c:acc*] \rightarrow v[c:acc]: \alpha[c:acc]$

Parasitic gaps: no matching case

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

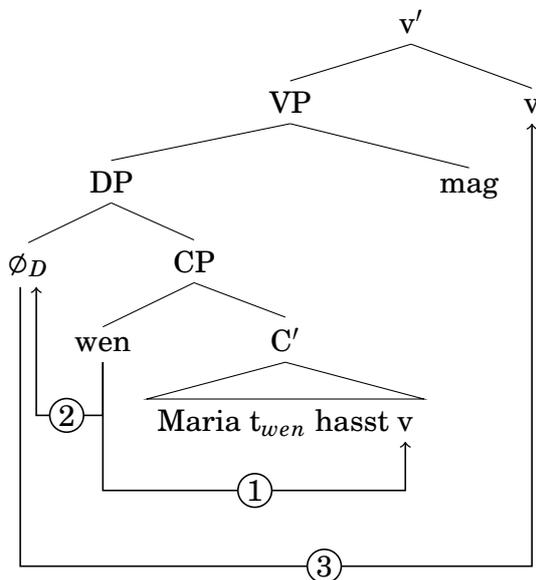
The syntactic derivation of (18) works as in (17). The difference lies in the cases being assigned. In the derivation of (18), \emptyset receives dative case in the embedded clause. Case feature valuation of *die Frau* fails because a case conflict on α will arise.

(18) i. $\emptyset[c:_] \rightarrow Appl[c:dat]: \emptyset[c:dat]$ i. $\emptyset[c:_] \rightarrow Appl[c:dat]: \emptyset[c:dat]$
 ii. $\alpha[*c:_*] \rightarrow v[c:acc]: \alpha[*c:acc*]$ ii. $\alpha[*c:_*] \rightarrow \emptyset[c:dat]: \alpha[*c:dat*]$
 iii. $\alpha[*c:acc*] \rightarrow \emptyset[c:dat]: \cancel{\downarrow}$ iii. $\alpha[*c:dat*] \rightarrow v[c:acc]: \cancel{\downarrow}$

Free relatives: matching case

(19) Hans mag_{acc}, wen_{acc} Maria hasst_{acc}.
 Hans likes who Maria hates
 “Hans likes who Maria hates.”

(20) Syntactic Agree-Link



This time, the case feature of the overt item *wen* must be valued first because it is lower in the structure. Since the case feature is a double probe, two orders are possible.

(20) i. $\alpha[*c:_*] \rightarrow v[c:acc]: \alpha[*c:acc*]$ i. $\alpha[*c:_*] \rightarrow \emptyset[c:_]: \alpha[c:_]$
 ii. $\alpha[*c:acc*] \rightarrow \emptyset[c:_]: \alpha[c:acc]$ ii. $\alpha[c:_] \rightarrow v[c:acc]: \alpha[c:acc]$

After *wen* has received its case feature value, the case feature on \emptyset can be valued.

(20) iii. $\emptyset[c:_] \rightarrow v[c:acc]: \emptyset[c:acc]$

Free relatives: no matching case

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

The syntactic derivation of (21) is similar to (20), only this time the wh-phrase receives dative case in the embedded clause. Since the probe of the additional agreement relation between the wh-phrase and the empty D head is lower than the goal, no case conflict will arise.

- | | | | | |
|------|------|---|------|---|
| (21) | i. | $\alpha[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[*c:\text{dat}*]$ | i. | $\alpha[*c:_*] \rightarrow \emptyset[c:_]: \alpha[c:_]$ |
| | ii. | $\alpha[*c:\text{dat}*] \rightarrow \emptyset[c:_]: \alpha[c:\text{dat}]$ | ii. | $\alpha[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}]$ |
| | iii. | $\emptyset[c:_] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$ | iii. | $\emptyset[c:_] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$ |

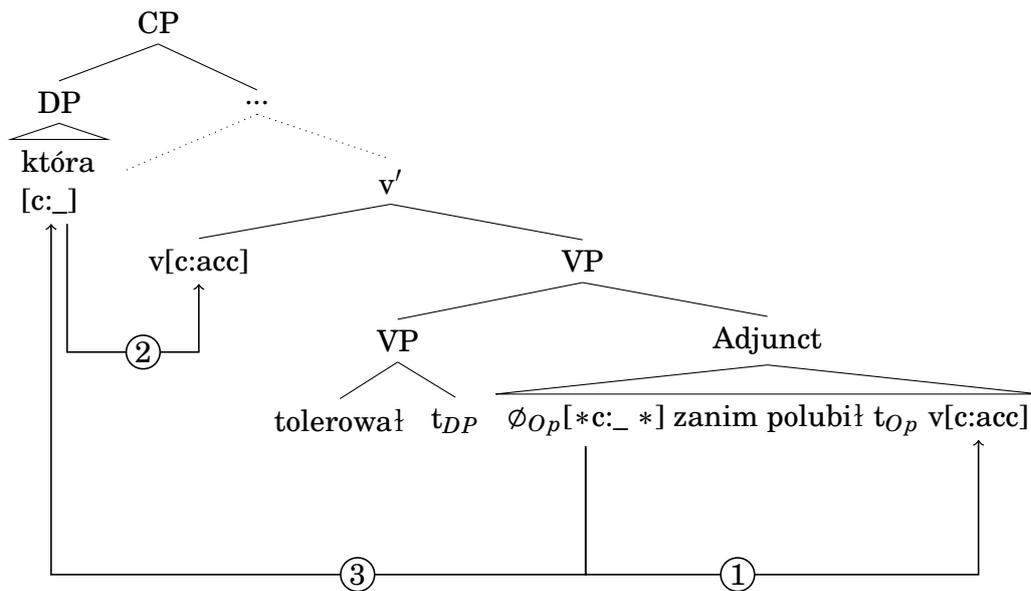
3.2.2 Polish: Probe on \emptyset

Parasitic gaps: matching case

- (22) 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 \emptyset bears the case feature that probes twice. Thus, we have upward agreement in PGs in Polish and no case conflict will arise (similar to free relatives in German).

- (23) *Syntactic Agree-Link*



- | | | | | |
|------|------|--|------|---|
| (23) | i. | $\emptyset[*c:_*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{acc}*]$ | i. | $\emptyset[*c:_*] \rightarrow \alpha[c:_]: \emptyset[c:_]$ |
| | ii. | $\emptyset[*c:\text{acc}*] \rightarrow \alpha[c:_]: \emptyset[c:\text{acc}]$ | ii. | $\emptyset[c:_] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$ |
| | iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ | iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ |

Parasitic gaps: no matching case

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

The order of case feature valuation is similar to FRs in German. Because the probe of the α - \emptyset agreement relation is lower than the goal, upward agreement will result in empty valuation. Consequently, no mismatch can arise.

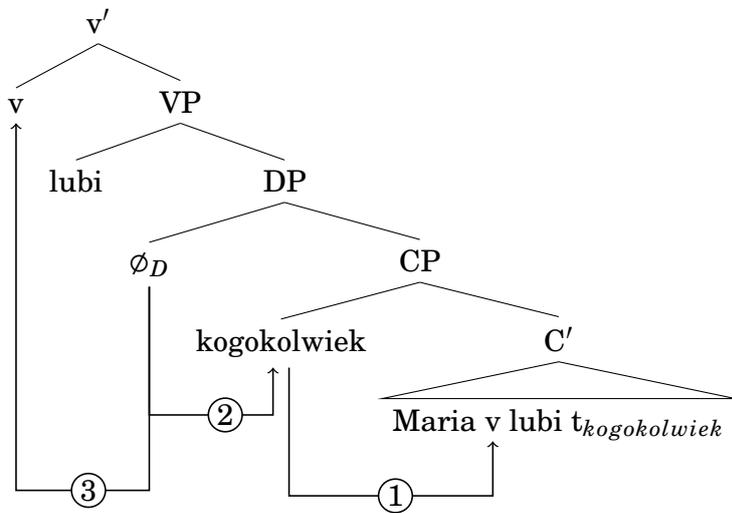
- (24)
- | | | | |
|------|--|------|---|
| i. | $\emptyset[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[*c:\text{dat}^*]$ | i. | $\emptyset[*c:_*] \rightarrow \alpha[c:_]: \emptyset[c:_]$ |
| ii. | $\emptyset[*c:\text{dat}^*] \rightarrow \alpha[c:_]: \emptyset[c:\text{dat}]$ | ii. | $\emptyset[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[c:\text{dat}]$ |
| iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ | iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ |

Free relatives: matching case

- (25) Jan *lubi_{acc}* kogokolwiek_{acc} Maria *lubi_{acc}*.
 Jan likes whoever Maria likes

In contrast to German FRs, the probe of the additional case agreement relation is higher than the goal. Thus, both the case value of α and the case value of \emptyset will have to match.

- (26) *Syntactic Agree-Link*



- (26)
- | | | | |
|------|---|------|--|
| i. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ | i. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ |
| ii. | $\emptyset[*c:_*] \rightarrow \alpha[c:\text{acc}]: \emptyset[*c:\text{acc}^*]$ | ii. | $\emptyset[*c:_*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{acc}^*]$ |
| iii. | $\emptyset[*c:\text{acc}^*] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$ | iii. | $\emptyset[*c:\text{acc}^*] \rightarrow \alpha[c:\text{acc}]: \emptyset[c:\text{acc}]$ |

Free relatives: no matching case

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

The derivation is the same as in (27). This time however, the wh-phrase receives dative case in the embedded clause. This will lead to a case conflict on the double probe \emptyset .

- (27)
- | | | | |
|------|---|------|---|
| i. | $\alpha[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}]$ | i. | $\alpha[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}]$ |
| ii. | $\emptyset[*c:_*] \rightarrow \alpha[c:\text{dat}]: \emptyset[*c:\text{dat}^*]$ | ii. | $\emptyset[*c:_*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{acc}^*]$ |
| iii. | $\emptyset[*c:\text{dat}^*] \rightarrow v[c:\text{acc}]: \zeta$ | iii. | $\emptyset[*c:\text{acc}^*] \rightarrow \alpha[c:\text{dat}]: \zeta$ |

3.3 Opacity

Question:

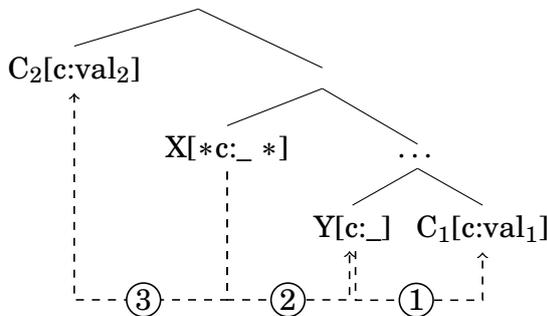
Some of the derivations above are opaque. At the surface (after step iii.), both α and \emptyset bear case feature values. But only in some of the derivations, conflicting case values on α and \emptyset cause a violation of case matching. Why?

Answer:

- In all derivations there are three case assignment relations that are important:
 1. between the lower element Y and a case assigning head C_1
 2. between the higher element X and another case assigning head C_2
 3. between X and Y
- Case matching arises if the first two case assignment relations result in feature valuation. Then the third case assignment relation will be case feature checking and the two values must be identical. Thus, case feature valuation can bleed final case checking.
- Due to the bottom-up derivation of case feature valuation, Y will receive its case feature value first. If Y is the probe in the case agreement relation between X and Y, we have upward agreement, which will always result in empty feature valuation, thus no final case feature checking, i.e., no case matching, arises. Consequently, there is no bleeding relation. We have counter-bleeding.

Bleeding:

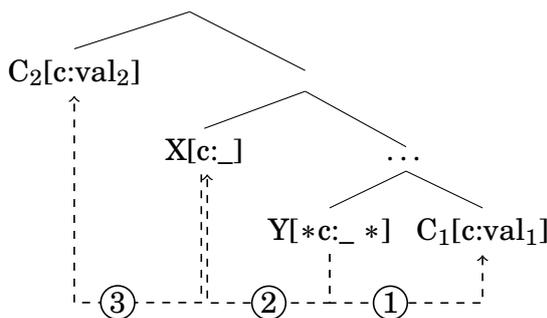
(28)



1. $Y[c:_] \Rightarrow Y[c:val_1]$
2. $X[*c:_*] \Rightarrow X[*c:val_2*]$
3. $X[*c:val_2*] \Rightarrow X[c:val_2]$ (*matching required*)

Counter-Bleeding:

(29)



1. $Y[*c:_*] \rightarrow Y[*c:val_1*]$
2. $Y[*c:val_1*] \rightarrow Y[c:val_1]$ (*empty valuation*)
3. $X[c:_] \rightarrow X[c:val_2]$ (*no matching required*)

Summary:

Upward agreement leads to counter-bleeding! (cf. Georgi (2014))

4 Alternatives

Question:

Are there any alternatives to the present account?

The number of possibilities to analyze structures, where one item seems to be a dependent of two verbs, is limited. In principle, there are three strategies:

1. *Agreement approaches (the present account):*

Postulate an extra empty category and let the empty and the overt category communicate in some way – usually, 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)).

2. *Identity approaches:*

Let the overt category be the only category and model the additional syntactic dependency differently (multidominance accounts à la Riemsdijk (2006); Kasai (2008); Citko (2005, 2013) or movement accounts (Huybregts and van Riemsdijk (1985); Bennis and Hoekstra (1985); Williams (1990); Nunes (2004); Rooryck (1994); Caponigro (2003); Donati and Cecchetto (2011); Ott (2011))).

3. *Reanalysis:*

Treat FRs and PGs differently in different languages. In some languages, FRs/PGs involve empty categories and in some, they don't.

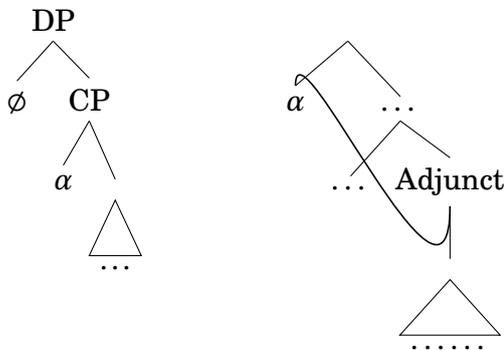
Question:

The present analysis of case matching effects is an example for how agreement approaches can deal with the pattern discussed in section 2. What about the alternatives?

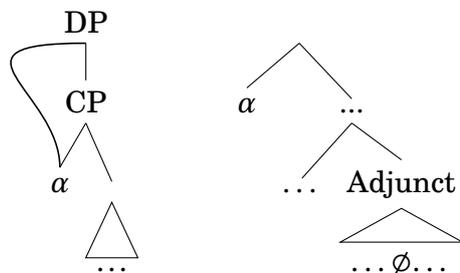
4.1 Reanalysis

Strict matching is due to identity: there is only one element that has to satisfy the case requirements of two verbs. The absence of matching effects arises if a second, empty, category is involved.

- (30) *German*
 a. *Free Relatives* b. *Parasitic Gaps*



- (31) *Polish*
 a. *Free Relatives* b. *Parasitic Gaps*



The solution is very simple but requires additional evidence that the two constructions really have different derivations in different languages.

As for Polish and German, there are no major differences concerning parasitic gaps (see Bondaruk (1996) on PGs in Polish) and free relatives:

(32) *Parasitic Gaps*

	<i>German</i>	<i>Polish</i>
island sensitivity	✓	✓
categorial restrictions	✓	✓
ban against licensing in-situ	✓	?
ban on A-movement licensing	✓	✓
tensed environments	*	✓

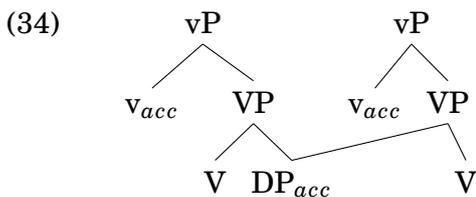
(33) *Free Relatives*

	<i>German</i>	<i>Polish</i>
lack of overt nominal head	✓	✓
wh-phrase instead of relative pronoun	✓	✓
clause with gap	✓	✓
replaceable with truth-conditionally equivalent DP or PP	✓	✓

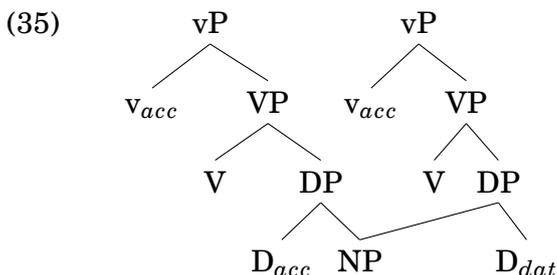
4.1.1 Citko (2013)

In Citko (2013), an analysis of variation is presented that builds on multidominance. However, only one part of the variation, namely the variation within one language, can be derived.

Strict matching occurs when a DP is subject to multidominance: The case feature that is located on D is shared between two verbs. Thus, it has to match the requirements of both verbs.



Mismatching is due to multidominance of an NP. The 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.



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.

The analysis faces problems with cross-linguistic variation: German PGs and FRs must be different from Polish PGs and FRs: In German, PGs requires sharing of a DP and FRs require sharing of an NP. This resembles a reanalysis strategy which lacks independent evidence.

A further problem 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

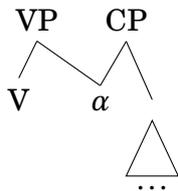
case features, the single case feature on the NP cannot meet the requirements of both D heads, that is, the violation of matching is expected to occur on the NP level.

4.2 Identity

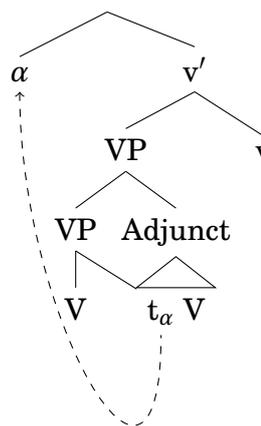
4.2.1 Multidominance

In both FRs and PGs, the overt category is shared between two verbs. Thus it is dependent on the case requirements of both verbs.

(36) *Free Relatives*



(37) *Parasitic Gaps*

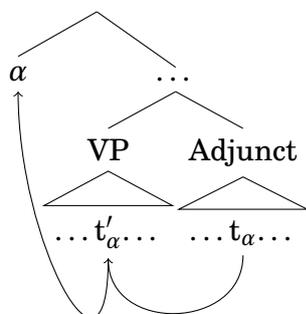


Grafting approaches have a problem explaining variation. The cross-linguistic variation can be handled by assuming that in one language a matching condition holds while in the other language, there is no matching condition. However, since abstractly, FRs and PGs have the same derivation, both constructions in one language are predicted to either show case matching effects or not. The only way out would be that the matching condition is construction-specific which predicts that there should be languages which do not show matching effects in any of the two constructions, contrary to what is reported in the literature.

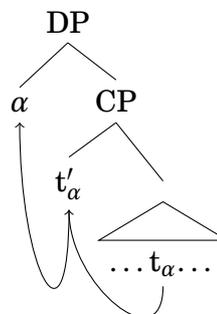
4.2.2 Movement

The overt category is merged in the embedded clause, where it receives case. It then 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 have to match the new case requirements.

(38) *Parasitic Gaps*



(39) *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 the same.

4.3 Unidirectional Agree

4.3.1 Only Upward Agree

Case assignment is Agree between a probing argument and a case assigning head (Pesetsky and Torrego 2007). Since there is evidence that case assigning heads are higher than the case probe, this case Agree must be upwards. In a unidirectional approach to Agree, upward can be the only possible direction of Agree (Zeijlstra 2012).

But then, downward case agreement, as we experience it e.g. in German parasitic gaps must be a different genuinely different syntactic process. Having excluded movement and multidominance since these processes are too rigid to account for variation, it is unclear which syntactic process is responsible.

Furthermore, this case matching shows some of the core properties of the Agree operation:

- *Locality*: Agree relations and case assignment are subject to certain locality restrictions, e.g. they do not cross a finite clause boundary (see e.g. Vainikka and Brattico 2014). This also applies to parasitic gap configurations in German (40). Examples where the empty operator and the antecedent are separated by a finite clause boundary are ungrammatical despite case matching (40-b). This follows if Agree cannot apply across finite clause boundaries. Since the probing features on the antecedent cannot be checked, the derivation crashes.

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

- *Intervention*: Agree is subject to intervention effects. This can be observed in parasitic gap configurations (see Assmann and Heck 2013).¹

- (41) *wenn jemand der Anette₂[*c:dat*] das Buch[*c:acc*] [Op[c:dat] ohne zu vertrauen
if someone the Anette the book without to trust
] ausleiht
lends
“if someone lends Anette the book without trusting her”

4.3.2 Only Downward Agree

Case assignment is a process different from Agree (Chomsky 2001) and downward is the only possible direction for Agree, including case agreement as necessary for matching.

The variation in matching effects is derived if all cases of upward Agree in the present account are simply missing and if unchecked Agree features do not lead to a crash of the derivation (Bošković 2009; Preminger 2010).

¹Note that the picture is more complicated than shown here. See Assmann and Heck (2013) for discussion.

But if unchecked features can be deleted without successful Agree, we lose an explanation for:

- the ungrammaticality of mismatches
- locality

5 Conclusion

The pattern:

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.

Analysis:

The present account essentially builds on the order of post-syntactic agreement operations and the bidirectionality of Agree. Both in FRs and PGs, there is an overt item α and a covert item \emptyset which have to Agree in case features additionally to their normal case agreement relation with case assigning heads. 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.

Conclusion:

- An analysis based on the existence of a two-step bidirectional Agree operation can capture a kind of variation that other types of approaches struggle to derive.
- Furthermore, the analysis is an argument for a derivational view on post-syntactic operations since certain output representations are opaque. The opacity is resolved by making use of ordering of operations.

6 Complicating the pattern

6.1 More patterns

Not every speaker of Polish or German allows non-syncretic case mismatches in parasitic gap constructions (Bondaruk (1996)) or free relatives respectively (Riemsdijk (2006)).

Such varieties can be derived under the assumption that the agreement relation between α and \emptyset is symmetric: both α and \emptyset are probes. Intuitively, the strict varieties have both the Polish and the German property.

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

(43) *Speakers that allow mismatches*

i. $\alpha[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[*c:\text{dat}*]$	i. $\alpha[*c:_*] \rightarrow \emptyset[c:_]: \alpha[c:_]$
ii. $\alpha[*c:\text{dat}*] \rightarrow \emptyset[c:_]: \alpha[c:\text{dat}]$	ii. $\alpha[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}]$
iii. $\emptyset[c:_] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$	iii. $\emptyset[c:_] \rightarrow v[c:\text{acc}]: \emptyset[c:\text{acc}]$

(44) *Speakers that don't allow mismatches*

i. $\alpha[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[*c:\text{dat}*]$	i. $\alpha[*c:_*] \rightarrow \emptyset[*c:_*]: \alpha[c:_]$
ii. $\alpha[*c:\text{dat}*] \rightarrow \emptyset[*c:_*]: \alpha[c:\text{dat}]$	ii. $\alpha[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \alpha[c:\text{dat}]$
iii. $\emptyset[*c:_*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{acc}*]$	iii. $\emptyset[*c:_*] \rightarrow \alpha[c:\text{dat}]: \emptyset[*c:\text{dat}*]$
iv. $\emptyset[*c:\text{acc}*] \rightarrow \alpha[c:\text{dat}]: \not\downarrow$	iv. $\emptyset[*c:\text{dat}*] \rightarrow v[c:\text{acc}]: \not\downarrow$

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

(46) *Speakers that allow mismatch*

- | | | | |
|------|---|------|--|
| i. | $\emptyset[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[*c:\text{dat}*]$ | i. | $\emptyset[*c:_*] \rightarrow \alpha[c:_]: \emptyset[c:_]$ |
| ii. | $\emptyset[*c:\text{dat}*] \rightarrow \alpha[c:_]: \emptyset[c:\text{dat}]$ | ii. | $\emptyset[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[c:\text{dat}]$ |
| iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ | iii. | $\alpha[c:_] \rightarrow v[c:\text{acc}]: \alpha[c:\text{acc}]$ |

(47) *Speakers that don't allow mismatch*

- | | | | |
|------|---|------|---|
| i. | $\emptyset[*c:_*] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[*c:\text{dat}*]$ | i. | $\emptyset[*c:_*] \rightarrow \alpha[*c:_*]: \emptyset[c:_]$ |
| ii. | $\emptyset[*c:\text{dat}*] \rightarrow \alpha[*c:_*]: \emptyset[c:\text{dat}]$ | ii. | $\emptyset[c:_] \rightarrow \text{Appl}[c:\text{dat}]: \emptyset[c:\text{dat}]$ |
| iii. | $\alpha[*c:_*] \rightarrow v[c:\text{acc}]: \alpha[*c:\text{acc}*]$ | iii. | $\alpha[*c:_*] \rightarrow \emptyset[c:\text{dat}]: \alpha[*c:\text{dat}*]$ |
| iv. | $\alpha[*c:\text{acc}*] \rightarrow \emptyset[c:\text{dat}]: \zeta$ | iv. | $\alpha[*c:\text{dat}*] \rightarrow v[c:\text{acc}]: \zeta$ |

Assuming that the agreement relation between α and \emptyset is an essential property of FR and PG constructions, no language can be derived in which both constructions allow case mismatches. In fact, such a distribution of case matching effects has not been reported in the literature.

6.2 Syncretisms

In all the data above, syncretic forms could 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.

Assumptions:

Syncretic forms result from special morphological rules. For the sake of concreteness, I assume that syncretisms are due to language-specific *feature changing syncretism rules* (cf. Noyer (1992, 129)). Alternatively, impoverishment rules can be used.

- (48) a. To jest dziewczyna, której_{gen/dat} Jan się bał_{gen} zanim zaczął pomagać_{dat}.
 this is girl which Jan REFL fear before started 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.'

(49) *Syncretism rules in Polish*

- a. [acc] \rightarrow [gen]/[anim]
 b. [dat] \rightarrow [gen]/[rel], [fem]

Furthermore the condition under which agreement fails must be refined: agreement can always apply and potentially add additional values to features. Only if a mismatch on the probe cannot be circumvented by a syncretism rule, the derivation fails. That is, the syncretism rules apply early before the final case assignment operation applies (cf. Trommer (2002); Keine (2010)).

(50) *Derivation of (48-b)*

- | | | | |
|------|---|------|--|
| i. | $\alpha[c:_] \rightarrow \text{Appl}[c:\text{gen}]: \alpha[c:\text{gen}]$ | i. | $\alpha[c:_] \rightarrow \text{Appl}[c:\text{gen}]: \alpha[c:\text{gen}]$ |
| ii. | $\emptyset[*c:_*] \rightarrow \alpha[c:\text{gen}]: \emptyset[*c:\text{gen}*]$ | ii. | $\emptyset[*c:_*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{acc}*]$
$[acc] \rightarrow [gen]/[anim]: \emptyset[*c:\text{gen}*]$ |
| iii. | $\emptyset[*c:\text{gen}*] \rightarrow v[c:\text{acc}]: \emptyset[*c:\text{gen},\text{acc}*]$
$[acc] \rightarrow [gen]/[anim]: \emptyset[*c:\text{gen},\text{gen}*] = \emptyset[*c:\text{gen}*]$ | iii. | $\emptyset[*c:\text{gen}*] \rightarrow \alpha[c:\text{gen}]: \emptyset[c:\text{gen}]$ |

Note:

By assumption, \emptyset and α agree in other features such as animacy as well. Therefore, the context for feature changing is also given on \emptyset .

6.3 Case Hierarchy Effects in German

Not all case mismatches in German free relatives are tolerable. As has been observed in the literature, the pattern seems to involve the case hierarchy in (51) (Pittner (1991, 1995); Vogel (2001); Grosu (2003)). The condition is given in (52).

(51) *Case Hierarchy*

Nom > Acc > Dat/Gen

(52) *Matching Condition*

If the case assigned by the matrix clause is higher on the case hierarchy than the case assigned within the FR, the wh-phrase may bear the case of the FR, violating matching.

(53) a. Hans mag_{acc}, *wen_{acc}/wem_{dat} Maria vertraut_{dat}.

Hans likes who Maria trusts

b. Hans vertraut_{dat}, *wen_{acc}/*wem_{dat} Maria mag_{acc}.

Hans likes who Maria trusts

The case hierarchy effects do not follow from the present analysis without further ado. In order to solve the problem, the following assumptions can be added:

- In Assmann (2013) the problem was solved by decomposing case features so that the hierarchy is implemented as a subset relation, e.g. as in (54) (cf. Caha (2009), Béjar and Řezáč (2009)).

(54) *Case Hierarchy*

{nom} ⊂ {nom,acc} ⊂ {nom, acc, dat}/{nom, acc, gen}

- Furthermore, the symmetry of case assignment relations must be subject to parametrization: In German the relation $\alpha - \emptyset$ is symmetric and the relation DP – case assigner is asymmetric, while in Polish it is reversed.

Note: This gives rise to more patterns: languages with strict case matching in both constructions always have symmetric relations, while languages for which lack symmetric relations entirely should allow case mismatches without case hierarchy effects.

- Case assignment between a case assigning head and a DP fails if both probe and goal bear case features and if a case feature on the goal does not find a match on the probe (goal ⊆ probe).
- Case assignment between α and \emptyset , i.e. between two DPs, fails if both bear cases and the cases are not identical (goal = probe).
- The order of operations must be more restricted. A double probe first receives the case features of the closest goal.

6.3.1 German

(55) *FR: \emptyset [acc], α [dat]*

ii. $\alpha[*c:_*] \rightarrow \emptyset[*c:_*]: \alpha[c:_]$

ii. $\alpha[c:_] \rightarrow \text{Appl}[c:\{\text{dat,acc,nom}\}]: \alpha[c:\{\text{dat,acc,nom}\}]$

iii. $\emptyset[*c:_*] \rightarrow \alpha[c:\{\text{dat,acc,nom}\}]: \emptyset[*c:\{\text{dat,acc,nom}\}^*]$

iv. $\emptyset[*c:\{\text{dat,acc,nom}\}^*] \rightarrow v[c:\{\text{acc,nom}\}]: \emptyset[c:\{\text{dat,acc,nom}\}] (v \subseteq \emptyset)$

Importantly, the closest goal of the probe on \emptyset is α . Thus, agreement between \emptyset and α must precede agreement between \emptyset and matrix v. The configuration is successful.

- (56) *FR: $\emptyset[dat], \alpha[acc]$*
- i. $\alpha[*c:_*] \rightarrow \emptyset[*c:_*]: \alpha[c:_]$
 - ii. $\alpha[c:_] \rightarrow v[c:\{acc,nom\}]: \alpha[c:\{acc,nom\}]$
-
- iii. $\emptyset[*c:_*] \rightarrow \alpha[c:\{acc,nom\}]: \emptyset[*c:\{acc,nom\}^*]$
 - iv. $\emptyset[*c:\{acc,nom\}^*] \rightarrow Appl[c:\{dat,acc,nom\}]: \downarrow (Appl \not\subseteq \emptyset)$
- (57) *PG: $\emptyset[dat], \alpha[acc]$*
- i. $\emptyset[*c:_*] \rightarrow Appl[c:\{dat,acc,nom\}]: \emptyset[*c:\{dat,acc,nom\}^*]$
 - ii. $\emptyset[*c:\{dat,acc,nom\}^*] \rightarrow \alpha[*c:_*]: \emptyset[c:\{dat,acc,nom\}]$
-
- iii. $\alpha[*c:_*] \rightarrow v[c:\{acc,nom\}]: \alpha[*c:\{acc,nom\}^*]$
 - iv. $\alpha[*c:\{acc,nom\}^*] \rightarrow \emptyset[c:\{dat,acc,nom\}]: \downarrow (\alpha \neq \emptyset)$
- (58) *PG: $\emptyset[acc], \alpha[dat]$*
- i. $\emptyset[*c:_*] \rightarrow v[c:\{acc,nom\}]: \emptyset[*c:\{acc,nom\}^*]$
 - ii. $\emptyset[*c:\{acc,nom\}^*] \rightarrow \alpha[*c:_*]: \emptyset[c:\{acc,nom\}]$
-
- iii. $\alpha[*c:_*] \rightarrow Appl[c:\{dat,acc,nom\}]: \alpha[*c:\{dat,acc,nom\}^*]$
 - iv. $\alpha[*c:\{dat,acc,nom\}^*] \rightarrow \emptyset[c:\{acc,nom\}]: \downarrow (\alpha \neq \emptyset)$

By assumption, the dative assigning head *Appl* is merged outside VP (cf. McFadden (2004)). Then *Appl* is (just like *v*) closer to α than \emptyset .

6.3.2 Polish

- (59) *FRs: $\alpha[dat], \emptyset[acc]$*
- i. $Appl[*c:\{dat,acc,nom\}^*] \rightarrow \alpha[*c:_*]: Appl[c:\{dat,acc,nom\}]$
 - ii. $\alpha[c:_] \rightarrow Appl[c:\{dat,acc,nom\}]: \alpha[c:\{dat,acc,nom\}]$
-
- iii. $\emptyset[*c:_*] \rightarrow \alpha[c:\{dat,acc,nom\}]: \emptyset[*c:\{dat,acc,nom\}^*]$
 - iv. $\emptyset[*c:\{dat,acc,nom\}^*] \rightarrow v[c:\{acc,nom\}]: \emptyset[c:\{dat,acc,nom\}]$
 - v. $v[*c:\{acc,nom\}^*] \rightarrow \emptyset[c:dat,acc,nom]: \downarrow (\emptyset \not\subseteq v)$
- (60) *FRs: $\alpha[acc], \emptyset[dat]$*
- i. $v[*c:\{acc,nom\}^*] \rightarrow \alpha[*c:_*]: v[c:\{acc,nom\}]$
 - ii. $\alpha[c:_] \rightarrow v[c:\{acc,nom\}]: \alpha[c:\{acc,nom\}]$
-
- iii. $\emptyset[*c:_*] \rightarrow \alpha[c:\{acc,nom\}]: \emptyset[*c:\{acc,nom\}^*]$
 - iv. $\emptyset[*c:\{acc,nom\}^*] \rightarrow Appl[c:\{dat,acc,nom\}]: \downarrow (Appl \not\subseteq \emptyset)$
- (61) *PGs: $\emptyset[dat], \alpha[acc]$*
- i. $Appl[*c:\{dat,acc,nom\}^*] \rightarrow \emptyset[*c:_*]: Appl[c:\{dat,acc,nom\}]$
 - ii. $\emptyset[*c:_*] \rightarrow Appl[c:\{dat,acc,nom\}]: \emptyset[*c:\{dat,acc,nom\}^*]$
 - iii. $\emptyset[*c:\{dat,acc,nom\}^*] \rightarrow \alpha[*c:_*]: \emptyset[c:\{dat,acc,nom\}]$
 - iv. $v[*c:\{acc,nom\}^*] \rightarrow \alpha[c:_]: v[c:\{acc,nom\}]$
 - v. $\alpha[c:_] \rightarrow v[c:\{acc,nom\}]: \alpha[c:\{acc,nom\}]$
- (62) *PGs: $\emptyset[acc], \alpha[dat]$*
- i. $v[*c:\{acc,nom\}^*] \rightarrow \emptyset[*c:_*]: v[c:\{acc,nom\}]$
 - ii. $\emptyset[*c:_*] \rightarrow v[c:\{acc,nom\}]: \emptyset[*c:\{acc,nom\}^*]$
 - iii. $\emptyset[*c:\{acc,nom\}^*] \rightarrow \alpha[*c:_*]: \emptyset[c:\{acc,nom\}]$
 - iv. $Appl[*c:\{dat,acc,nom\}^*] \rightarrow \alpha[c:_]: Appl[c:\{dat,acc,nom\}]$
 - v. $\alpha[c:_] \rightarrow Appl[c:\{dat,acc,nom\}]: \alpha[c:\{dat,acc,nom\}]$

References

- Arregi, Karlos and Andrew Nevins (2012), *Morphotactics: Basque Auxiliaries and the Structure of Spellout*, Springer, Dordrecht.
- 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.
- Assmann, Anke and Fabian Heck (2013), Opaque Intervention, in F.Heck and A.Assmann, eds, 'Rule Interaction in Grammar (Linguistische Arbeitsberichte 90)', Universität Leipzig, Leipzig, pp. 463–509.
- 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 Āezáč (2009), 'Cyclic agree', *Linguistic Inquiry* **40**(1), 35–73.
- Bennis, Hans and Teun Hoekstra (1985), 'Gaps and Parasitic Gaps', *The Linguistic Review* **4**, 29–87.
- 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 (2009), 'Unifying first and last conjunct agreement', *Natural Language and Linguistic Theory* **27**, 455–496.
- Bresnan, Joan and Jane Grimshaw (1978), 'The Syntax of Free Relatives in English', *Linguistic Inquiry* **9**(3), 331–391.
- Caha, Pavel (2009), The Nanosyntax of Case, PhD thesis, University of Tromsø, Tromsø.
- 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.
- 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 (2001), Derivation by Phase, in M.Kenstowicz, ed., 'Ken Hale: A Life in Language', MIT Press, Cambridge, Massachusetts, pp. 1–52.
- Cinque, Guglielmo (1990), *Types of A'-dependencies*, MIT Press, Cambridge, Massachusetts.
- 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.
- Donati, Caterina and Carlo Cecchetto (2011), 'Relabeling Heads: A Unified Account for Relativization Structures', *Linguistic Inquiry* **42**(4), 519–560.
- Engdahl, Elisabet (1983), 'Parasitic gaps', *Linguistics and Philosophy* **6**, 5–34.
- Fanselow, Gisbert (1993), 'Die Rückkehr der Basisgenerierer', *Groninger Arbeiten zur germanistischen Linguistik* **36**, 1–74.
- 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.
- Grewendorf, Günther and Erich Groat (2013), Labeling and Agree in free relatives. Ms., Goethe University Frankfurt.
- 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 of Markedness in Generative Grammar: Proceedings of the 1979 GLOW Conference', Scuola Normale Superiore, Pisa, pp. 171–216.
- 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.
- Harbert, Wayne (1983), 'On the Nature of the Matching Parameter', *The Linguistic Review* **2**, 237–284.
- Hirschbühler, Paul and María-Luisa Rivero (1981), 'Catalan Restrictive Relatives: Core and Periphery', *Language* **57**, 591–625.
- 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.
- 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.
- Keine, Stefan (2010), *Case and Agreement from Fringe to Core: A Minimalist Approach*, Vol. 536 of *Linguistische Arbeiten*, De Gruyter, Berlin.

- Kuroda, Sige-Yuki (1968), 'English Relativization and Certain Related Problems', *Language* **44**, 244–266.
- McFadden, Thomas (2004), The position of morphological case in the derivation: a study on the syntax-morphology interface, PhD thesis, University of Pennsylvania, Philadelphia.
- 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 (2004), *Linearization of Chains and Sideward Movement.*, MIT Press, Cambridge, Massachusetts.
- Ott, Dennis (2011), 'A Note on Free Relative Clauses in the Theory of Phases', *Linguistic Inquiry* **42**(1), 183–192.
- Pesetsky, David and Esther Torrego (2007), The Syntax of Valuation and the Interpretability of Features, in S.Karimi, V.Samiian 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.
- Preminger, Omer (2010), Failure to Agree is Not a Failure: ϕ -Agreement with Post-Verbal Subjects in Hebrew, in J. v.Craenenbroeck and J.Rooryck, eds, 'Linguistic Variation Yearbook 9', John Benjamins, Amsterdam, pp. 241–278.
- 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.
- 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.
- 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.
- Truckenbrodt, Hubert (1995), Extraposition from NP and Prosodic Structure, in J.Beckman, ed., 'Proceedings of the North Eastern Linguistics Society 25', GLSA Publications, Amherst, Massachusetts, pp. 503–517.
- 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.
- Williams, Edwin (1990), 'The ATB theory of parasitic gaps', *The Linguistic Review* **6**, 265–279.
- Zeijlstra, Hedde (2012), 'There is only one way to agree', *The Linguistic Review* **29**, 491–539.