

DEFAULTS AND THE THEORY OF GRAMMAR

By

Kali Elizabeth Morris

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Linguistics – Doctor of Philosophy

2018

ABSTRACT

DEFAULTS AND THE THEORY OF GRAMMAR

By

Kali Elizabeth Morris

This thesis is concerned with the nature of syntactic defaults and what their investigation can tell us about the theory of grammar. Since Chomsky (1995) first introduced feature-checking, we've understood the need to value features to be central to how the grammar regulates grammaticality. The failure of an uninterpretable feature to receive a value and be deleted before reaching the interface induces the derivation crash that differentiates grammatical sentences from ungrammatical ones. Defaults offer us an interesting domain of inquiry because we would expect them to be impossible to generate in this type of system; nonetheless, they surface in a number of core syntactic domains. The existence of syntactic defaults raises three central questions: first, how is it that defaults are produced in a system where the failure to value features causes ungrammaticality? Second, how is it that the production of defaults is constrained such that whatever mechanism accounts for their production doesn't overapply to instances where they aren't licit? Finally, given that syntactic defaults appear to involve underspecification, what can an understanding of the default mechanism tell us about the role of underspecification in the syntactic domain?

In this thesis, I focus our attention on two arenas: defaults in the domains of case and φ -agreement. A number of proposals have been made in recent years that address these issues, at least in part. They share a similar logic: the way to account for how defaults surface in the system is to abandon the notion that failing to value a feature is fatal to the derivation. I will argue in this thesis that by making small modifications to the generally accepted framework, we can account for the production of defaults without having to abandon that notion.

One such departure is dependent case theory – a configurational approach to case whereby case features are valued not through their relationships with case-assigning functional heads, but rather by their relative positions to other nominals. Built into this system is a default case, assigned as a

last resort to nominals that have failed to receive a more specific value. While a desire to understand defaults is not what originally guided the proposal of dependent case theory, its ability to easily account for their production has certainly contributed to its widespread adoption. In the domain of φ -agreement, another departure called obligatory operations addresses the default issue more directly and proposes a new understanding of what drives derivations. It is not the need to value features that explains why φ -agreement is obligatory, but rather that the operations responsible for establishing those dependencies are obligatory themselves. By shifting the explanatory burden to the triggering of operations, rather than their outcomes, obligatory operations claims that syntactic operations can fail, without inducing ungrammaticality; thus providing a solution to the default production problem. While the departures in both arenas have directly addressed the issue of how defaults are produced, neither has been too successful in understanding how that production is constrained. Furthermore, in order to solve the production issue each has to abandon the central tenet of feature valuation.

I argue that in light of a host of deep conceptual and empirical issues regarding these two departures, we are better served to handle the default problem by making modest modifications to the standard syntactic framework that the field has adopted since (Chomsky, 2000, 2001). I extend a decomposition of AGREE that is sensitive to inherent hierarchical relationships between features to both produce and – more importantly – constrain the distribution of syntactic defaults (Béjar, 2003). This decomposition produces three outcomes of agreement – rather than the standard two – and it is in this third outcome where we find syntactic defaults and other interesting types of underspecification and repairs. What is available to us through this proposal is an understanding of how defaults are both produced and how that production is constrained and the simultaneous ability to maintain standard assumptions about the role of feature valuation in regulating grammaticality. Through this system, we can also gain further insight into the nature of underspecification and its role in the syntactic component.

Copyright by
KALI ELIZABETH MORRIS
2018

To all those who have believed in me more than I've believed in myself.
I hope to have made you proud.

ACKNOWLEDGEMENTS

this is why we're called advisors, not founts of information

— Alan Munn

I have dreaded writing this acknowledgments page for years because it is sure to fall short in expressing the immense gratitude I have for everyone who has helped me get to where I am today.

Alan has simply been the best advisor I could have dreamed of; he is the true embodiment of the word. He is an admirable role model and I owe him a lifetime of gratitude for how much I've grown under his guidance, both intellectually and personally. He introduced me to syntax, taught me to love theory, how to teach, how to create an argument (and how not to). He was immensely encouraging in the honest kind of way that you trusted he wasn't just pulling you along. He was unwavering in his support and I've never once doubted I was a priority or that he cared. He has been insanely generous with his time and always knows exactly what to say to calm me down, or get me moving faster on something. No matter how stressful things were, meeting with Alan instantly removed all worry. He was incredibly patient and kind while simultaneously pushing me to do better. My life has been enriched in numerous ways by knowing him and while I am excited for whatever chapter next awaits me, I am so terribly sad to give up our weekly meetings. I hope that my time learning from him is far from over. Thank you so very, very much.

I am so grateful for the amazing guidance Suzanne has given me, especially during my first few years in the department. We wrangled twitter data together, taught in the trenches of IAH together, and she was such a huge support in helping me transition to living so far from home. She always encouraged me to find a good balance between the different parts of my life and I always appreciated that. Working for and with her gave me an appreciation for sociolinguistics and rigorous quantitative work and was such a pleasure. Of course I showed that appreciation by writing an entirely theoretical syntax dissertation, but I hope she'll forgive me - haha. Thank you for everything.

I don't think Cristina realizes how many times her reassurance has pulled me out of a self-deprecating spiral. Her insights are always so perfect and exactly what you need to hear to draw the connection you were missing. I learned so much from observing how she presented arguments and how she was able to cut immediately to the big untouchable, messy issue that no one wanted to talk about. Having her support through this project has been such a source of strength and I have such fond memories of syntax reading group in the lab. Thank you, thank you.

Watching Marcin get excited about a good idea is probably one of my favorite things about the work half of my life. His passion for good argumentation, clever ideas no matter how crazy, and his ability to see straight through all the bullshit is something that I truly enjoy watching and deeply admire. He also always asks the tough questions right out the gate, which is so helpful when you're trying to be better. He has been so fun to talk to throughout this process and his questions always get me to think about the broader – not just this tiny part of syntax – picture. Thank you for all you've done for me.

Thank you to Dr. Brown for teaching me to write, Dr. Kendall for introducing me to linguistics for the very first time, and Dr. Kallendorf for pushing me to not defer so much.

To my grad school bestie, Jessica. I don't know what I would have done without you. I am so glad that out of all the things I picked up while in grad school, a lifelong friend is one of them. Here's to many more years of friendship – and hopefully we can convince Steve to move y'all down to Texas one day ;)

My syntax big brothers: Greg, Joe, Daniel, Matt H. and Matt K. Y'all welcomed me so kindly when I was new and I feel so lucky to have had so many people looking out for me and guiding me. Plus talking about syntax is so much more fun over beers. I miss y'all so much.

To Greg – I can't tell you how much I appreciate our friendship and how it has grown over the years. You took me under your wing from the very first day and many years later, you're one of my closest friends. Thank you isn't nearly enough. Now that I'm done, let's take over the south!

Hannah, Curt, Ai, and Karthik – Hannah, I don't think I've ever met such a genuinely kind person. You made my life in Michigan so much better and whether you knew it or not, you helped

encourage me to be a more patient, understanding person. Thank you. Curt, thanks for putting up with my monologues for such a long time and for always being down to go work or just hangout at the bar/coffee shop when I needed company. Grad school was way more enjoyable with you there, thank you. Ai, you are that magical mix of approachable while yet so good at what you do it's intimidating. I learned so much from you about teaching and I always enjoyed every one of our long chats. Karthik, I have learned so much about so many different things from you I can't even start to list them. Those lessons have greatly adjusted my perspective on the world around me and I'm a better person for it. Thank you. You are always so much fun to be around and to talk with. I miss our weekly trivia nights and our discussions about How I Met your Mother.

To the lovely staff at the Starbucks Reserve on Rayford and Kuykendahl.

My mom and dad are the loveliest and most supportive people on the planet. There are simply no words to express how grateful I am for everything they've done for me. Thank you for putting such a high value on education, for sending me to the best schools, and for all the sacrifices you guys have made along the way to provide me with the best opportunities – and of course a fantastic place to write the final chapters of this dissertation. I love y'all so, so much.

Nana and Granddad – my second set of parents. Thank you for all the unconditional support through the years. I hope I've made y'all proud. Thank you for the lifetime of love, long boozy lunches, and chats on the porch. I love you both.

I am surrounded by a family full of incredibly supportive people who are always looking out for me: Pete, Devin, David, Kevin, Kathy, Katie, Mike, Tim, and Lisa. Thank you for being interested in my work, for being invested in my future, and for being so much fun to be around.

To my inner circle – Bonnie, Brook, Candice, Danny, Jerod, Katelyn, McKinney, Mike, and Sarah – y'all have kept me sane and grounded by always giving me a place to keep a foot in the “real world”. And I am forever grateful for the years of amazing friendship (decades in some cases – when we did we get old?). Thank you for listening to 800 elevator pitches even though no one still really has a clue what I do – your unconditional support otherwise has made a world of difference. You are the best cheerleaders a girl could hope for!

I have been incredibly lucky to marry into a wonderful family full of the most kind and supportive people. Thank you Mark, Cindy, Lisa, Dustin, Witten, Madison(yay!), Gramps, Grandmom, and Angie for being so wonderful to me over the years. I appreciate everything you've done for me.

To Joe, Jane, and Brutus – thank you for bringing me such joy.

And finally, there will never be enough words to convey my appreciation for my husband Andrew. He has been a pillar of strength and support throughout the whole process: from 12 hour Skype sessions when I was homesick, to the near constant reassurance I needed towards the end, the hugs when I was having an anxious day, and all the million other small things that made such a huge difference. There is no world in which I could have imagined getting through this without you by my side. I'm just happy to have the rest of my life to express my gratitude. I promise I'll stop going to school now (maybe)– hahaha.

This dissertation is dedicated to all those who've believed in me more than I believe in myself. Thank you and I hope I've made you proud.

TABLE OF CONTENTS

LIST OF TABLES	xiii
KEY TO ABBREVIATIONS	xiv
CHAPTER 1 INTRODUCTION	1
1.1 Introducing Defaults	1
1.1.1 Some Data and What it Means	2
1.1.2 The Crux of the Default Problem	7
1.2 Domain-Specific Defaults	8
1.2.1 φ -agreement and Case	9
1.2.2 Defaults in the φ -agreement Domain	13
1.2.3 Defaults in the Case Domain	15
1.3 Jumping Ship lands us in Bizarre Boats	16
1.3.1 Dependent Case Theory	17
1.3.2 Separation of Case from Licensing	18
1.3.3 Obligatory Operations	20
1.4 Goals of this Thesis	20
CHAPTER 2 DEPENDENT CASE THEORY	23
2.1 The Default Case Issue	23
2.2 Dependent Case Theory	25
2.2.1 Overview of model	26
2.2.1.1 Early Versions	26
2.2.1.2 Modern Versions	32
2.2.1.3 Interim Walkthrough	46
2.2.2 Implications	50
2.2.2.1 Abandonment of Government	51
2.2.2.2 Parameterization	62
2.2.2.3 Dependency Establishment	65
2.3 Separation of Case from Licensing	66
2.3.1 Motivations	67
2.3.2 Implications	68
2.4 Conclusions	75
CHAPTER 3 OBLIGATORY OPERATIONS	76
3.1 Introduction	76
3.2 Obligatory Operations	76
3.3 An alternative	90
3.3.1 An overview of MATCH/VALUE	91
3.3.2 Accounting for Kichean	98
3.4 Failed Agreement isn't Always Default Agreement	106

3.4.1	Person Hierarchy Effects	106
3.4.1.1	Morphological Effects	108
3.4.1.2	Syntactic Effects	122
3.4.1.3	Probe Modification	128
3.4.2	Dative Intervention	133
3.4.3	Conjunct Agreement	138
3.4.4	Interim Summary	144
3.5	The Premature Overapplication of Defaults	145
3.6	Some Conceptual Issues	150
3.6.1	Framework-wide adoption	150
3.6.2	What are probes?	154
3.7	Conclusions	156
CHAPTER 4 AGREE-BASED CASE		158
4.1	Introduction	158
4.1.1	Revisiting the Problem of Default Case	158
4.1.2	Previous Agree-based Approaches	160
4.2	A New Approach	164
4.2.1	Case Feature Systems	164
4.2.1.1	Preliminary Concerns	165
4.2.1.2	The Hierarchical Nature of Case Features	171
4.2.2	A Proposed Feature System	177
4.2.2.1	Morphological Functions	177
4.2.2.2	Syntactic Function	183
4.2.2.3	A Summary	188
4.2.3	Accounting for Default Case	190
4.2.3.1	Canonical Case Valuation	193
4.2.3.2	Quirky Case	198
4.2.3.3	Hanging Topic/Left-Dislocation	200
4.2.3.4	Coordination	203
4.2.3.5	Gapping	205
4.2.3.6	ACC-ing gerunds	207
4.2.3.7	Modified Pronouns	210
4.3	Evaluating Our Options	214
4.3.1	Some Problems for Dependent Case Models	215
4.3.1.1	Sole Accusative Arguments	215
4.3.1.2	Dependent Case Theory and Default Case	217
4.3.2	Final Remarks	222
CHAPTER 5 CONCLUSIONS		225
5.1	Introduction	225
5.2	The Lifeboats are Headed in the Wrong Direction	226
5.3	Putting out the Fire Allows us to Maintain the Course	228
5.4	What Have We Learned	228

REFERENCES	231
----------------------	-----

LIST OF TABLES

Table 3.1: Kichean Agreement Markers	80
Table 3.2: MATCH outcomes	93
Table 3.3: VALUE outcomes	93
Table 3.4: MATCH and VALUE interactions	94
Table 3.5: Georgian Agreement Morphemes	109
Table 3.6: Karok Agreement Morphemes	115
Table 3.7: Mordvinian Agreement Morphemes	117
Table 3.8: Nishnaabemwin Agreement Morphemes	132
Table 4.1: Accidental Homophony in Russian	172
Table 4.2: Greek Adjective ‘wise’	173
Table 4.3: Finnish Syncretism Core/Non-core	175
Table 4.4: Erzja Mordvin Syncretism Non-core Cases	176
Table 4.5: Cross-Classification of Features	178
Table 4.6: Decomposition of Cases:	193

KEY TO ABBREVIATIONS

1	first person	GEN	genitive
2	second person	IMP	imperative
3	third person	INF	infinitive
A	agent argument	INTR	intransitive
ABL	ablative	LOC	locative
ABS	absolutive	M	masculine
ACC	accusative	N	neuter
ADJ	adjective	NEG	negative
ADV	adverb	NOM	nominative
AGR	agreement	OBJ	object
ALL	allative	OBL	oblique
APPL	applicative	PFV	perfective
AUX	auxiliary	POSS	possessive
COM	comitative	PRF	perfect
COMP	complementizer	PRS	present
DAT	dative	PROG	progressive
DET	determiner	PST	past
ERG	ergative	SUBJ	subject
F	feminine	TOP	topic
FOC	focus		

CHAPTER 1

INTRODUCTION

1.1 Introducing Defaults

This thesis is concerned with the existence of syntactic defaults and the issues their existence raises for the set of theoretical assumptions we consider standard. Most important is the assumption that what drives derivations is the need to value any unvalued features; failure to do so causes derivation crashes when those unvalued features reach the interfaces. Defaults, at first blush, appear to constitute significant difficulties for maintaining a framework that centers on the failure-to-value assumption because it is exactly this failure that is assumed to be responsible for their production. The apparent incompatibility of default data with this framework has encouraged a number of proposals that involve quite radical departures from these traditional assumptions. In the domain of case, default data has in part triggered a move towards the separation of case from DP licensing and the adoption of an alternative model of case valuation called dependent case theory (Baker, 2015; Levin & Preminger, 2015; Marantz, 1991; McFadden, 2004). In the domain of agreement, default data has triggered the adoption of an alternative model of syntactic operations, Preminger's (2014) obligatory operations model. Each of these proposals has received much deserved praise; however I suggest that the optimism surrounding their adoption is overstated and that we should refocus our attention towards addressing the default issues without completely recasting these basic assumptions. At its core, this thesis is a call for a more conservative approach, arguing that what we lose by 'jumping ship' doesn't outweigh what we gain by maintaining the course *if* we can reconcile how defaults exist in a framework that categorically appears to rule them out.

1.1.1 Some Data and What it Means

We must first make a distinction between two related questions: (i) what are defaults in an empirical sense and (ii) how do we understand those defaults to be formally represented? To address the first, consider the following data from Hindi-Urdu (Bhatt, 2005). In Hindi-Urdu, the φ -features of finite T must agree with the closest argument that is not morphologically case-marked. This is the pattern we see in (1a) and (1b). In (1a) the closest non-case-marked argument is the subject; agreement is successful and the subject's φ -features surface on the verb. In (1b) agreement with the subject is blocked by the ergative case marking on the subject and so agreement proceeds with the next closest argument, the object. What is relevant to the default discussion is that when both arguments are morphologically case marked, and therefore both unavailable for φ -agreement, the derivation does not crash, but instead produces (1c), with masculine singular features appearing on the verb; Bhatt (2005) among others who work with similar data, classify this as *default agreement*.

- (1) a. **Mona** amruud **khaa-tii thii**
Mona.F guava.F eat.HAB.F be.PR.F.F.SG
'Mona used to eat guava' SUBJECT AGREEMENT
- b. Ram-ne **imlii khaa-yii thii**
Ram.M.ERG tamarind.F eat.PFV.F be.PST.F.SG
'Ram had eaten tamarind' OBJECT AGREEMENT
- c. Mona-ne is kitaab-ko parh-aa thaa
Mona.F.ERG this book.F.ACC read.PFV.M.SG be.PST.M.SG
'Mona had read this book' DEFAULT AGREEMENT

We observe a similar phenomenon in the domain of case assignment as well. In the English sentence in (2) we see an example of what Schütze (2001) among others calls *default case* where the DP surfaces with accusative case, despite the absence of an accusative case assigner.

- (2) What?! **Him** wear a tuxedo?! No way!

The data in (1c) and (2) both show the appearance of features, masculine singular and accusative respectively, despite no source for them in their respective derivations. Furthermore, the features that surface in each example are consistent within each language. In every example of default

agreement in Hindi-Urdu, it is the masculine singular features that surface; in every instance of English default case, accusative case features are the ones we observe.

While the set of features that surface in these examples is consistent within a language, it can vary cross-linguistically. Example (3) lists the environments where we observe default accusative case in English, while examples (4) and (5) show that these same environments produce the nominative forms in German and Spanish, respectively (Schütze, 2001).

(3) **Default Case in English:**

a. *Hanging Topic/Left-Dislocation*

What?! **Him** wear a tuxedo?!

b. *Gapping*

She will eat cake, **him** brownies.

c. *Coordination*

Me and **him** will go to the store.

d. *Modified Pronouns*

Lucky **me** has to clean all the toilets.

(4) **Default Case in German:**

Hanging Topic/Left-Dislocation

- a. **Der/*Dem** Hans, mit dem spreche ich nicht mehr.
the.NOM/*DAT with him.DAT speak I not anymore.

(Schütze, 2001)

(5) **Default Case in Spanish**

Coordination

- a. para **tú** y **yo**
for **you.NOM** and **I.NOM**
- b. *para ti y mí
for you.ACC and me.ACC
- c. para ti/*tú
for you.ACC/*you.NOM

- d. para mi/*yo
 for me.ACC/*you.NOM

(Schütze, 2001)

The intra-linguistic consistency we see in (3) coupled with the cross-linguistic variation shown in (4) and (5) tells us that the task at hand is not to account for unexpected accusative valuation. Rather, we need to explain how accusative forms appear in English while nominative forms appear in German and Spanish, despite appearing in the same positions and despite the lack of either an accusative or nominative source of features. This data therefore suggests that the grammar has a robust and powerful default mechanism that supplies default features in certain instances where the derivation fails to do so for one reason or another (Legate, 2008; McFadden, 2004; Schütze, 2001). One broad goal of this thesis is to better understand this powerful mechanism and how it is integrated into our theoretical framework across a number of different domains.

With some clarification of how we identify defaults empirically, we are in a position to address the second question – how we understand defaults are formally represented. Of course the way we understand the data in (1c) and (2) and, by extension, how we understand defaults formally, depends in large part on the set of theoretical assumptions we choose to hold. I'll begin with a broad overview, but will clarify the details further for specific domains in section 1.2. Adopting the general framework set up in Chomsky (1995, 2000, 2001), we assume that the syntactic system is a feature-based derivational one and that features are what drive the two primary syntactic operations: MERGE and AGREE. Each terminal syntactic object is a bundle of morphosyntactic features. These features come in two flavors: valued features and unvalued ones. Valued features are, not unsurprisingly, features for which a value is inherently specified. Unvalued features by contrast are features whose value is not inherently specified and therefore must come from somewhere else in the derivation, usually through establishing a relationship with a valued feature bearing object via AGREE. Unvalued features cannot survive once the derivation is sent to spell out and in this way, it is through the valuation of features that the requirements of various syntactic pieces are satisfied. A successful derivation is one in which all unvalued features have received a value before the derivation is sent in phases to the interfaces at spell out. Successful phases are sent to the morpho-phonological

component where vocabulary items are then inserted into each syntactic node (Halle & Marantz, 1993). This insertion is governed by a subset principle which essentially requires that the features on an eligible vocabulary item must constitute a subset of the features specified on the node into which they are to be inserted. Of the eligible vocabulary items, the one that is most specified will be inserted.

This system also makes use of underspecification, both within the syntactic nodes themselves and in the vocabulary items the morphological component makes available for insertion. We are probably most familiar with underspecified vocabulary items, called *elsewhere forms*, which are essentially just vocabulary items with a completely reduced set of morpho-phonological features. To illustrate, we can imagine that the following vocabulary items in (6) are available for the present tense of the verb *to be* in English to account for the paradigm in (7). The elsewhere form in this example – *are* – is inserted into every syntactic node that is unable to insert either one of the more specified vocabulary items *am* and *is* because its feature set does not constitute a superset of those respective vocabulary items.

(6)	[+singular, +author]	→	am	(7)	I	am	we	are
	[+singular, –participant]	→	is		you	are	y'all	are
	<i>elsewhere</i>	→	are		he	is	they	are

Elsewhere forms are distinct from the type of underspecification we observe within in the syntactic nodes themselves, despite both making use of underspecification. The syntactic type of underspecification is typically due to one of the following: either a syntactic node is generated underspecified, it winds up agreeing with something that was generated underspecified, or a node's specification was modified in some way to become underspecified under a defined set of circumstances. A common way for this modification to happen is through an operation like Impoverishment (Halle & Marantz, 1993), which I'll illustrate here. Take the feature specifications for the English pronoun paradigm in (8). The unvalued φ -features on verb nodes in English will be valued with the set of φ -features that corresponds with the pronoun that controls the agreement.

(8)	I	[+singular, +participant, +author]
	you	[+singular, +participant, –author]
	he/she/it	[+singular, –participant, –author]
	we	[–singular, +participant, +author]
	y’all	[–singular, +participant, –author]
	they	[–singular, –participant, –author]

After agreement, the verb node itself now has a featural specification identical to one of the items in (8) and the grammar will insert a corresponding vocabulary item, obeying the subset principle. For the past tense form of the verb *to be*, imagine English has the two vocabulary items in (9):

(9)	[+singular]	→	was
	<i>elsewhere</i>	→	were

The grammar would therefore insert *was* for every syntactic node that contained a [+singular] feature and insert *were* for every syntactic node that did not contain that feature. Note however that this system makes the wrong prediction for verbal nodes that agree with the second person singular pronoun *you*. Since the featural specification for the syntactic node *you* includes a [+singular] feature, the grammar should be directed to insert the most specified eligible vocabulary item, which in this case is [+singular] → *was*, not the *were* that we expect. However, imagine English has access to an additional rule that manipulates the featural specification of verbal nodes that agree with the second person singular pronoun, shown in (10).

(10)	[+singular] → [∅] / [+participant, +author]
------	---

What this rule says is that in the presence of both [+participant, +author] features, delete the [+singular] feature. Deleting this feature removes the [+singular] → *was* vocabulary item from the set of eligible competitors because it no longer contains a subset of the features on the modified verbal node. This deletion process is called Impoverishment and through this operation syntactic nodes *become* underspecified, despite originally having more featural information. Impoverishment is one way in which we can observe underspecification of the syntactic node itself, rather than the

more straightforward underspecification of the vocabulary items.

Because vocabulary insertion is governed by the subset principle, it's often the case that underspecified syntactic nodes are inserted with underspecified vocabulary items and in this way, they often co-occur. It's crucial though to understand going forward that the two are distinct: they occupy different components of the grammar and as we'll later see, this has consequences for how we're comfortable modeling them.

Defaults are quite intuitively similar to the underspecified elsewhere forms. Defaults and these forms both share a last-resort quality whereby they each only appear in the absence of something more specific. However, in order for default vocabulary items to ever “win” the insertion competition, the syntactic nodes into which they are inserted must also be underspecified. We can now summarize how we intend to define defaults formally through the theoretical lens we've just established. We can view default agreement as the failure of the verb to receive enough φ -features through agreement to be spelled out with a more specified vocabulary item. Likewise, we can understand default case as the failure of the DP to receive enough case feature information through the mechanism responsible for case valuation to dictate which morphological form to take.

1.1.2 The Crux of the Default Problem

The fact that underspecification can occupy distinct components of the grammar also means that the range and nature of the problems they can pose is quite different. Defaults on the morphological level are generally unproblematic. Given that one of the primary functions of this component is to deliver pronounceable strings of language, it's fairly intuitive to assume that there are default forms available that can be inserted when the instructions for pronunciation are unspecified. As an interface component it is also reasonable to assume that communication between the two components it connects is imperfect and defaults can serve to bridge that gap.

Defaults in the syntactic domain, however, are far more problematic given the theoretical framework we've established. The crux of the default problem is this: if defaults involve the kind of underspecification that results from the failure to value some set of features, how is it that the

grammar allows this given that one of its central tenets is the ungrammaticality that results from unvalued features surviving to the interfaces? Additionally, once we have an answer, how do we prevent whatever mechanism allows for the production of defaults to not apply in instances where we'd predict ungrammaticality. In other words, once we allow defaults to 'save' derivations, how do we prevent them from saving the wrong ones? Because defaults question the validity of whether or not the grammar can tolerate the failure to value features, their proper analysis has deep theoretical implications. We can understand these issues as guiding three research questions to keep in mind as we explore the role of defaults in various syntactic domains.

- (i) How does the grammar *produce* defaults?
- (ii) How does the grammar *constrain* the production of defaults?
- (iii) What can an understanding of syntactic defaults tell us about how the syntax can encode underspecification?

Syntactic defaults exist across a number of different syntactic domains. This thesis will focus on their existence in two: syntactic defaults in the domain of φ -agreement and syntactic defaults in the domain of case. The next sections will outline in a bit more detail some of the domain-specific assumptions we consider standard and how the default problem is explicitly expressed in each domain.

1.2 Domain-Specific Defaults

In this section, I'd like to provide a more thorough discussion of the default problem in the context of the two syntactic domains that will be the focus of this thesis. This involves again outlining what we consider standard assumptions, but this time with more of the domain-relevant details. We'll also address how the default issue specifically arises in both φ -agreement and case domains, when considering those assumptions. Because the goal is to identify which assumptions are considered fairly standard, this discussion is sure to gloss over many important details and

popular disagreements. What remains is a stripped down version of the relevant theories, but one that allows us to clearly see the problem at hand.

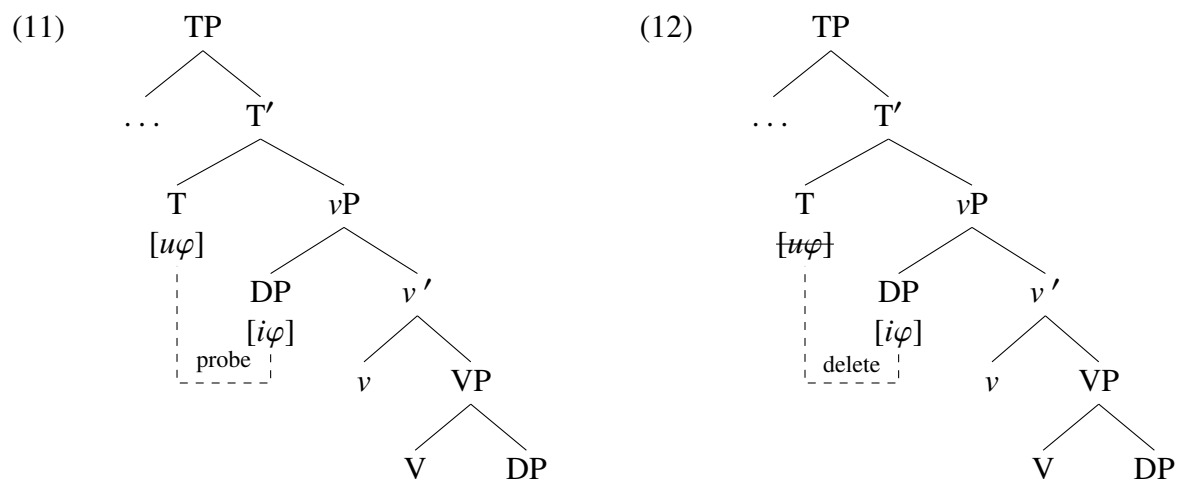
1.2.1 φ -agreement and Case

We begin with φ -agreement and the operation AGREE. Chomsky (2000, 2001) proposes that the way to distinguish between the two types of syntactic features is to ground the difference in whether a feature is interpretable by the semantic interface.¹ There are some features that the interface can interpret (interpretable features) and others that the interface cannot (uninterpretable features). Because the semantic interface is unable to ‘read’ those uninterpretable features, Chomsky argues that in order for a structure to be grammatical, these uninterpretable features must be removed before that structure reaches the semantic interface. AGREE is the operation responsible for ensuring this removal. The basic logic is that if a syntactic object that bears an uninterpretable feature can find an interpretable instance of the same feature category on another syntactic object, it can establish a relationship that will license the uninterpretable feature’s removal. The assumption is that it is only through establishing these kinds of relationships that uninterpretable features can be removed.

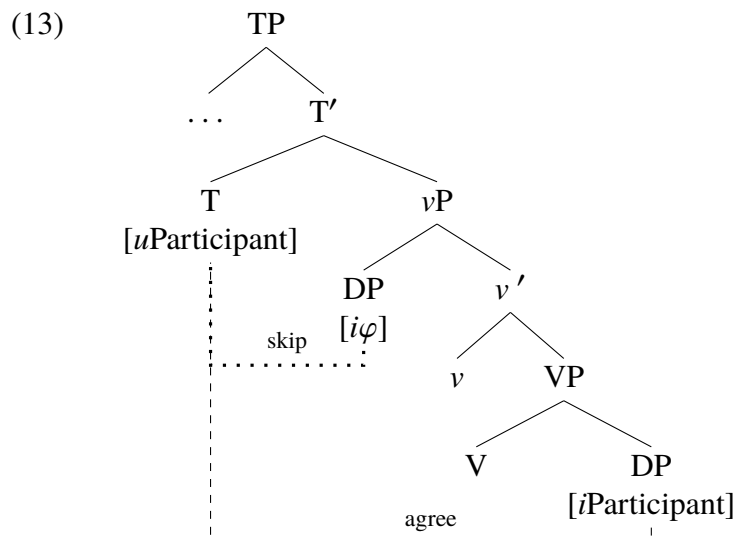
Chomsky assumes a fairly limited set of these uninterpretable features largely because by definition their existence is an imperfection of the system, violating the Interpretability Condition which assumes all features are properties of sound and meaning and are thus interpretable by the two interfaces. The φ -agreement and case domains are two domains in which these uninterpretable features have an especially significant role. In the φ -agreement domain, there exist interpretable φ -features on nominals and uninterpretable φ -features on some of the core functional categories, like C, T and *v*. To remove the uninterpretable instances of φ -features, the grammar must establish a feature removing relationship between the relevant functional heads and any nominals that bear the

¹Note that this is a slightly different distinction than the one made in the previous section. We’ll clarify the effect that these different distinctions have on the framework in chapter 4. Also note that uninterpretability at the semantic interface does not mean that interpretable features have semantic content. The classic case is gender features. Grammatical gender is not at all semantic, but can exist as interpretable features on nominals.

interpretable φ -feature counterparts. Chomsky defines the uninterpretable φ -features on functional heads as *probes* that search for nominal *goals* with which to agree. A probe can search for a matching interpretable feature category within its c-command domain (11) and if it successfully finds one, the feature removing relationship is established, the value of the interpretable feature is transferred to the uninterpretable feature-bearing syntactic object, and the uninterpretable feature is removed (12).

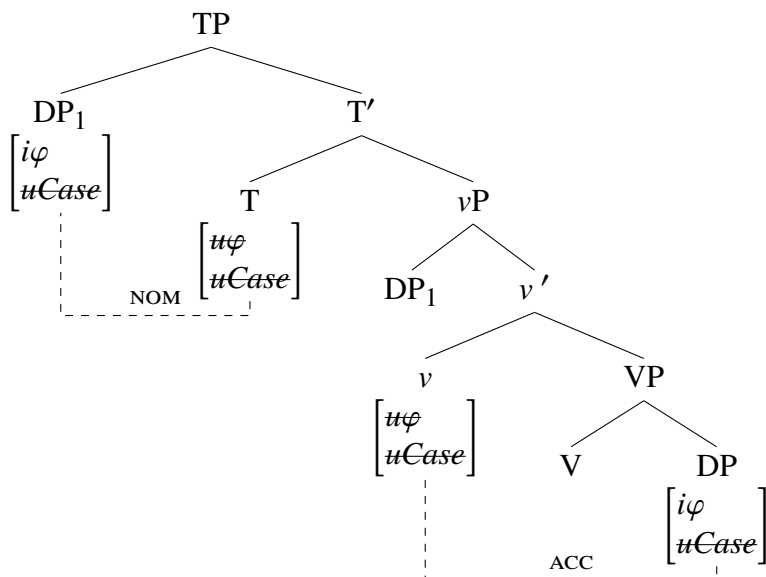


It's also fairly standard to assume some version of relativized minimality (Rizzi, 1990) which further refines the degree of specificity to which probes are sensitive, essentially allowing probes to be more specific in what they consider a match. For example, if a probe is relativized to search for a [participant] feature, only nominals that encode interpretable first or second person are capable of establishing the relationship that would remove the uninterpretable φ -feature from the probe. This also means that probes can 'skip over' nominals in their search domain that do not bear these features – like the external argument in (13) – in favor of lower nominals that do.



Identifying a standard set of assumptions in the case domain is arguably a bit more difficult as the assumptions outlined Chomsky (2000, 2001) aren't as widely adopted as those for φ -agreement. Chomsky argues that case features, unlike φ -features, are uninterpretable *both* on functional heads and nominals. It therefore follows that it is impossible for case features to find a matching interpretable instance – since none exist – and thus case features are not considered probes. Because they cannot probe and subsequently agree on their own, they can only be deleted if they exist on syntactic objects that have participated in another agreement relation, namely φ -agreement. In this way Chomsky explicitly connects up the domain of case and the domain of agreement by framing case assignment as the reflex of a successful φ -agreement relation. If an uninterpretable φ -probe finds a matching interpretable φ -feature on a nominal, a relationship will be established and AGREE will delete not only the uninterpretable φ -feature on the functional head, but also any uninterpretable case features that exist on both the functional head and the nominal. If a nominal agrees with the functional head T, the grammar will spell out nominative features; if a nominal agrees with the functional head v , the grammar will spell out accusative features (14). It's important to note that these operations are assumed to be syntactic and thus may or may not be morphologically marked in a particular language. For example, if a language does not overtly mark object agreement, this model would still assume that abstract object agreement obtains, but is simply not morphologically expressed.

(14)



The details outlined above aren't universally adopted, but the inability of nominals to surface without having received case is a fairly uncontroversial standard assumption. In this way the Case Filter (Chomsky, 1981; Vergnaud, 2008) which ruled out nominals that failed to receive case is maintained, although the violation is no longer restricted just to case features, but rather all uninterpretable features. Likewise, it is fairly standard to assume that functional heads are the things responsible for assigning the various cases and the set of functional heads that serve as case assigners is largely agreed upon: finite T is responsible for assigning nominative case, v is responsible for assigning accusative case, etc. Also fairly uncontroversial is the assumption that case assignment – at least in part – occurs syntactically, even for languages that do not morphologically differentiate the different case categories.

Both case and φ -agreement have a morphological function beyond their relative syntactic ones, dictating the morphological forms of nominals in the domain of case and agreement morphology on functional heads in the domain of φ -agreement. Another set of assumptions surrounds the relationship between the functions in these independent components. Generally, morphological case and agreement are assumed to be a reflection of syntactic abstract case and φ -features. Languages can vary in the degree to which they overtly reflect these abstract syntactic relationships, but they are assumed to be a universal phenomenon.

While many have modified the details of these systems to account for various things in different ways, the basic scaffold of the system remains widely adopted. Uninterpretable features must be removed in order for the derivation to be interpreted by the interfaces and both case assignment and φ -agreement are largely the result of a similar set of operations. Later in this thesis I will follow suit and likewise propose some modifications to how the system accounts for φ -agreement and case, but will remain as committed to the crucial tenets of the system as the nature of the default problem allows.

1.2.2 Defaults in the φ -agreement Domain

As we saw in the previous section, the framework established in Chomsky (2000, 2001) requires that uninterpretable features must be removed from probes via AGREE if they are to produce a grammatical sentence. What this should disallow therefore is the failure of a probe to find an agreeing goal with which to establish this relationship.

Default agreement appears to pose some problems for these assumptions if we assume defaults are the result of a probe's uninterpretable feature failing to establish an agreement relation and the subsequent removal of that feature before spell out. The apparent tolerated failure of φ -feature agreement is a wide-spread phenomenon. We've already seen data from Hindi-Urdu, repeated below in (15) that illustrates this tolerated failure. The uninterpretable φ -features on the finite T probe search in their c-commanding domain for an interpretable φ -feature bearing goal with which to agree. Being overtly case-marked in Hindi-Urdu independently prevents certain nominals from being eligible goals for the probe. This means that in (15c) the finite T cannot agree with the subject, nor can it agree with the object, which results in the failure of the grammar to remove the uninterpretable features on the finite T probe. This should cause the derivation to crash, but as we have seen, the sentence is perfectly acceptable.

- (15) a. **Mona amruud khaa-tii thii**
 Mona.F guava.F eat.HAB.F be.PRF.F.SG
 'Mona used to eat guava'

SUBJECT AGREEMENT

- b. Ram-ne **imlii** **khaa-yii thii**
 Ram.M.ERG tamarind.F eat.PFV.F be.PST.F.SG
 ‘Ram had eaten tamarind’ OBJECT AGREEMENT
- c. Mona-ne is kitaab-ko parh-aa thaa
 Mona.F.ERG this book.F.ACC read.PFV.M.SG be.PST.M.SG
 ‘Mona had read this book’ DEFAULT AGREEMENT

We see a similar pattern in Kichean, a member of the Mayan language family. Preminger (2014) shows evidence that probes in Kichean are relativized to search for [participant] bearing arguments. This is seen in (16) where first and second person arguments control agreement over third person arguments, regardless of their syntactic position. This preference is explained if one assumes that finite T bears an uninterpretable [participant] feature that can only be removed through agreeing with an interpretable [participant] bearing nominal. Essentially, this increased specificity allows the probe to ignore nominals in its search domain that do not bear this feature, third person nominals.

- (16) a. ja **rat** x-at/*-ø-ax-an ri achin
 FOC **you(sg)** COM-**2SG.ABS**/*3SG.ABS-hear-AF the man
 ‘it was you that heard the man’
- b. ja ri achin x-at/*-ø-ax-an **rat**
 FOC the man COM-**2SG.ABS**/*3SG.ABS-hear-AF **you(sg)**
 ‘it was the man that heard you(sg)’

Kichean exhibits similar behavior when both arguments are third person (17) – and therefore lack an interpretable [participant] feature – to what we’ve seen for Hindi-Urdu. Since neither argument bears the interpretable version of what the probe is searching for, neither argument is available to delete the uninterpretable [participant] feature on finite T. Once again, we’d expect this sort of data to cause a derivation crash since the derivation appears to involve the survival of an uninterpretable feature, but yet again the sentence is perfectly acceptable.

- (17) a. ja ri tz’i’ x-ø-etzel-an ri sian
 FOC the dog COM-**3SG.ABS**-hate-AF the cat
 ‘it was the dog that hated the cat.’

- b. ja ri xoq x-Ø-tz'et-o ri achin
 FOC the woman COM-3SG.ABS-see-AF the man
 'it was the woman who saw the man.'

As I mentioned in the last section, this is the primary issue that defaults raise: how is it that these derivations produce acceptable sentences when they appear to involve the failure to remove a crash-inducing uninterpretable feature?

1.2.3 Defaults in the Case Domain

Likewise, in the domain of case, what rules out a sentence like (18) is that the DP *her* is unable to value its unvalued case feature because non-finite T does not have case features to assign and there is no other source of case available.

- (18) *It is likely **her** to leave the party early.

In addition to the data we saw in section 1.1.1, examples (19)-(22) show that default case is a widespread phenomenon. Languages can differ in which case they select as the default; in most languages nominative case is the default case, however, English – along with Danish, Norwegian, and Irish – do appear to be unique in that they use accusative case, rather than nominative case to mark these default case environments (see Schütze, 2001, for a cross-linguistic survey).

- (19) **Default Nominative Case in German:**

Der/*Dem Hans, mit dem spreche ich nicht mehr.
the.NOM/*DAT with him.DAT speak I not anymore.

- (20) **Default Nominative Case in Greek:**

O paraksenos anthropos, dhen ton idhame
 the.NOM strange.NOM person.NOM not him.ACC saw
 'The strange person, we didn't see him.'

(21) **Default Accusative Case in Danish**

Hende med de blå øjne er svensker
her.ACC with the blue eyes is a.Swede
'She/Her with the blue eyes is a Swede.'

(22) **Default Accusative Case in Irish**

Rinne Eoghan **é** féin **é**
did Owen **him.ACC** EMPH it
'Owen himself did it.'

(Schütze, 2001)

As was true in the φ -agreement domain, the failure of the derivation to value the case features on nominals should predict that the derivations produce ungrammatical structures. Instead, what we see are perfectly acceptable sentences. Similar questions are raised: how does the grammar produce defaults despite this failure, and how does the grammar constrain the production of those defaults so that they don't erroneously appear in places like (18) where the failure to value case *does* induce ungrammaticality?

1.3 Jumping Ship lands us in Bizarre Boats

The deep nature of this default problem has in part prompted researchers to propose a number of solutions that require quite radical departures from the standard set of theoretical assumptions. At their core, these departures share the same logic: if we remove the theoretical power of failed feature valuation, we remove at least the primary issue that defaults raise for the grammar – how defaults are produced. To address issues of grammatical failed φ -agreement, Preminger proposes an entirely new model for derivations: his (2014) obligatory operations model. Under the assumptions he proposes, the failure to value features does not trigger ungrammaticality at all; rather it is the failure to trigger a set of obligatory operations that is responsible. In addressing default case, among other issues surrounding the assignment of morphological case features, researchers have found promise both in adopting an alternative model of case valuation called dependent case theory and in abandoning the long-held view that case features play a role in regulating nominal distribution.

It is important to say that while both types of proposals do constitute radical departures from the traditional set of theoretical assumptions, neither could be considered fringe proposals as both have seen recent mainstream adoption, especially the dependent case model. The specifics of these departures will be addressed in greater detail in future chapters, but I'd like to provide a quick preview of the approaches here.

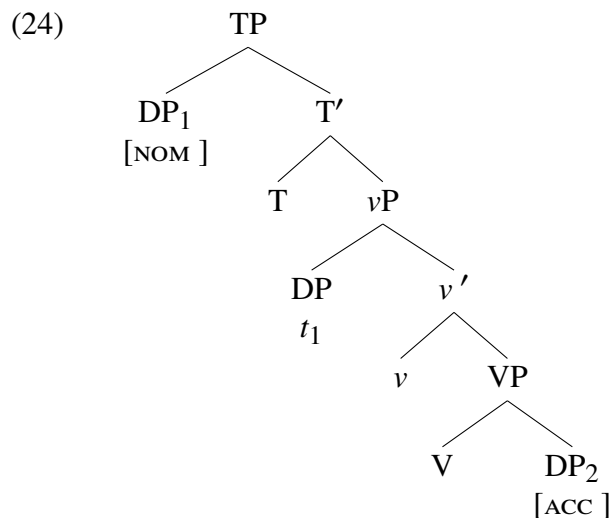
1.3.1 Dependent Case Theory

Dependent case theory is an alternative model of case valuation that assigns case configurationally by examining not the relationships between functional heads and nominals, but rather the relationships between the nominals themselves. Case is assigned to a nominal if that nominal exists in a certain configuration with respect to other nominals in the same domain. A simplified example is shown below in (23). The algorithm in (23) says that accusative case features are assigned to a nominal only if that nominal is c-commanded by another nominal in the same TP spell out domain. DP_2 in (24) therefore would be assigned accusative case features because it is c-commanded by another nominal, DP_1 that shares its TP spell out domain. Nominative case features are instead assigned by default² to nominals that do not exist in that configuration.

(23) *Dependent Case*

If there are two distinct NPs in the same spell-out domain such that NP_1 c-commands NP_2 , then value the case feature of NP_2 as ACC unless NP_1 has already been marked for case.

²Dependent case theory does outline a distinction between unmarked case and default case, the details of which we'll explore in chapter 2.



Under these assumptions, DP_2 receives accusative case features not because it stands in some relation to the v of the clause, but rather because it stands in some configuration relative to another DP. DP_1 by contrast receive nominative features because it does not stand in that configuration relative to another DP. Dependent case theory therefore redefines case as the reflection of a relationship between two nominals in a given domain.

Nominative features in this system are assigned in the absence of a configuration and in this way they share the kind of intuitive logic we assign to defaults more generally. It's this default nature that makes dependent case theory an attractive possibility when confronted with the default case data. Given the default nature of unmarked case, dependent case theory is not consistent with a framework that rules derivation ungrammatical if nominals fails to receive case. Proponents of this model therefore are forced into adopting an additional theoretical departure: the separation of case from licensing, which we'll preview next.

1.3.2 Separation of Case from Licensing

A perfectly reasonable way to handle the production related problem of default case is to assume that the failure to value case features does not have the power to rule derivations ungrammatical. This requires suspending the long-held assumption that receiving case is a grammatical requirement for

nominal licensing and thus requires alternative explanation for the varied set of data shown below in (25). The need for case was what primarily drove movement (25a), what prevented superfluous movement (25b), what explained the inability of non-finite clauses to host overt subjects (25c), and what explained the distribution and form of nominals in passives (25d) and unaccusatives (25e), among other things.³

- (25)
- a. John_i is likely t_i to win the race.
 - b. *John_i is likely that t_i will win the race.
 - c. *It is likely him to win the race.
 - d. John_i was invited t_i.
 - e. John_i arrived t_i.

With the adoption of the EPP feature, movement of the theme argument in passives (25d) and unaccusatives (25e) to the subject position no longer needed to be tied to the need for case on the theme argument. Likewise, the adoption of phase heads and spell out domains further reduced the role of case in regulating superraising (25a)-(25b). While we have theoretical tools to provide alternative explanations for some of the data in (25), data like that in (26) is arguably much more difficult to explain without reference to the failure to receive case.

- (26)
- a. *John hoped him to win the lottery.
 - b. *It is likely her to leave the party early.

We'll see in chapter 2 that those who wish to eliminate case's role in regulating nominal licensing have proposed an extension of the Empty Category Principle – the idea that overt complementizers cannot precede empty categories – to account for the type of data in (26). Their view is essentially that this data is the last frontier for the classical case theory and that if we can propose a reasonable alternative, we can make an argument that the assumption that case must be valued can be eliminated.

³This is not intended to be an exhaustive list, just a summary of some of the big facts.

1.3.3 Obligatory Operations

Preminger (2014) uses a similar logic with respect to modeling what look like grammatical agreement failures. He takes the position that what default φ -agreement data shows us is that the grammar *does* tolerate the failure to receive a feature value and therefore we must recast our assumptions about what enforces grammatical requirements. He proposes that what is required of a derivation is not the successful removal of uninterpretable features, but instead that all operations that are obligatory must be initiated. In simple terms, it's not the outcomes that the grammar cares about, but rather that all obligatory processes have been attempted. To replace AGREE, he proposes FIND (27) which requires that a probe search for an accessible goal with which to agree. However, its failure is completely tolerated so long as it was initiated in every context it could. When it does fail, the grammar is able to insert default features on the relevant functional heads.

(27) FIND(f)

Given an unvalued feature f on a head H^0 , look for an XP bearing a valued instance of f and assign that value to H^0 .

What all of these theoretical departures share is the claim that the failure to value and subsequently remove uninterpretable features is not a crash-inducing circumstance. Of course, since the assumption that failure-to-value is fatal has been one of the central tenets of frameworks standardly adopted in the Minimalist era, abandoning it constitutes a dramatic departure which will predictably have great effect across a number of syntactic domains.

1.4 Goals of this Thesis

This thesis will broadly argue that by jumping ship, so to speak, we've landed in some problematic lifeboats that are headed in the wrong direction. If we can instead put out the fire that caused us to jump ship in the first place, we can maintain the course. There are two main claims that I will advance:

- (i) There are some serious conceptual and empirical problems that arise from dependent case theory and obligatory operations that warrant their rejection.
- (ii) There is a solution available that allows us to maintain the basic scaffold of the standard framework and solves both the production and the constraint problems that defaults introduce.

Most of the discussion surrounding claim (i) will focus on conceptual arguments against these proposals and will also center along the claim that while these departures provide a mechanism for the production of defaults, the mechanisms available don't constrain that production very well. The discussion surrounding claim (ii) will involve proposing a solution to the default problem that fits within the basic tenets of the standard framework and advancing the argument that its adoption should be favored over the departures.

Chapter 2 and chapter 3 will advance the first main claim of this dissertation: that the radical departures that the existence of defaults has triggered are more problematic than they first appear. The goal here is to argue that there is reason to revisit the standard approach, despite recent calls for its abandonment. Chapter 2 will outline the radical departures involving the role of case in the regulation of DP licensing and the mechanisms that the grammar has available to assign it. I will show that while the dependent case theory of morphological case assignment covers a wide range of empirical data, it suffers from serious conceptual issues that make its adoption unattractive. First, I'll show how modern versions of the model induce an Inclusiveness Condition violation with respect to the assignment of case feature values. This violation is serious not because of strict obedience to Minimalism, but rather because it makes our understanding of which syntactic objects actually house case features unclear. I'll also argue that under the dependent case model, case does not reflect a consistent relationship between syntactic objects, undermining its classification as a system. Here, I also discuss issues that the model raises for the structure of case features themselves and how we want to understand the limits of parameterization. I then present some empirical facts that are also difficult to model under the dependent case approach (although the bulk of that discussion happens in chapter 4). The chapter concludes with an argument that the explanations

that are proposed to pivot away from assuming a licensing role for case are insufficiently fleshed out, leaving room for a proposal that models default case within the boundaries of a system that treats unvalued features as fatal to derivations.

Chapter 3 will take a similar approach, instead investigating the notion of obligatory operations, as proposed in Preminger (2014). As was true of chapter 2, the goal of this chapter will be to argue that there are some serious issues – both empirical and conceptual – with the adoption of obligatory operations, enough that modifications to the standard system are warranted. I show that the obligatory operations approach to φ -agreement is not necessitated by default agreement data by outlining an alternative proposal made by Béjar (2003) that operates within a more standard framework. I then present arguments that show that when we examine data more complicated than what's used to motivate the obligatory operations model, we are unable to fully account for the varied outcomes of failed agreement by simply allowing operations to fail. A major claim of this chapter is that agreement does not have a binary set of outcomes and thus needs to be accounted for with a model that can capture this fact. The obligatory operations approach is unable to do so, once again making room for a proposal of defaults that is more in-line with standard assumptions.

Chapter 4 will advance the second main claim of this dissertation: that in light of these issues, we should instead pursue a more modest approach, one that largely maintains the standard set of theoretical assumptions surrounding case and agreement that have provided many insights over many decades of research in these areas. This proposal will focus on accounting for the production of default case, how we constrain that production, and will provide clarification on some of the issues surrounding how case features are modeled. I argue for a novel understanding of case features and show how an approach similar to Béjar (2003) can operate over these features to produce the three-way set of outcomes observed in the data: canonical case, default case, and ungrammaticality. Chapter 4 will also show that this solution should be preferred over the radical departures discussed in chapters 2 and 3. The main claim is that the theoretical concessions that dependent case theory, separation of case from licensing, and obligatory operations require us to make are severe enough to warrant their rejection, despite the empirical coverage benefits they offer. Chapter 5 will conclude.

CHAPTER 2

DEPENDENT CASE THEORY

This chapter explores the theoretical implications that follow if one adopts either the separation of case from licensing or the dependent case model of case valuation. The hope is to provide arguments that validate a reinvestigation of the more standard approaches that these radical departures reject, saving proposals of an alternative for chapter 4. While these next two chapters may seem trivially negative, it is important to motivate that there is reason to return to more standard approaches, especially since these radical departures are largely motivated on claims that the more standard approaches aren't tenable. We begin with a discussion of the first set of departures intended to address the problematic issues that are illustrated by default case. Recall that the crux of the issue is that a system that enforces grammatical requirements in part through the valuation of case features should be unable to handle grammatical instances where nominals survive with their case features unvalued. This type of data is in part addressed by the adoption of the dependent case model of case valuation, the topic of section 2.2, and the separation of case from licensing, which will be the focus of section 2.3. The conclusion reached in this chapter is that these departures require adopting theoretical systems that are further from Minimalist ideals than their more standard counterparts and thus validate an attempt to modify the more standard approaches in ways that address the problems that defaults introduce.

2.1 The Default Case Issue

Case has had a central role in standard syntactic frameworks since Vergnaud's famous letter to Chomsky in 1977 suggesting that we can use case to provide an explanation for nominal distribution (Vergnaud, 2008). This revolutionary idea birthed the Case Filter, shown in (1), which stated that the only licit NPs were ones that received case from somewhere else in the structure (Chomsky, 1981).

(1) **Case Filter:**

*NP that does not have Case

Case is assumed to have two primary functions: (i) it regulates nominal distribution via licensing and (ii) it provides the morphological marking of NPs. These two functions, called Abstract Case¹ and morphological case respectively, are standardly assumed to be related; morphological case is the physical realization of Abstract Case features. Languages vary in how richly they express this relationship with some languages having rich morphological case systems and others failing to make any case distinctions at all. One of the most revolutionary aspects of Vergnaud's original proposal was that he argued that Abstract Case was a part of UG; that even languages without a rich morphological case system still obeyed the Case Filter, requiring that all nominals must be licensed by receiving Abstract Case.

In modern versions of case theory, case is modeled as the reflex of successful φ -agreement (Chomsky, 2000, 2001). When a nominal values the uninterpretable φ -features on a particular functional head like finite T, it has its own uninterpretable case features valued as a result. The functional head with which the nominal agrees determines which morphological case category the nominal will express – nominative if agreement is with a finite T, accusative if agreement is with a *v*. However, if a nominal instead exists in a position where it is unable to establish a successful φ -agreement relationship with a full set of φ -features, case assignment – as a reflex of this relationship – will also fail. The result of that failure is ungrammaticality unless that nominal is unable to establish an alternative φ -agreement relationship, perhaps with an ECM embedding verb, as shown in (2a). What rules out a sentence like (2b), therefore, is the failure of the DP *her* to

¹A quick note on terminological conventions: the standard convention for distinguishing grammatical licensing from morphological case is to use capital “C” Case to refer to the former and lowercase “c” case to refer to the latter. In this particular thesis, however it is often necessary to refer to a more general understanding of case as either the combination of the two or to be agnostic about their relationship. To avoid copious use of the hard to read “C/case”, the following conventions will be used: when specifically referring to grammatical licensing I will use capital “C” Case and when specifically referring to morphological form I will use the term “morphological case”. When a distinction is either not needed or is not assumed I will use lowercase “c” case to refer to case in its general form.

have received case since it can't from non-finite T *and* there is no other source of case available. In this way, Case has been crucial for understanding how derivations deliver grammatical sentences.

- (2) a. I expect **her** to leave the party early.
b. *It is likely **her** to leave the party early.

However, as we've seen in chapter 1, there are a number of structures which like (2b) also contain a DP that has failed to receive a case value, but *unlike* (2b) produce a perfectly grammatical sentence. These are the instances where what is called default case surface, shown in (3):

(3) **Default Case in English:**

- a. *Hanging Topic/Left-Dislocation*

What?! **Him** wear a tuxedo?!

- b. *Gapping*

She will eat cake, **him** brownies.

- c. *Coordination*

Me and **him** will go to the store.

- d. *Modified Pronouns*

Lucky **me** has to clean all the toilets.

(Schütze, 2001)

What makes default case theoretically interesting is that it appears to constitute a counterexample to the Case Filter by virtue of being an instance where required case valuation has failed. At least two questions are raised: (i) how is default case *produced* in the first place, given that the failure to value case features should result in ungrammaticality and (ii) how is the production of default case *constrained* such that its overapplication couldn't incorrectly produce a grammatical version of (2b)?

2.2 Dependent Case Theory

We begin with a departure that involves adopting a model of case valuation that builds defaults directly into the system – an approach called dependent case (Baker, 2015; Levin & Preminger, 2015;

Marantz, 1991; McFadden, 2004). Dependent case theory assigns case using the configurational relationship between two nominals in a case assigning domain. If a nominal is not assigned one of the dependent cases, it may then be assigned either unmarked case or default case. The negative characterization of the environments in which unmarked and default cases are assigned should be familiar as it is similar in logic to how we intuitively define the environments in which defaults appear and is thus considered an attractive way to handle the problems raised by the existence of defaults.

This section outlines the details of dependent case theory and explores some of the empirical and conceptual implications that arise from its adoption. In section 2.2.1 we begin with a brief history followed by a overview of modern versions of dependent case theory. In section 2.2.2, I discuss some important implications that adopting this system requires of the grammar and then use them to argue there is still reason to pursue a more conservative standard approach. The details of that approach will be discussed in chapter 4.

2.2.1 Overview of model

2.2.1.1 Early Versions

The origins of dependent case predate both Minimalism itself (Chomsky, 1995) and standard Minimalist assumptions regarding case and agreement (Chomsky, 2000, 2001). Understanding these initial motivations and how this novel system compared to its contemporaries is important for understanding how the modern versions of dependent case fit into the larger theoretical picture. We begin therefore with one of the original versions of dependent case, Marantz (1991).² Marantz has two primary goals: (i) to rid the syntax of abstract Case and (ii) to contribute significant knowledge about the theory of morphological case.

²There are two other works that could also be considered pioneers of dependent case: (Bittner & Hale, 1996; Yip, Maling, & Jackendoff, 1987). I focus on (Marantz, 1991) here because the modern versions of dependent case are primarily based on his version. However it is important to note that many of the insights we gain from Marantz are echoed in these other works as well.

We start with data that illustrates what is called the ergative generalization, defined in (4) (Marantz, 1991). The data in (5) from Hindi (Marantz, 1991) shows that ergative case is only possible on subjects that originate in a thematic subject position – either transitive subjects (5d) or subjects of unergatives (5b)-(5c). Ergative case is disallowed on subjects that originate in a non-thematic subject position, like they would in an unaccusative (5a).

(4) *Ergative Generalization*

No ergative case on a non-thematic subject (ie: on an argument moved into a non-thematic subject position)

- (5)
- a. siita (*ne) aayii
Sita.FEM (*ERG) arrived/came.FEM
'Sita arrived.'
 - b. kutte bhoNke
dogs.MASC.PL barked.MASC.PL
'Dogs barked.'
 - c. kuttoN ne bhoNkaa
dogs.PL ERG barked.MASC.SG
'Dogs barked.'
 - d. raam ne roTii khaayii thii
Ram.MASC ERG bread.FEM eat.FEM be.PAST.FEM
'Ram was eating bread.'

Data that reflects the ergative generalization mirrors in part what Burzio's Generalization (6) hoped to capture: the inability for the object of an unaccusative to receive accusative case (7) (Burzio, 1986). At the time, Burzio's generalization was understood to be about the abstract syntactic Case that a nominal received, while the ergative generalization covered morphological case only. In order to draw a connection between the two, Marantz reframed Burzio's generalization to be about morphological case, allowing for both generalizations to be subsumed under the same morphological mechanism, thus contributing strongly to our understanding of morphological case theory.

(6) *Burzio's Generalization:*

If a verb's subject position is non-thematic, the verb will not assign accusative structural Case.

- (7) a. He_i arrived t_i.
b. *Him_i arrived t_i.

Of course, to do this, Marantz needed to make it attractive to assume that data like (8) could be captured without reference to abstract Case. Under the standard Burzio explanation, what rules out the examples in (8) (couched in modern terms) is the inability of both unaccusative and passive *v* to assign abstract accusative Case to the nominals *the man* and *the porcupine*, respectively. Since the *v* is unable to assign the necessary case, the derivations arrive at spell out having failed to value their abstract Case features, causing ungrammaticality.

- (8) a. *It arrived the man.
b. *It was purchased the porcupine.

If however, one assumes that we couple a requirement that subject positions must be filled – the Extended Projection Principle (Chomsky, 1982) – with a preference for move over merge as the strategy to obey the EPP, we can capture the data in (8) without referring to abstract case at all. What rules out the sentences in (8) under this view is the failure of the derivation to obey the move over merge preference, since both derivations involve the merger of an expletive instead of the movement of the theme argument to subject position.

Part of being able to reject the notion of abstract Case altogether requires some alternative understanding of Burzio's generalization, since it directly categorizes when and where abstract Case is assigned. Marantz proposes that instead of being about abstract Case, Burzio's generalization is actually about the assignment of morphological accusative case in particular – aligning it much more closely with the ergative generalization (9). The comparison drawn between these two generalizations gave birth to dependent case theory as we know it today, an influential contribution that still lies at the center of much debate in the modern case literature. Like modern versions

of dependent case theory, Marantz's original version is crucially only intended to explain how morphological case assignment works, as he (and those who've followed him since) rejects the idea that abstract Case (i) exists and (ii) has anything at all to do with the regulation of nominal distribution.

(9) a. *Ergative Generalization:*

no ergative case on a non-thematic subject (ie: on an argument moved into a non-thematic subject position)

b. *Burzio's Generalization (reframed):*

no accusative case on an object in a sentence with a non-thematic subject position

The primary question that Marantz – and others who work on morphological case – need to address is this: what is it that determines which particular case features show up on which nominals and why? It is common to assume that the morphological component, responsible for assigning morphological case, interprets the syntactic structure delivered to it. So although Marantz assumes that case assignment happens entirely in the morphological component, it is still the structural relations between relevant pieces that dictate the choice between the various cases. What is different under dependent case theory are the mechanisms by which these relationships are established. Marantz proposes a hierarchy that dictates the order in which different cases take precedence over others, shown below in (10):

- (10) a. lexically governed case
b. dependent case (accusative and ergative)
c. unmarked case (environment-sensitive)
d. default case

First, lexical cases are assigned to nominal chains by virtue of being governed by a verb that has quirky or lexical case to assign. The fact that it is the nominal chain, rather than just the highest position the nominal occupies that is relevant for government is how we capture Icelandic

data where quirky case is preserved despite movement to a subject position. To illustrate see the example below in (11) (Harley, 1995). Since V will still always govern part of the chain of the nominal, even after the nominal moves, V is still capable of assigning case to that nominal. In this way, a quirky case-assigning verb is able to assign quirky case to a subject because it governs a position that the subject once occupied. This accounts for the preservation of quirky case under movement.

- (11) Morgum studentum liki/*lika verkið
 many student.PL.DAT like.3SG/*3PL job.the.NOM
 ‘many students like the job’

If a lexical case assigning verb is not present in a given structure, then the V+I complex is able to assign case features to nominals that it governs. Case assignment is determined by looking at not only the particular nominal in question, but also other nominals that the same V+I complex governs. This case, called dependent case because its assignment is dependent on the presence of other nominals, is either accusative or ergative and is assigned according to the following algorithm, shown in (12):

- (12) Dependent case is assigned by V+I to a position governed by V+I when a distinct position governed by V+I is:
- a. not “marked” (not part of a chain governed by a lexical case determiner)
 - b. distinct from the chain being assigned dependent case
- dependent case assigned up to subject: ergative
 dependent case assigned down to object: accusative

(Marantz, 1991)

In plain terms, the V+I complex assigns dependent case to a nominal if there is another distinct nominal that either c-commands it (if the language exhibits nominative-accusative alignment) or that it c-commands (if the language exhibits ergative-absolutive alignment). In the English example in (13a) we see that the nominal *her* is c-commanded by another nominal *he*. In GB frameworks,

these two nominals represent distinct chains, but are both governed by the same V+I complex. Since *he* is not lexically case marked, the dependent case algorithm in (12) would direct the V+I complex to assign dependent accusative case down to the object, *her*. The same mechanism applied in the opposite direction is shown for data like (13b) from Burashaski (Willson, 1996). Because Burashaski is an ergative language, the dependent case mechanism would direct the V+I complex to assign dependent case upward, to the subject. As in (13a), there are two nominals in (13b) that represent distinct chains, both governed by the same V+I complex. This time, the dependent case algorithm would direct V+I to assign dependent case upward to the subject *hilés-e*, marking it with the dependent case for ergative-absolutive languages – ergative case. In this way, the case that a nominal receives is dependent on whether or not there is another nominal with particular properties (not already case-marked) in a given local domain (the governing domain of V+I). In many ways, we can view dependent case as the logical successor of Burzio’s generalization: at their core, both essentially describe the assignment of accusative case as dependent on the presence of a higher, distinct argument.

- (13) a. **He** saw **her**
 NOM saw **ACC**
- b. **Hilés-e** **dasin** mu-ye’ets-imi
 boy-ERG girl.ABS 3.F-SEE-PAST.3MSG
 ‘The boy saw the girl’

(Willson, 1996)

Next, if there exists a nominal to which the dependent case algorithm does not apply, the nominal is eligible to receive an unmarked case. Unmarked case is assigned to nominals after both the mechanisms behind lexical case and dependent case have applied and is context sensitive in that different environments trigger different cases. Nominative and absolutive are assumed to be the unmarked cases assigned to nominals that don’t receive dependent case in the TP environment, while genitive case is the unmarked case assigned to nominals that are within another NP environment. After the dependent case algorithm has applied to all the nominals it can in (13a), the nominal *he*

remains. Since this nominal is in a TP environment, it receives the unmarked nominative. Likewise, in (13b), after the dependent case mechanism assigned ergative case to *hilés-e*, the nominal *dasin* remains unmarked. Since *dasin* is in the TP environment, it would receive unmarked absolutive case.

Lastly, if there exists a nominal such that none of the above case assigning rules in (12) are able to apply, a more general language-wide default that is not contextually sensitive is assigned – the default case introduced in chapter 1. Case assignment in the original proposal of this system is assumed to occur post-syntactically in the morphological component, but see Levin (2015) for work that places dependent case in the syntax itself. The existence of a default case that applies as a last resort to nominals that have not received either lexical, dependent, or unmarked case is why dependent case is categorically incompatible with a framework that affords case the ability to act as a grammatical filter in the syntax – it would render that filter meaningless.

2.2.1.2 Modern Versions

The impact of Marantz (1991) is still strongly felt today in modern versions of dependent case theory. This next section will describe what current versions of this model look like and outline the relevant updates that have been made to Marantz’s original proposal to bring it in-line with modern theoretical assumptions. From there, we can discuss the theoretical and conceptual implications that adopting this model appears to require. The eventual goal is to provide arguments against adopting this method of case valuation (despite its impressive empirical coverage), saving an illustration of what a reasonable alternative could look like for chapter 4.

Since 1991, the standard framework in which we build our theories has changed significantly and with it the standard assumptions we hold about the nature and relationship of case and licensing. Unsurprisingly, the ways in which the field has changed require commensurate changes in how dependent case is implemented today. Most significant is the abandonment of government as a syntactic relation. Modern versions of dependent case theory have therefore had to redefine the conditions upon which the dependent case rules apply because Marantz defined the assignment of

dependent case in exactly those terms. Today, proponents of dependent case theory largely adopt the same set of assumptions and details regarding the assignment of dependent case, most of which come out of Baker (2015). I will follow suit and thereby focus our discussion on this version, pointing out relevant departures when needed.

Baker (2015) provides the most comprehensive analysis of how dependent case could operate within a modern Minimalist framework and in doing so, constitutes the first system with enough detail that it is now a true modern theoretical competitor to the more standard agreement-based approach first outlined in Chomsky (2000, 2001). Baker's model is a hybrid one, combining elements from both agreement-based case proposals and dependent case ones and is focused on morphological case issues only, leaving explanations of nominal licensing to others – an important distinction we'll see is relevant later in this chapter. The defining feature of his approach is the high degree of parameterization, something that allows him to capture an impressive amount of empirical coverage across a wide range of language families and case system types. This high degree of parameterization is reflected both on a narrow scale in the specific details of dependent case assignment and also on a more broad scale, being reflected in the choice of assignment mechanisms themselves.³

Since the highest level of parameterization is in the choice of assignment mechanisms themselves, we'll begin our discussion there. Baker's main thesis is that the variety in the morphological case patterns found in the world's diverse languages supports a model of case assignment that provides the grammar with two⁴ mechanisms by which case features can be valued/assigned: one that is agreement-based⁵ and one that is configurational, called dependent case. The motivation

³This last point is the singular source of major contention between most modern researchers. While Baker argues that we still need remnants of the standard agreement-based mechanism, there are those who would rather eliminate any need for reference to agreement in assigning case (Levin & Preminger, 2015). A more thorough discussion of the theoretical implications of this disagreement will follow in the next section.

⁴I say "two" here, putting aside for now issues regarding lexical case. We will come back to that point later.

⁵A quick note for clarification: the sources discussed in this chapter are arguing for and against a very particular model of standard case assignment. The relevant approach is the one proposed in

for this hybrid system is that where one model has been weak, the other is strong. Baker argues that agreement provides a solid theory of morphological case for “some cases in some languages”, but not for “all cases in all languages” (Baker, 2015, p.47). For those cases or languages where agreement does not account well for the various morphological patterns, the grammar needs to have an additional method of case assignment available in order to capture that data. This choice in case assigning mechanism is not a parameter that is always set on a language-level; it can actually be set case-by-case. In other words, there can exist languages like Sakha in which some cases – nominative and genitive – are assigned via agreement, while others – accusative and dative – are assigned via dependent case.

A strong argument for maintaining agreement-based case is the long observed strict co-occurrence between nominative case and φ -agreement. Baker (2015) and Baker and Vinokurova (2010) use the language Sakha to illustrate this point. In Sakha, the nominative subject typically controls verb agreement; this is what we see in (14).⁶

- (14) **Masha aqa-ta** kinige-ni atyylas-**ta**
 Masha father-3SG.NOM_{poss} book-ACC buy-PAST.3SG_{subj}
 ‘Masha’s father bought the book.’

Subject agreement is also present in relative clauses in Sakha. Relative clauses consist of a participle that precedes a head noun along, of course, with a subject (15). In relative clauses, the standard

Chomsky (2000, 2001) where case assignment occurs as a reflex of a nominal having established a successful φ -agreement relationship with a probe. This is distinct from later models that use the AGREE operation to establish case assignment but divorce it from φ -agreement itself (Adger, 2003; Carstens, 2016, among others) Since the topic of this chapter is dependent case and its proponents, I intentionally use the phrase *agreement-based case* as opposed to AGREE-based case in order to distinguish the two.

⁶A quick comment on the notation used here: for the data from Sakha, I’ve included some annotation Baker used to mark what certain elements have agreed with. This is especially helpful since the morphological facts aren’t especially obvious in this language. These additional annotations are subscripted on the element that has been agreed with and are either SUBJ, OBJ, or POSS. To illustrate, if a verb is subscripted with a SUBJ as it is in (14), this shows agreement has successfully been established with the subject, *Masha’s father*’. Since in (14) both the subject and object are third person, this helps to clarify what is going on in the data. Likewise, if something is subscripted with POSS, as *‘father’* is in (14), this tells us that it has agreed with a possessor.

agreement between the subject and the verb that we saw in (14) is not allowed, as we can see in (15b) where *ih-er* cannot agree with the subject of the relative clause ‘*Masha*’. What is allowed however, is agreement between the subject and the head noun of the relative clause; in (15a), *caakky-ta* agrees with the subject of the relative clause *Masha* and the relative clause is grammatical. What is important about this data with respect to case assignment is the fact that *Masha* in (15a) is argued to be genitive, not nominative.

- (15) a. *Masha* *cej ih-er* *caakky-ta*
Masha *tea drink-AOR cup-3SG_{poss}*
‘a cup that *Masha* drinks tea from’
- b. **Masha* *cej ih-er-e* *caakky*
Masha *tea drink-AOR-3SG_{subj}* *cup*
‘a cup that *Masha* drinks tea from’

Of course, this observation is not immediately obvious as both nominative and genitive are typically expressed as null morphemes in Sakha. However, Baker provides a comparison that helps buttress this claim. Compare (14) and (16). In (16), *Masha* agrees with the head noun *at-a*, but instead of the subject *Masha aqa-ty-n* receiving nominative case as it does in (14) when it agreed with the verb, it receives genitive case.

- (16) *Masha* *aqa-ty-n* *atyylas-pyt at-a*
Masha *father-3SG_{poss}-GEN* *buy-PTPL* *horse-3SG_{poss}*
‘the horse that *Masha*’s father bought’

Baker uses this data to show that when subject agreement is with a verbal element we get nominative case and when subject agreement is with a nominal element, we get genitive case. Furthermore, when agreement is totally absent (17), we either get a null subject (17a) or the clause is ungrammatical (17b).

- (17) a. *cej ih-er* *caakky*
tea drink-AOR cup
‘a cup that one drinks from’

- b. *Masha cej ih-er caaky
 Masha tea drink.AOR cup
 ‘ a cup Masha drinks from’

This data signals a strong correlation between overt subject-verb agreement and nominative case. When φ -agreement between the subject and the verb is present, nominative case surfaces (14) and when φ -agreement between the subject and the verb is not present, either because it has agreed with something else (15a) or has failed to agree at all (17), nominative case is disallowed.

Additionally, example (18) further illustrates the close relationship between successful subject-verb φ -agreement and the expression of nominative case. In Sakha, the theme of passive verbs can be either accusative or nominative. When it is nominative, as it is assumed to be in (18a), we see φ -agreement between subject and verb. However, when the theme argument is accusative, as it is in (18b), φ -agreement does not surface; we can see that the verb in (18b) is marked with singular features, rather than the plural features that would surface had agreement been successful.

- (18) a. **Sonun-nar** aaq-lyln-ny-lar
 news.PL read.PASS.PAST.3PL_{subj}
 ‘the news was read’
 b. Sonun-nar-y aaq-lyln-na
 news.PL.ACC read.PASS.PAST.3SG_{subj}
 ‘the news was read’

These examples (among others shown in both Baker and Vinokurova (2010) and Baker (2015)) illustrate the following descriptive principle for how nominative case is distributed in Sakha:

- (19) Overt NP X has nominative case if, and only if, exactly one verbal form in the clause containing X agrees with it.

(19) reflects a dependency between φ -agreement and the appearance of nominative case. Since the standard traditional agreement approach models case as the reflex of having established a successful agreement relationship, it accounts for (19) quite directly and it is therefore easy to understand why Baker concludes that there are cases in some languages that should still be modeled with an

agreement-based case mechanism.⁷

There are also cases and languages for which Baker argues an agreement-based approach is the wrong approach. The arguments against agreement-based case are quite similar in logic in that it's hard to adopt an agreement-based case account if there is either (i) doubt that the agreement operation exists, (ii) serious mismatches between the agreement system and the case system, or (iii) the presence of case marking in the absence of the functional head assumed to be responsible. To address this first point, there is a large number of languages which exhibit overt case marking, but do not exhibit any sort of verbal agreement, making it difficult to make a connection between the two. For example, Japanese has a fairly robust case-marking system, but does not appear to have any sort of verbal agreement for any of the typical φ -feature categories like person, number, or gender. Baker couples this fact with the existence of other proposals that use a complete lack of agreement to account for other phenomena (Kuroda, 1988) to argue that an agreement-based account seems very unattractive for languages that don't appear to have it as an active operation.

Similarly, there are also a number of languages that may have φ -agreement more generally, but do not exhibit any object agreement specifically. To the extent that these languages do exhibit accusative case, it becomes difficult to attribute that accusative case to non-existent object agreement. Addressing the second argument, Baker illustrates some mismatches between case and agreement that appear to make it difficult to adopt an agreement-based account of case assignment. A quick example is shown in (20). What (20) shows is that in Amharic, we observe successful object agreement with nominals that are differently case-marked (Baker, 2012b; Leslau, 1995). In (20a), there is object agreement with the dative argument *Almaz*. In (20b), a nominative argument controls the object agreement and in (20c) an instrumental argument controls object agreement. One might then conclude that we should not ascribe the particular case-marking on each nominal

⁷Baker acknowledges that a dependent case alternative to nominative case in Sakha has been proposed (see Levin and Preminger (2015) for details), but he maintains a preference for a more traditional agreement largely because the dependent case alternative requires a number of stipulations that are not necessary if one follows Baker's proposal and, as Baker points out, are not universally true. However, it is important to acknowledge that there are dependent case alternatives that are able to capture the data to some degree

to the relationship formed by object agreement, since this relationship is not consistently reflected by the same case.⁸

- (20) a. Ləmma **l-Almaz** məts'əhaf-u-n sət't'-at
 Lemma.M **DAT-Almaz.F** book.M-DEF-ACC give-3_{M_{SUBJ}}-3_{F_{OBJ}}
 'Lemma gave the book to Almaz' (Baker, 2012a)
- b. **Aster** wiʃʃa all-at
Aster.F dog.M exist-3_{M_{SUBJ}}-3_{F_{OBJ}}
 'Aster has a dog.'
- c. Aster **bə-mə t'rə giya-w** dədʒdʒ t'ərrəg-ətʃtʃ-iʃb-ət
 Aster.F **INST-broom.M-DEF** doorway sweep-3_{F_{SUBJ}}-with-3_{M_{OBJ}}
 'Aster swept a doorway with the broom.' (Leslau, 1995)

To address the third piece of the argument, Baker asks: what happens when the heads that are assumed to be responsible for case assignment go “missing”? If we examined English for an answer, we might conclude that an agreement-based approach is more successful after all. (21) shows that when finite T is present, nominative case is grammatical. However, when finite T is absent, nominative case is impossible, but other case-markings are grammatical. This kind of data has famously been used to support the idea that finite T is responsible for the assignment of nominative case (Chomsky, 1981; Vergnaud, 2008).

- (21) a. He will find some money in the park.
 b. [PRO/for him/*he to find some money] would be a lucky break.
 c. [PRO/Him/His/*He finding some money in the park] was a big help to his budget.

(Baker, 2015)

However, as Baker illustrates, this fact is not universal. In (22) from Tamil, we see an ability for subjects to appear with nominative case, despite a lack of finite T. This shows that whatever is responsible for nominative case assignment in languages like Tamil, it is not finite T. (See McFadden

⁸See Baker (2015) and McFadden (2004) for more examples that illustrate case/agreement mismatches. Baker (2015) extends this point by showing how ergative languages are especially robust in their agreement/case mismatches.

and Sundaresan (2010) for a thorough discussion of this data).

- (22) Champa-vukku [Sudha oru samosa-vai saappid-a] venḍ-um
 Champa-DAT Sudha.NOM a samosa-ACC eat-INF want-3_{N_{SUBJ}}
 ‘Champa wants Sudha to eat a samosa’

Taking these arguments together, Baker advances the proposal that case marking is – at least some of the time – not dependent on the establishment of an agreement relationship. He does admit that it’s possible to propose accounts that rely on notions of abstract agreement that morphologically mark syntactic objects in ways that belie any syntactic relationship; however he pursues a different route, asking if we shouldn’t trust more seriously what the surface morphology is telling us. The model of dependent case he proposes is an exploration of what an alternative system for languages that aren’t as conducive to an agreement-based approach could look like.

Before moving forward with the details of how dependent case can be assigned in the grammar, it’s important to clarify where lexical/quirky/inherent case fits into this system since I’ve been describing the case assignment mechanism parameter as having just the two options. Baker follows fairly standard assumptions about lexical case, arguing that it applies immediately, via the relationship an argument forms by being merged into a projection with a quirky case assigning verbal head. Any assignment of either dependent case or agreement-based case comes after. More discussion about the timing of this system will follow.

In its most general terms, dependent case assignment proceeds according to the following abstract schema in (23) where each variable represents an area where there is some degree of parameterization. I’ll discuss each of these variables in turn, outlining the range of parameters the grammar is able to set. For reasons of space, I direct the reader to Chapters 3, 4, and 5 in Baker (2015) for detailed argumentation and motivation for each of these parameters.

- (23) If a category XP bears c-command relationship R to another category ZP in domain W, then assign case C to XP.

Baker outlines three relationships that the relationship R parameter can take: (i) c-command, (ii) is

c-commanded by, and (iii) negative c-command. The first two are understood in the familiar way. The novel negative c-command relation is introduced to account for case patterns called marked nominative and marked absolutive. In marked nominative languages, the subject of both transitive and intransitive verbs are overtly marked with a nominative affix and the object of a transitive verb is not marked with any case affix at all. An example is shown from Oromo in (24) (Owens, 1985). While similar to nominative-accusative languages, marked nominative languages differ in that the object of a transitive does not bear any case marking. To account for languages that exhibit this type of pattern, Baker proposes a relationship called negative c-command which says that there is no NP₂ such that NP₂ c-commands NP₁; essentially ensuring that the nominal in question is the highest nominal in a case assigning domain. This negative c-command parameter can manifest in the dependent case schema in the following way shown in (25a):

- (24) a. Sárée-**n** adii-n nî iyyi-f-i
 dog.**MNOM** white.MNOM FOC bark-F-IMPF
 ‘the dog is barking.’ UNERGATIVE
- b. D’axáa-**n** maná duubá: b-bu’e
 rock.**MNOM** house behind LOC-fell
 ‘the rock fell behind the house.’ UNACCUSATIVE
- c. Húrrée-**n** arká d’olki-t-i
 fog.**MNOM** sight.ABS prevent-F-IMPF
 ‘fog reduces visibility.’ TRANSITIVE
- (Owens, 1985)

- (25) a. Assign NP₁ marked nominative if there is no other NP, NP₂ in the same domain WP as NP₁ such that NP₂ c-commands NP₁.
- b. Assign NP₁ marked absolutive if there is no other NP, NP₂ in the same domain WP as NP₁ such that NP₂ is c-commanded NP₁.

The algorithm in (25b) is intended to take care of marked absolutive languages, but Baker remarks that we’ve only found evidence that one language exhibits this pattern: Nias (Donohue & Brown, 1999). While the transitive object and the intransitive subject share a case marker in Nias, this

marker doesn't take a typical affixal form, but is rather expressed via a feature change in the initial consonant (26). With respect to the negative c-command relationship more generally, Baker argues in favor of proposing it not just to align marked nominative within the bounds of dependent case, but more importantly because he believes a dependent case account is actually superior for this type of data. Since agreement appears independent of case marking in these languages, he argues that an agreement account would require a mismatch between syntactic markedness and morphological markedness – nominative in marked nominative languages would be syntactically unmarked, but morphologically marked.

- (26) a. Manavuli sui [n-ama-da Tohönavanaetu] ba Maenamölö
 return again [MABS.father.1PLPOSS Tohönavanaetu LOC Maenamölö
 'Ama Tohonavanaetu came back again to Maenamölö.'
- b. I-a [m-bavai] [ama Gumi]
 3SG_{SUBJ}.real-eat ABS.pig father.ERG Gumi
 'Ama Gumi eats pigs.' (Donohue & Brown, 1999)

The next variable to explore is the domain variable W. We can broadly characterize this variable as the set of spell out domains, but "to what degree [the grammar] assign[s] different cases in different domains is another one of its parameters" (Baker, 2015, p.182). These domains include CP-TP, which is the typical clausal domain. We can see these effects quite easily by comparing (27b) and (27a). In (27a) we see that the presence of the matrix subject, which c-commands the embedded subject does not trigger accusative case marking on the embedded subject, plausibly because the two nominals in question are in distinct domains. Conversely, in (27b) the presence of the matrix subject *does* trigger accusative case on the embedded subject because this time the two nominals do occupy the same phase.

- (27) a. Jane hopes that he will win.
 b. Jane expects him to win.

Baker argues that if we assume the smaller vP-VP domain is one that case assignment is sensitive to, we can provide explanations for both difficult differential object marking (DOM) patterns like those

from Sakha in (28) and “special” dependent cases assigned within the VP like dative, oblique and partitive cases. A standard approach to DOM is to suggest that there are two structures available: one where the object remains inside the VP and one where the object moves outside the VP into the ν P. The effect this has for dependent case assignment is that this movement can feed the application of dependent case. If the nominal stays inside its original VP, then it and the subject are in different spell out domains (29) and the dependent case schema will not trigger the assignment of any dependent case. However, if the object moves outside that VP (30), then it will be spelled out along with the contents of the TP when the phase head C is merged into the structure. In this instance, there *will* be two nominals in the same spell out domain and therefore the dependent case mechanism will apply, assigning accusative to the object.

- (28) a. Masha salamaat-y sie-te
 Masha porridge.ACC eat.PAST.3SG_{subj}
 ‘Masha ate the porridge’
 b. Masha salamaat sie-te
 Masha porridge eat.PAST.3SG_{subj}
 ‘Masha ate porridge’(Baker & Vinokurova, 2010)

(29) [TP Subject T [_V Object]].

(30) [TP Subject T Object_i [_V t_i]].

The ability for ν P-VP to act as a domain to account for DOM requires an additional theoretical assumption, one that Baker admits will be quite controversial. The problem is that in accounting for DOM in this way, we introduce the question of how the dependent case mechanism can apply properly in languages that don’t exhibit DOM. The problem is this: if we assume that ν P-VP is a spell out domain that triggers dependent case assignment, we must assume that all objects move out of VP in non-DOM languages in order for them to be able to exist in the same spell out domain as the subject to receive accusative case. Baker instead suggests that the solution is yet another parameter: languages can select whether their ν P is a soft phase or a hard phase. Soft phases are those for which the grammar can continue to see into the ν P-VP spell out domain after spell

out and hard phases are those for which the contents of the ν P-VP are invisible once spelled out. Baker explains that while superficially similar to Chomsky's weak/strong phase heads (Chomsky, 2000, 2001), it differs in how the choice is made. Chomsky tried to make the distinction universal and derivable from head type; passives and unaccusatives would require weak phase heads, for example while other verb types would require strong. The distinction between Baker's soft/hard phases is instead a parameter that a language can just choose to set one way or the other; hard phase languages would be predicted to allow DOM, while soft phase languages would not.

The other benefit of the grammar being able to take advantage of the smaller ν P-VP domain is that we can explore the assignment of dative, oblique, and partitive cases as an additional kind of dependent case. Baker proposes the rules in (31) to account for these "special" cases and pairs them with the rules in (32) to show symmetry between dependent case operating in the CP-TP spell out domain and dependent case operating in the ν P-VP spell out domain.

- (31) a. If XP c-commands ZP in VP, then assign Case U (dative) to XP
- b. If XP is c-commanded by ZP in VP, then assign Case V (oblique) to XP.
- c. Elsewhere NP in VP is assigned case W (partitive).

- (32) a. If XP c-commands ZP in TP, then assign Case X (ergative) to XP.
- b. If XP is c-commanded by ZP in TP, then assign Case Y (accusative) to XP.
- c. Elsewhere NP in TP is assigned Case Z (nominative/absolutive)

Baker does propose one additional phase head that can condition dependent case assignment: the aspect head. Like ν P, the ability of an aspect head to serve as a phase is a parameter which languages can set. The motivation for this part of the proposal is the potential to account for split ergativity data where case can alternate and is conditioned in part by aspect. The contrast between examples (33) and (34) from Coast Tsimshian shows that the choice of tense-aspect marker affects case alignment patterns (Dunn, 1995).

- (33) a. Yágwa húumsg-a geen.
 PRES sniff.ABS skunk
 ‘the skunk is sniffing around.’
- b. Yagwa-t t’uus-da ’yuua-(a) hana’k
 PRES.3SG_{ERG} push.ERG man.ABS woman
 ‘the man is pushing the woman.’ (Dunn, 1995)
- (34) a. Nah siipg-a hana’a
 PAST be-sick.ABS woman
 ‘the woman was sick.’
- b. Nah t’uus-a ’yuuta-(a) hana’k
 PAST push.ABS man.(ABS) woman
 ‘the man pushed the woman.’ (Dunn, 1995)

The final domain capable of conditioning dependent case assignment is the DP-NP spell out domain, a proposal present in Marantz’s (1991) version. The main function this domain serves is to allow for the assignment of genitive case as a type of unmarked case, distinct from nominative, that is assigned in the DP-NP domain rather than the CP-TP domain.

At last we come to the final variable – the categories XP and ZP that can participate in the assignment of dependent case. It is arguably within this variable that we see the greatest degree of parameterization. Most broadly, the things that can participate in dependent case assignment are syntactic objects that contain “referential indices” (Baker, 2015, p.183). The two variables XP and ZP can be of the same category type, the difference in labels X and Z is intended to differentiate the case receiver from any case competitors. Baker suggests that what counts as a case competitor ZP is something that is parameterized along a scale that is dependent on what sorts of features each nominal-like thing has. This scale is shown in (35). The idea is that each category type on that scale has a different degree of nominal features. The ones with a full set will always be case competitors, then ones with none will never be and the rest are ranked along a scale such that languages can choose where the cut off point is for them. This scale captures an incredible degree of interesting data, the details of which are too large to outline here (see Baker (2015) chapter 5 for a full walkthrough). It is important to note that he does admit that the details of exactly what features are relevant for each different category along the scale are still to be worked out.

	overt NP & clitics	full nominal features
	null referential pronouns (pro)	
(35)	controlled PRO	
	uncontrolled PRO	
	weak implicit arguments	
	PPs, VP, etc	no nominal features

With the dependent case schema in place and an understanding of the range of values that each parameter can take, I quickly review Baker's assumptions regarding the timing of these different modes of case assignment. Lexical/quirky/inherent case is assumed to be assigned first, via the immediate relationship formed by merging into a predicate projection that assigns quirky case. From there, Baker argues that the dependent case mechanism assigns dependent case according to the schema outline in (36) with the relevant parameters set given for the particular language and/or case. After this mechanism applies, the grammar is then able to assign agreement-based case to anything that fits the relevant structural description. Finally, the grammar will then assign unmarked case to any nominals that, for whatever reason, were unable to receive case via one of the earlier methods. It is also worth mentioning here that like most modern applications of dependent case (Levin, 2015; Levin & Preminger, 2015), Baker assumes that this kind of case assignment happens in the narrow syntax, at or before spell out, not in a post-syntactic component as assumed by Marantz (1991), McFadden (2004), and Bobaljik (2008).

- (36) If a category XP bears c-command relationship R to another category ZP in domain W, then assign case C to XP.

Before moving on to an exploration of the theoretical implications of this model, it is worth mentioning one interesting benefit of adopting case assigned via the dependent case model. Let's fill in the schema in (36) with the following values to produce (37a) and (37b). Baker explains that if we allow these two dependent case rules to apply independently, then 4 outcomes are logically possible: (i) (37a) only applies, producing nominative/accusative languages, (ii) (37b)

only applies, producing ergative/absolutive languages, (iii) both (37a) and (37b) apply, producing tri-partite languages,⁹ and (iv) neither (37a) nor (37b) apply, producing languages like Bantu which seem to make no use of case morphology at all (Diercks, 2012).

- (37) a. If an NP₁ is c-commanded by another NP₂ in domain TP, then assign accusative to NP₁.
 b. If an NP₁ c-commands another NP₂ in domain TP, then assign ergative to NP₁.

This review, while lengthy, of course does not capture the breadth of the entire proposal, but my hope is that it provides enough information to discuss some of its broader theoretical implications. The next section will focus on exploring what those are with the intention of arriving at the conclusion that there are some troubling results we are forced into accepting by adopting such a model, despite how well it captures this wide range of data.

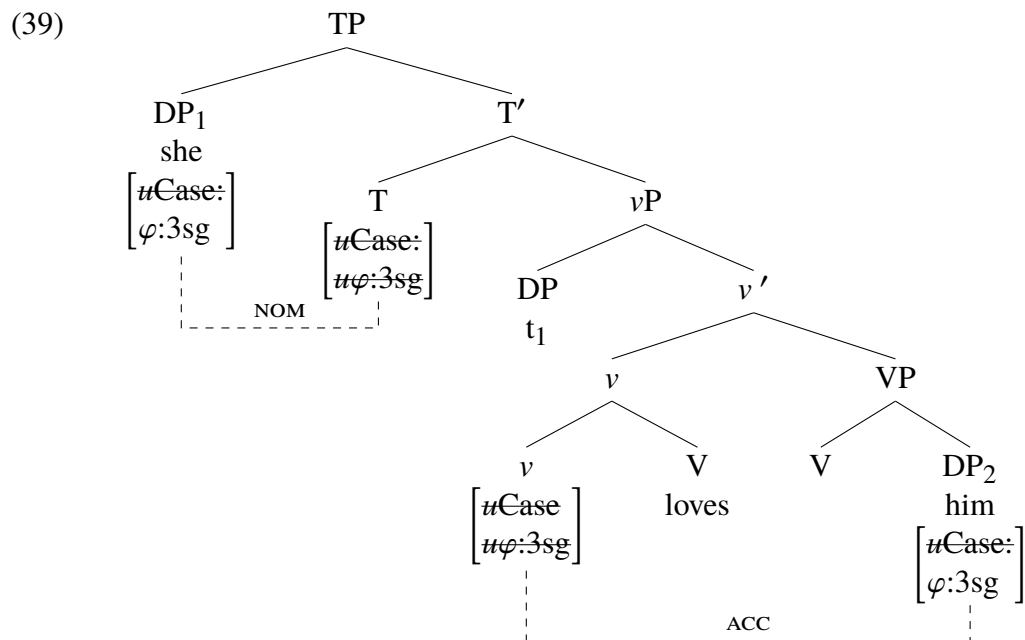
2.2.1.3 Interim Walkthrough

Before discussing the implications of adopting the dependent case approach, I want to provide a quick summary of how the two basic models of case assignment would account for some familiar case patterns so that we may enter into that conversation having seen the two systems comparatively illustrated. In a simple transitive clause like (38) (whose derivation is shown in (39)), *v* is specified with uninterpretable φ -features and an uninterpretable case feature. The uninterpretable φ -feature probes, looking for a valued instance with which to agree. It finds a viable goal in the DP object *him* which has valued third person singular features and an uninterpretable case feature of its own; the probing of φ -features is represented by the dashed lines drawn. By reflex of AGREE, the interpretable third person singular features on the object will value the uninterpretable φ -features on *v*. As a result, *v* will assign accusative case to the DP object. A similar relationship is formed between the finite T and the DP subject. Finite T also has uninterpretable φ -features that need a value and they

⁹Tripartite languages, such as Nez Perce, exhibit case morphology patterns that are viewed as having shared properties between nominative/accusative languages and ergative/absolutive languages. (See Deal (2016) for more discussion on these patterns).

find a suitable goal with valued φ -features in the DP subject *she* which has third person singular features. As a result of having agreed with finite T, the φ -features on T are valued and the DP subject receives nominative case.

(38) She loves him.



Dependent case theory would assign case in this example differently. First, we'd assume that the following parameters are set for English such that accusative case is calculated according to the following schema in (40). Note that since we're following Baker's approach, we'd also need to assume that the v phase head is a soft phase head, thus allowing the contents of the VP spell out domain to be visible to things outside the domain. Along with this algorithm dictating how dependent accusative case is assigned is another assumption that nominative case is the unmarked case that is assigned in the negative environments that this algorithm does not identify.

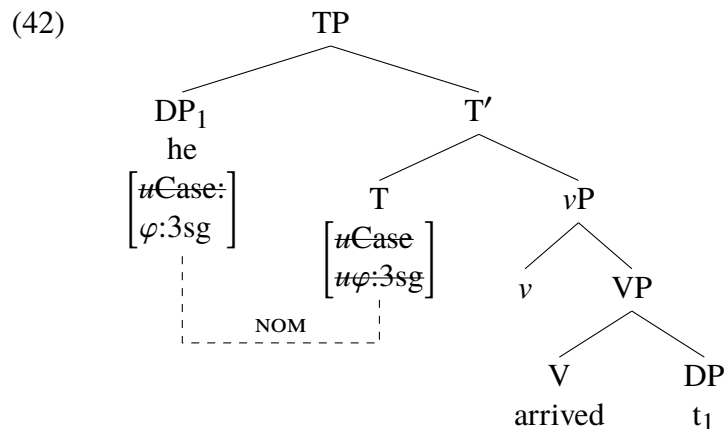
(40) If DP_2 is c-commanded by another DP_1 in the spell out domain TP, assign accusative case to DP_2 , provided DP_1 does not already have case.

This algorithm would assign dependent accusative case to the object because the object constitutes

a DP that is c-commanded by another DP that shares the spell out domain TP. After accusative case is assigned by this mechanism, the grammar then investigates the subject nominal, sees that it is caseless and as a result of being in the domain defined by the unmarked case (TP), assigns it unmarked nominative case.

For a simple unaccusative clause like that in (41) and (42) under the agreement-based approach, finite T's uninterpretable φ -features would probe – again represented by the dashed line – searching for a valued instance of φ -features, finding success with the theme argument. An EPP feature triggers the argument's movement to the specifier of TP position, the uninterpretable φ -features on finite T are valued by the φ -features on the theme argument, and the theme's uninterpretable case feature is valued nominative as a result. Dependent case would instead use the same algorithm shown in the example above to examine the spell out domain TP, see that there is no DP for which it is c-commanded by another DP, and therefore fails to assign accusative case. The grammar would then assign unmarked case to any nominals which did not receive accusative case, valuing the caseless subject with nominative case.

(41) He arrived.

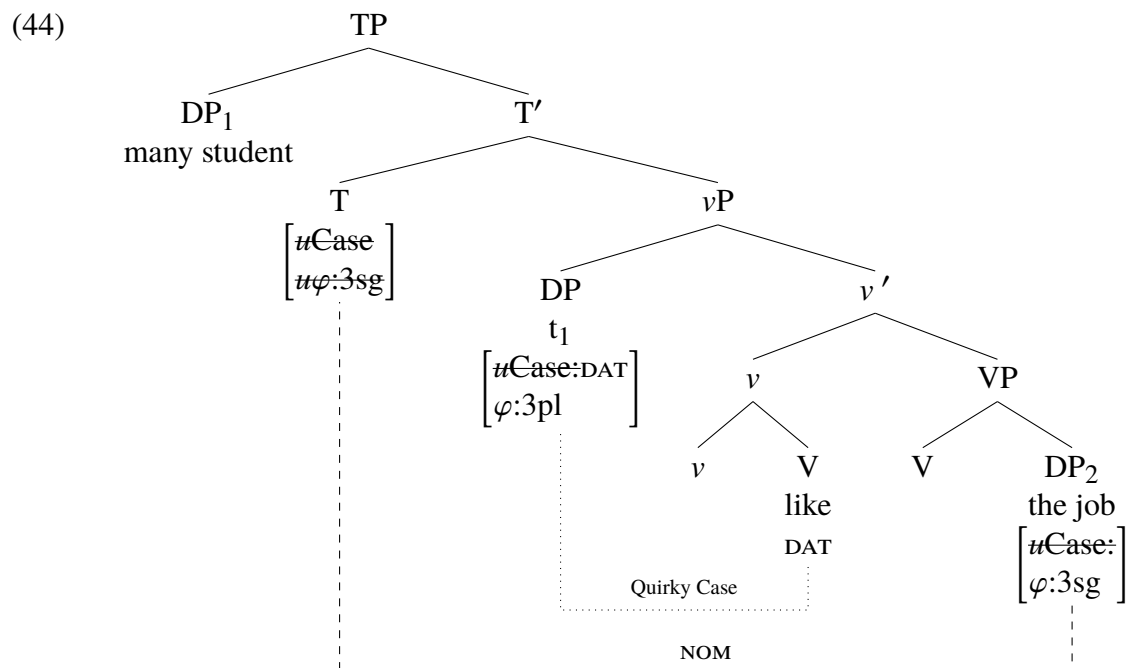


Both agreement-based case mechanisms and dependent case ones would handle Icelandic transitive data as they did with English transitives like the example shown in (39), but an Icelandic quirky case example would be assigned case differently. Here, the subject is assigned a quirky dative

case and the object receives nominative (43). Both agreement-based case models and dependent case ones assume that quirky dative is assigned by the verbal projection, but the two models differ in how they assume the object receives nominative. Agreement-based case models argue that finite T agrees with the object, assigning it nominative as a result (44). The idea is that the subject is able to be bypassed due to its already having received case – it is not visible to the φ -agreement probe. This is supported by data like that in (43b) which shows that when the subject is assigned quirky case, it is the object that controls number agreement. Dependent case would operate in the same way it has for the previous two examples: once TP is spelled out, the dependent case algorithm is not able to assign accusative case since there is no DP for which there is another caseless c-commanding DP, so the unmarked nominative is assigned to the object instead.

- (43) a. Morgum studentum liki/*lika verkið
 many student.PL.DAT like.3SG/*3PL job.the.NOM
 ‘many students like the job’
- b. Henni leiddust þeir
 she.DAT was-bored-by.3PL they.NOM
 ‘She was bored with them.’

(Harley, 1995)



2.2.2 Implications

Most of those who adopt dependent case have directed their focus toward empirical coverage, Baker in particular. While we've learned much from the impressive empirical coverage this model has achieved, there are many conceptual and theoretical implications that still need exploration and subsequent evaluation. The rest of this chapter is an attempt to that aim. I will argue here that this evaluation should lead us to be more skeptical of the tenability of a dependent case approach and should motivate us to make other modifications to case theory that align it more closely with the standard approach.

Each of the works that employ dependent case as either the singular or primary case assignment mechanism argue well against the standard agreement-based case approach proposed by Chomsky (2000, 2001). It is clear that this model, as proposed there, will not be able to account for the widely varied morphological case patterns; some modifications will surely need to be made. Default case data provides an especially clear argument. I will offer a proposal of what I suggest those modifications should be in chapter 4. First, I think it prudent to provide motivation for why we should do so, when there is a model of case valuation available (dependent case) that appears to fill in these empirical gaps quite well. The arguments laid out here are therefore relatively modest ones in the face of the impressive empirical coverage achieved by Baker (2015); the impact of which cannot be overstated. However, it is also incredibly important for us as theoreticians to understand what concessions we make about theoretical concerns through its adoption. I will argue here, and in chapter 3, that adopting these systems requires a very rich and detailed UG, one that we should be cautious of if we entertain seriously the biolinguistic perspective that calls for a minimally lean UG (Chomsky, 2005).

There are two dimensions along which we need to frame our arguments: those against dependent case more generally and those against the type of hybrid case approach proposed by Baker. Now that we have worked out enough details to understand how dependent case needs to work under Minimalist assumptions, it is time to investigate the model's conceptual and theoretical implications and evaluate whether the data captured is of enough benefit to concede any theoretical

unattractiveness. There are three central topics to discuss: (i) the theoretical impact of abandoning the GB central notion of government, (ii) the high degree of parameterization, and (iii) the nature of dependency establishment. The following sections will outline the theoretical concerns of each.

2.2.2.1 Abandonment of Government

In this section I discuss a few conceptual issues that have not yet been thoroughly addressed that arise from the replacement of government as the domain-defining relationship in favor of phase heads and spell out domains. Because the government relation was the defining relationship that dictated the assignment of case features in the original 1991 version, it is not surprising that its abandonment has great effect.

When comparing modern dependent case theory to its contemporary agreement-based alternative or to case theory as it existed under the GB framework (Chomsky, 1981), it is easy to come away with the impression that they are radically different approaches that share very little in common. Although dependent case was indeed a radical new proposal, it didn't completely upend the system in ways that required us to reconfigure the framework. It employed the same relation – government – to assign case to nominals, an assumption also held in the standard GB case framework. What distinguished Marantz (1991) from its contemporaries was that Marantz assumed the case-assigning heads used a different kind of information – configurational information – to make the decisions about which nominals received which specific case features. While the information used to calculate case was different, the central notion of government and the role functional heads played remained primarily the same.

As we've discussed, with the abandonment of government came the requirement to redefine the conditions under which dependent case applies. Recent updates to these conditions replace the government domain specification with one that defines the domain with reference to phases: case is assigned configurationally to nominals that occur within the same phase. This shift is successful both empirically and theoretically in that it both captures the relevant case patterns and uses the quintessential Minimalist domain that we assume to be at the center of derivation construction. It

thus constitutes an intuitive and reasonable way to update what it is that defines case assigning domains. While it is easy to treat this update as a trivial one, made solely to bring dependent case in-line with modern theoretical assumptions, I argue that this update introduces three non-trivial difficulties: (i) it removes the source of case features in a way that violates the central Minimalist assumption of Inclusiveness, (ii) it removes the ability to make the empirical distinction between unmarked and default cases, and (iii) it results in an inconsistent syntactic conceptualization of case that is both problematic for acquisition and theoretically unattractive, especially if one follows Baker (2015); Levin (2015); Levin and Preminger (2015); Preminger (2014) in assuming case is assigned in the syntax.

If the dependent case model applies in the narrow syntax, it is not a trivial detail to figure out how exactly the assignment of case feature values works. So while it is intuitive to say that “case features are assigned”, it isn’t a trivial question to ask “by what”, especially when working in a feature-driven model of the grammar. At minimum, standard frameworks with Minimalist assumptions require that, like all syntactic objects, features must come from somewhere (the numeration) and they must originate on something as they are defined as properties of syntactic objects. The Inclusiveness Condition, one of the most central Minimalist constraints, further refines these requirements (Chomsky, 1995, 2000). The idea is that new syntactic features cannot be added throughout the course of the derivation. They must be generated on some lexical item and cannot enter after this initial selection into the array is made. This achieves two aims: first, it greatly constrains the power of the generative ability of the grammar. If features could be inserted at any stage, we would have to propose a number of additional principles that would further constrain the range of possibilities that unfettered insertion would produce. Assuming the derivation has everything it needs at the beginning is the most minimal assumption in that sense. Furthermore, we can make a connection between this Inclusiveness Condition and the Chomsky-Borer Conjecture (Borer, 1984; Chomsky, 2001). This conjecture is not only widely adopted, but it is also likely true and offers the most principled understanding of how humans could actually acquire these rules and constraints. The strong version of this idea is that all syntactic variation must be visible at

the level of the lexical items themselves, since they are the only acquirable/tangible piece to which the language learner has access. What this means for the acquisition of case features and any mechanisms available for case assignment is that we have to clearly understand which lexical items house the individual case features and how those features are assigned from one syntactic object to another.

Without the relation of government conditioning which nominals are in the case assigning domain, the actual locus and assignment of case features is not defined. I see three imaginable options: (i) case features are located on V+T or some other relevant case-assigning functional head, (ii) case features are located on the nominals themselves, or (iii) case features are supplied “by the grammar” or some dependent case mechanism. Let’s first consider option (i), that the case features originate on functional heads like V+T, as they do in early versions of this model. This option is first problematic in that no modern dependent case proponent could reasonably adopt it without seriously undermining the defining principle of the model. The hallmark advantage of modern dependent case is its ability to remove any dependence on the presence of some case-assigning functional head. Not only is this the primary motivator, it is also the model’s defining characteristic. By locating the derivational origin of case features on V+T as the original version does, we once again revert to a dependence on the presence of those functional heads for case-assignment, negating the primary benefit over the agreement model. It’s reasonable to ask about the original version, which did locate those features on functional heads; this is where the abandonment of the relation government becomes relevant. Say modern versions of dependent case mimicked Marantz (1991) in locating the source of the case features on V+T, putting aside the reluctance to do so because of a hesitance to rely on functional heads. Without access to the government relation, we lose the connection between the origin of the case features and their eventual derivational destination. We thereby create a system where case is conceptualized as the reflection of a relationship between two syntactic objects (the nominals), but the features that signal this relationship come from an uninvolved third-party, obfuscating the very relationship case is supposed to reflect. The third party status of those functional heads is what is at issue here. The GB-era version was able to avoid this

conceptual issue because through reference to government, the third party V+T complex wasn't an *uninvolved* one. Case-assignment in that early version, although calculated using information about the existence of other nominals, still operated under the assumption that nominals were only under consideration for case assignment if they were governed by the functional head that was the source of those features. Case was simultaneously the reflection of a relationship between a nominal and a governing functional head and a nominal and its government domain-mates. Without the notion of government and with the decision to dissociate case assignment from agreement, it seems untenable to locate case features on any functional head like V+T without introducing some serious conceptual issues.

Option (ii) appears similarly untenable. Case is calculated on information about the relationship between two nominals, so it's not unreasonable to wonder if the origin of case features is the nominals themselves. It should be fairly obvious why this is a nonstarter, but for thoroughness, let's briefly examine why. If we pursued this option, nominals would be generated with both a valued and an unvalued case feature. It is easy to see how the unvalued instance of the case feature would depend on configurational information, but it's important to notice that the valued instance of the feature would also require this information, making it difficult for each nominal to 'know' what feature to be generated with. In other words, not only do nominals depend on configurations to *receive* a feature value, they also would depend on those configurations to *assign* one. Since this configurational information is inaccessible at the point of generation, it is untenable to assume nominals are transferring feature values to one another.

Finally, we are left with the option that modern dependent case proponents actually adopt: that the dependent case mechanism itself is what assigns case features to nominals. This option is quite attractive upon first look because it mimics other sorts of pre-Minimalist grammatical rules. It follows an operation type logic where in the context of a particular structural description, the grammar performs a feature-assigning operation. What is more problematic upon a closer look is that, while familiar, this sort of process constitutes a clear violation of the Inclusiveness Condition – the assumption that new syntactic features may not be introduced into the derivation after their

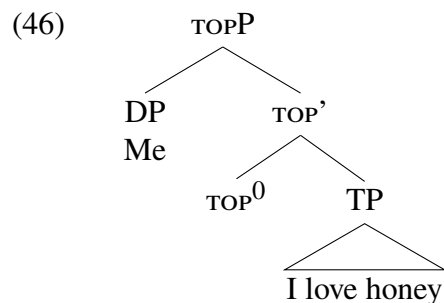
initial selection from the lexicon into the specific lexical array for the derivation under discussion. As properties of lexical items, case features by definition must originate on some lexical item. If they don't their addition later in the derivation, regardless of the mechanism, introduces exactly the type of dangerous theoretical power Inclusiveness tries to constrain. Violating Inclusiveness is not a trivial matter as it greatly restricts the power of the derivation by disallowing unnecessary diacritics, traces, or other convenient theoretical tools that reduce explanatory value and forces the derivation to adhere as strongly as possible to interface constraints. It is also important to note that the need to adhere to this condition goes beyond dogmatic obeisance to Minimalism as a program. Violations of the Inclusiveness Condition constitute real issues for understanding how it is that individual humans acquire the syntactic system as they minimize the tangible linguistic evidence that a language learner can observe. They also raise serious issues for understanding how groups of humans as a species acquired or evolved the capacity for language as they greatly expand the set of things that must be part of UG. It's possible for violations of the Inclusiveness Condition to be tolerated if we could consider those violations perfect solutions to interface constraints; it is not obvious however that dependent case mechanism could be framed in this way. Since under a dependent case model, neither of the three available explanations for the origins of case features appears tenable, the abandonment of government raises some serious concerns about the tenability of the approach itself, at least as an mechanism that is active in the syntax.

The abandonment of government creates another problem: it removes the ability to draw an important distinction between unmarked case and default case. The inability to draw the theoretical distinction between the two doesn't cause any empirical issues for most languages, whose unmarked case happens to be synonymous with its default case. There are however, a number of languages like English, Dutch, and Norwegian whose unmarked case is nominative, but whose default case is accusative. Therefore an inability to make a distinction between the two raises an empirical issue for this subset of languages.

Since government was the domain defining condition under Marantz's original proposal, it was possible to draw a distinction between unmarked case (case assigned by the governing head when it

didn't assign the dependent case) and default case (case assigned by the grammar when a nominal wasn't governed by a case assigning head). With the replacement of this relation with the concept of phase domains, we are no longer able to maintain this simple distinction because it is impossible for a nominal to not be in some spell out domain. One might get around this worry by suggesting that the label of the phase head *X* is what allows us to distinguish between the unmarked and the default cases. This could work quite nicely for examples where the default nominal is outside a domain like TP, presumably in some sort of focus or topic phrase. We could argue that the unmarked case is restricted to the domain defined explicitly by TP spell out. In this way, the unmarked nominative case nominal *I* that is in the TP domain could be distinguished from the default accusative case nominal *Me* that is not in the TP domain and is in TOPP domain instead.

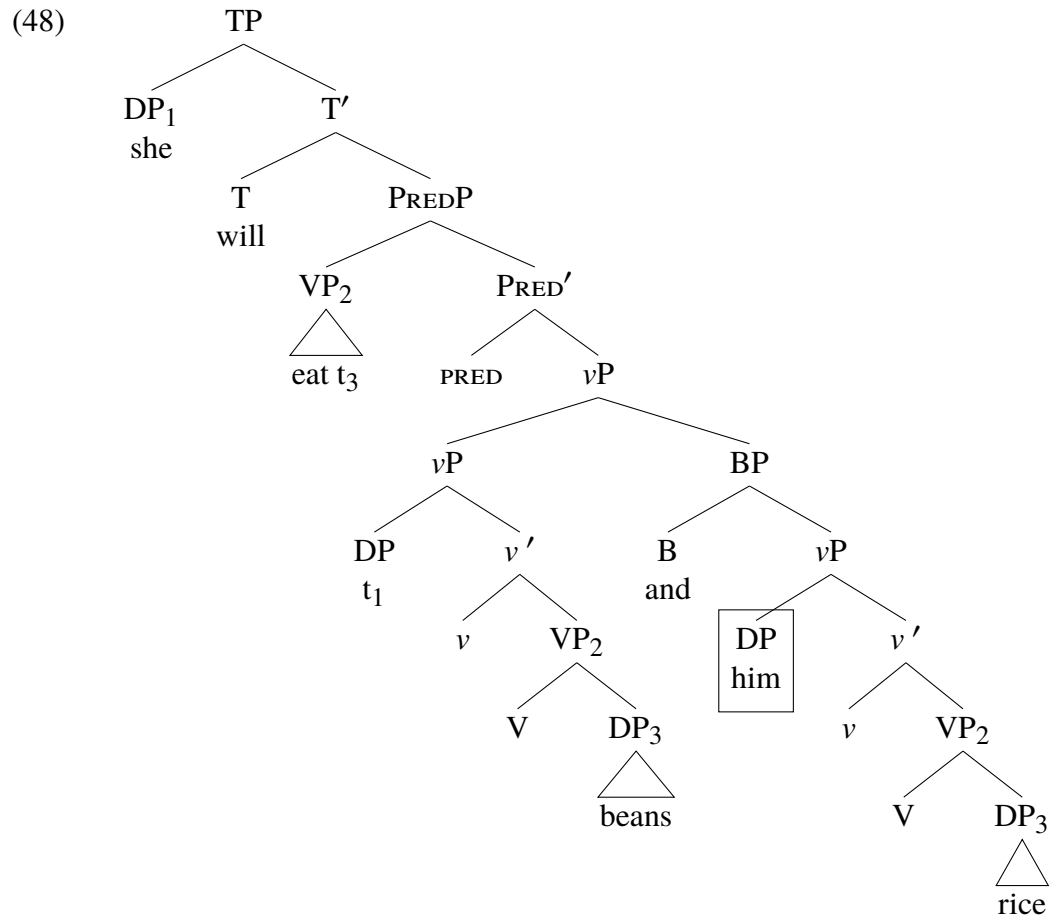
(45) Me, I love honey.



Where this sort of solution becomes problematic is with examples where default case nominals do appear within the main clausal structure. This makes it quite difficult to argue that we could make a distinction using phase domain labels. Take the gapping example shown in (47). For this discussion, I will assume the proposal suggested in Johnson (2009), shown in (48). There are three parts to his approach: (i) low coordination of the *v*Ps (ii) heavy NP shift of their objects, and (ii) across the board movement of the verb phrases. Johnson assumes that when two *v*Ps are coordinated, that coordination can trigger two separate processes: the rightward shift of the objects outside of their respective VPs and the subsequent across the board movement of those VPs to the specifier of a predicate phrase. The subject of the first *v*P, as the highest DP in the structure, will

be the one targeted by the EPP feature of T and will raise to subject position. The subject of the second vP will remain in its original position. It is this position where it receives default case as there isn't a canonical accusative case assigner available to assign its case features. In order to distinguish unmarked from default case in the clausal domain, we would need to define a set of domains whereby each of the unmarked cases was assigned. In modern dependent case approaches, these domains are TP for unmarked nominative case and DP for unmarked genitive case. Only outside of these unmarked domains would the default case be allowed to surface. As we can see in the structure below, the default nominal *him* exists within a TP domain and as such, constitutes a clear difficulty in distinguishing it from the nominative unmarked case nominals.

(47) She will eat beans and him rice.



Another way to address the distinction problem would be to instead propose that we reverse the way the dependent case mechanism operates in English and these other languages such that the default accusative case is aligned with an accusative unmarked case instead of an unmarked nominative case. This would frame nominative case as the assigned dependent case instead. To be clear, this type of solution is entirely within the confines of dependent case theory and so no issues can be raised there. Instead of adopting the canonical dependent case algorithm for accusative case (49a), we could change the values of the parameters to reflect this reversal (49b).

- (49)
- a. If DP_2 c-commands another DP_1 in the spell out domain TP, assign accusative case to DP_1 , provided DP_2 does not already have case.
 - b. If DP_2 c-commands another DP_1 in the spell out domain TP, assign nominative case to DP_2 , provided DP_1 does not already have case.

Doing so would successfully align the unmarked and the default cases while obeying the rules of dependent case model, but would require a theoretical departure in how we understand how case features and categories relate to one another. While it is unproblematic to assume case feature inventories vary cross-linguistically, it is far more problematic to assume that the inherent relationships between these features differ, which is what this reversal would require. For languages where nominative is both the unmarked and default case, we would need to assume that the features responsible for nominative case are less specified than those for accusative case. Adopting the proposal that features have inherent hierarchical structure (Harley & Ritter, 2002) means that whatever features responsible for nominative would dominate those responsible for accusative. For languages where accusative is both the unmarked and default case, we would be forced to assume the opposite. Reconciling how both are true would require that we assume case features do not have any inherent feature structure, singling them apart from how we assume all other syntactic features are organized. This would seriously undermine one of the hallmark benefits of adopting dependent case: its ability to uniformly account for varied cross-linguistics morphological patterns using the same mechanism. Once again, the abandonment of government as a domain-defining relation has

made the modern extension of dependent case more problematic than it first appeared.

As an aside that we'll discuss later in chapter 4, it is also worth exploring what the distinction between unmarked and default case means for the relationship between the individual features that make up the various case categories. This is an arena where we haven't connected up the insights made by those whose primary focus is on morphological case and those whose focus is on syntactic case. Decomposing case categories into individual case features has its roots in the literature on the various case syncretic patterns observed cross-linguistically (McFadden, 2007; Müller, 2004a, 2004b, 2005). The main idea is that if two cases are syncretic in a language, they must share some set of case features with other, while maintaining enough distinction in case features with the other cases that are not syncretic. This dissociation is incredibly standard in the morphological case literature, despite the lack of a general consensus on exactly what those features are. Conversely in the syntactic literature, it is uncommon to discuss case features as singular units that comprise case categories, despite an implicit acknowledgment that this is of course true (Pesetsky, 2013). Instead, we talk about assigning nominative case as a whole, failing to connect up the individual parts that make this happen. Under a dependent case approach, we need to better understand how those individual features that are responsible for nominative case or accusative case are understood to operate. McFadden (2007) addresses this point explicitly and we'll return to this discussion in chapter 4 where I introduce an alternative proposal that addresses these issues.

Finally, modern versions of dependent case, without access to this government relation, result in a great deal of conceptual inconsistency when it comes to understanding what it is in the syntax that case morphologically reflects. Under a more standard agreement-based approach, case can be conceptualized as the reflection of a relationship between a nominal and a functional head that is formed when AGREE establishes a dependency between those two syntactic objects. It's interesting to note that this is also true of the original dependent case proposal. Even though dependent case uses configurational information to calculate which nominal receives which case, nominals in the original proposal did receive case from V+I under government. Case was therefore a reflection of the same functional head/nominal relationship – the V+I complex cannot assign case to nominals

that it does not govern.

Under either a hybrid model or a strict modern dependent case model, the syntactic relationship that case reflects is much less clear; case does not seem to have a consistent conceptualization beyond a way for the grammar to distinguish nominals. When case is assigned lexically, as it is in (50) where the subject is marked with quirky dative, it is understood to be the reflection of a ‘special’ relationship between a nominal and a ‘special’ verbal head. In the dependent case model, this is unexpected because one of the central principles is the avoidance of a reliance on functional heads for case assignment. Case is modeled as the reflection of a relationship formed between two nominals, not between a nominal and functional head. To address concerns about whether this would constitute the kind of dependence on functional heads that proponents of dependent case passionately avoid, Levin and Preminger (2015) argue that the sisterhood relationship formed by merge is local enough to obviate the worry.

- (50) Morgum studentum liki/*lika verkið
 many student.PL.DAT like.3SG/*3PL job.the.NOM
 ‘many students like the job’ (Harley, 1995)

This doesn’t however, address the conceptual inconsistency between lexical and dependent case that exists even in the strict dependent case model. Sisterhood, while perhaps a more palatable reliance on functional heads that dependent case proponents might tolerate, is still a relationship between a nominal and a functional head. The issue here isn’t about how the relationship becomes established and whether or not that constitutes a reasonable exception, but rather that the sisterhood exception exists in the first place. Under a dependent case model, quirky case is quirky because it reflects an unexpected case relationship, one between functional heads and nominals that explicitly doesn’t exist elsewhere in the system. Under a more standard approach, quirky case reflects an expected relationship – since all case reflects a relationship between nominals and functional heads – but with an unexpected functional head. It appears that the existence of quirky case is more unexpected under the dependent case model than it is under an agreement-based model.

With respect to accusative or ergative case assigned via the dependent case algorithm, the

relationship case reflects is even more unclear, given our discussion above about where to locate case features. If the source is the functional complex V+T, then we could maintain consistency at least between lexical and dependent case, but with the existence of the issues raised above and given that modern researchers are reluctant to depend on the presence of a functional head, this doesn't seem likely to be the case. If the source is either the nominals themselves or somewhere undefined in the grammar via an operation, we can reasonably conclude that case is the reflection of a relationship between two nominals. This signals a system where sometimes case is the reflection of a relationship between a functional head and a nominal (for lexical case and/or dependent case, dependent on the feature source) and sometimes the reflection of a relationship between two nominals. Additionally, unmarked case is the reflection of no relationship at all. So even within a strict dependent case model where there is no agreement-based case, case is not a conceptually consistent entity. This status is even more true of a hybrid model.

At this stage, it's reasonable for one to ask: why would this conceptual inconsistency be a problem? Perhaps all case is is the reflection of the need for the grammar to distinguish nominals in some arbitrary way. One could propose a function-based explanation where case distinctions help aid communication in some meaningful way. If true, then case doesn't need to consistently reflect the same sort of syntactic relationship; it simply needs to reinforce the fact that the nominals in question are syntactically distinct from one another. Examples shown in (51a) and (51b) however illustrate that if case differentiates nominals to aid in communication, the grammar doesn't appear to reinforce those functional distinctions in a consistent way, undermining that there is a system of case at all. Furthermore, this type of conceptual inconsistency makes the acquisition of case incredibly difficult to understand because it undermines that there's a system to acquire in the first place.

- (51) a. She expected **him** to hug **them**.
 b. **She** hoped **he** would hug them.

The removal of government as the domain-defining relationship therefore introduces three main

issues: (i) it creates a system where case assignment constitutes a violation of a crucial Inclusiveness Condition, (ii) it creates a system where we cannot maintain an empirically needed distinction between unmarked and default case, and (iii) its conceptual inconsistency undermines the existence of a case system itself. While these issues may turn out to have solutions, there is benefit to being explicit about what theoretical concessions we must adopt by adopting the dependent case system.

2.2.2.2 Parameterization

Modern dependent case captures a wide range of data through a high degree of parameterization. The implications that we must adopt by pursuing this approach will be the focus of this next section. I certainly don't want the reader to infer that the high degree of parameterization is a deficiency of the model on its own. It is a clear fact that the high degree of cross-linguistic variation in case patterns is daunting and Baker's model, with its high degree of parameterization, allows us to account for a widely disparate number of patterns, while constraining what those options are. This is empirically attractive and gives us lots of insight into how case patterns can be accurately modeled and predicted. What follows is an exploration of some of the questions that such a high degree of variability raises.

Parameters on their own are of course not problematic. It's not controversial to assume something like a head parameter, for example, where merge can choose on which "side" to locate a head. Notice that with a parameter of this type, the parametrical choice is more or less internal to how it works. By this I intend to mean that the head parameter does not *have* to exist as an external rule guiding how merge can apply; it follows from the logical set of possibilities available. Not all hypothetical parameters necessarily share this property, however, and we must therefore be careful to consider the types of parameters we allow and how much power we grant them. This is especially hard to do in the face of such empirically varied phenomena like case marking, as the high degree of variation alone invites the proposal of a highly varied set of external parameters.

The types of parameters proposed in modern versions of dependent case are of a type that should be at least a little concerning if we intend to pursue a Minimalist-aligned theory because they, unlike

a head parameter, don't appear to follow naturally from how the system works in quite the same way. The most obvious one of these would be the high-level parameter of which case assigning mechanism is chosen. While it's certainly possible that languages can decide to assign nominative case either via agreement or via the unmarked case part of the dependent case mechanism, it's not in any way obvious why these are the two choices available. Parameters modeled like this are thus conceptualized in a way that makes them an external sort of parameter – that operates over a particular operation – rather than a parameter that is derived from an independent, logical set of possibilities.

Additionally, in a theory with either a large number of parameters (or rules, as we'll see in the next chapter), we also have to understand how those parameters are organized in a way that makes their acquisition as a consistent set plausible. What the language learner ends up acquiring is an entire set of parameters that make up the system as a whole, despite the fact that they exist as independent 'rules'. In order for language learners to consistently acquire all of the parameters in the correct way, and not for example, miss one or set one in a way that it conflicts with a previously set parameter, there must be some sort of relationship between them that guides this acquisition. Furthermore, a high number of external parameters also causes some problems in understanding solutions to what's termed Darwin's problem, the question of how language arose so quickly in humans (see Hornstein (2018) for a review). The solution that guides the Minimalist program is that what cannot be explained by either the environment or non-language specific cognitive systems must be innate to humans and in order to understand how these innate features evolved so quickly in humans, we must assume they are quite minimal in number. Systems that propose a long list of external parameters or other rules deeply violate this assumption by making UG quite rich.

There are also implications that follow from how we want to understand the particular features that are involved in this case system. Since Baker allows assignment-based case and configurational case to co-exist not only in UG more broadly, but within in a single language, we have to ask what this variation means for the status of case categories in the grammar. It's fairly standard to conceptualize case categories as a middle-man type of label that we give to groups of features that appear to behave

the same way (Pesetsky, 2013). Once we couple this sort of variation with a decomposition of case features, it becomes incredibly difficult to see how this variation would actually be implemented. There are two problems here: one is that by doing so, we elevate the status of case category to a type of syntactic object that the grammar is aware of, which conflicts with more standard understandings of what case categories are, conceptually. The second is that when coupled with a decomposition of case features, it's not entirely clear how the grammar would be able to implement this sort of variation. Say, for example a language assigned nominative case via agreement, but assigned accusative case via the dependent case mechanism. Also assume that nominative case is made up of a set of individual case features, some of which are shared by the set of individual case features that make up accusative case and others that are not shared. For exposition, let's assume nominative is comprised of case features A and B, while accusative is comprised of case features B and C. The shared feature B is what allows the two cases to be syncretic and those that aren't shared, A and C, are what allows us to maintain a distinction. When we try to plug this decomposed case feature system into the mechanics of how case assignment is intended to work, we run into an important question: which of the independent case features that make up accusative case is the dependent case mechanism actually assigning? The most straightforward answer might be features B and C, the entire feature set that makes up accusative case. Likewise, it would follow that the agreement operation responsible for nominative case would be capable of assigning features A and B. However, if true, this means that the language in question has two independent mechanisms of assigning the same case feature B. If instead, the dependent case mechanism assigns only feature C, then we of course have to wonder how feature B appears on the nominal. We'll discuss this point in more detail without a toy sort of example in chapter 4, but for now the point is simply that the treatment of case categories in the modern version of dependent case becomes problematic when one tries to reconcile it with conclusions about feature composition that come from morphological research on syncretism.

It is also worth asking what it means that some of the parameter settings appear to be forced into a particular way. I have two of these in mind. The first is the assignment of ergative case. Baker

argues that ergative case is never assigned via agreement, only via the dependent case mechanism (he also rejects the idea that it's a lexical case). Not only does this require that the grammar is sensitive to the case categories themselves, but it also raises an important question: why for this case is the parameterization forced in one direction? In order to answer this question, we'd likely need to understand what is present in ergative data that forces the language learner to consistently set the ergative case assignment mechanism parameter. Furthermore, whatever this observable-to-the-learner data ends up being must be distinguished from the other cases for which this parameter is not fixed. The other fixed parameters setting comes from a potential solution to the unmarked vs default problem: perhaps for languages where the default is morphologically distinct from the unmarked, the parameter for unmarked nominative case assignment *must* be set to the agreement setting. More generally, these questions become: what does it mean for a system to exist with a set of parameters, only to have some range of them inaccessible for particular cases?

Many of these are largely conceptual issues, but I argue that they are ones that are important to explicitly consider when entertaining adopting such a radical proposal. Despite the impressive empirical coverage that dependent case admittedly offers, it's important to understand what theoretical concessions we're making in its adoption.

2.2.2.3 Dependency Establishment

Finally, we come to issues regarding the status of case assignment under the umbrella of syntactic dependencies. Like φ -agreement, case assignment can be viewed as a syntactic dependency in that it involves one form being dependent on the characteristics of another syntactic object and this dependency is based on structural relations. Relevant to this domain are the set of operations that we assume to be capable of establishing various dependencies in the grammar. Because discussions of dependent case are often (and reasonably I might note) restricted to the domain of case, it is often treated as an *alternative* model of case valuation. Within the boundaries of the case literature it absolutely is and all the work cited in this section has pitted agreement-based models against dependent case ones and let the data battle out the strengths and weaknesses. Through this

exercise though, it's easy to think that these two models are more or less equal when it comes to framework complexity; they differ only in their predictive empirical coverage and/or theoretical implications. However, when we expand our purview to a larger domain of phenomena, namely the establishment of dependencies more generally, the complexity of the two models we are forced into adopting is no longer equal. In both φ -agreement and case, we assume that the grammar, through some method, establishes dependencies between different syntactic objects. However, it's important to note that those who adopt dependent case explanations for the morphological forms of nominals, still adopt agreement-based models for establishing φ -agreement dependencies. In this way, dependent case is not purely an alternative model of case valuation, as it's often advertised, but it's an *additional* method of dependency establishment. Agreement-based models of case valuation do not require the assumption of an additional strategy for establishing dependencies between different syntactic objects because both are reflexes of the same operation. To this aim, we can level a reductionist argument against dependent case to the extent that under a Minimalist program, we should seek to minimize the number of operations and strategies that the grammar has available. Adopting dependent case requires the addition of a separate method of dependency establishment – a function the AGREE operation already readily performs.

2.3 Separation of Case from Licensing

As we saw in the previous section, dependent case theory allows for default case forms to surface when the algorithm has failed to assign them either lexical, dependent, or unmarked case. Clearly, this cannot be maintained in a framework where the failure to get case is fatal to the derivation, otherwise there would be no mechanism to prevent default case forms from being erroneously inserted into derivations where case has historically borne the theoretical explanation for ungrammaticality. A configurational approach to morphological case valuation therefore requires that case play no role in regulating the requirements that govern nominal licensing. This could be implemented in a few different ways: (i) it could mean that we eliminate nominal licensing requirements entirely (McFadden, 2004; Preminger, 2014) or (ii) it could mean maintaining those

requirements, but proposing that they're handled by something other than case (Levin, 2015). Both of these options are quite radical in that they upend central assumptions that mainstream theoretical syntax has held for over 40 years.

2.3.1 Motivations

In addition to the existence of default case, there have actually been a number of other things that have motivated researchers over the years to completely recast basic tenets of case theory. On the whole, they can be categorized as the recognition of the increasingly diminished role case is assumed to play in nominal distribution. In the early days of GB (Chomsky, 1981), case was assumed to be responsible for a host of disparate distribution facts and was argued to be one of the primary drivers of sentence construction.¹⁰ The need for case was what primarily drove movement (52a), what prevented superfluous movement (52b), what explained the inability of non-finite clauses to host overt subjects (52c), and what explained the distribution and form of nominals in passives (52d) and unaccusatives (52e), among other things.¹¹

- (52)
- a. John_{*i*} is likely *t_i* to win the race.
 - b. *John_{*i*} is likely that *t_i* will win the race.
 - c. *It is likely him to win the race.
 - d. John_{*i*} was invited *t_i*.
 - e. John_{*i*} arrived *t_i*.

Modern syntactic theory has since added a few theoretical tools that have greatly reduced the theoretical load that case carries (see (Levin, 2015; McFadden, 2004) for a detailed summary of these issues). With the adoption of the EPP feature, movement to the subject position in passives and unaccusatives no longer needed to be tied to the need for case on the moved argument. All clauses seem to require a subject and the feature responsible for encoding this need is what drives

¹⁰See Culicover (1997) for a summary. Also see Baker, Johnson, and Roberts (1989); Chomsky (1973, 1980).

¹¹This is not intended to be an exhaustive list, just a summary of some of the big facts.

the movement of the highest argument in the clause to the specifier of TP position. In this way, the theoretical work that case performed in this arena could be greatly reduced as it overlapped significantly with the EPP feature's role.

Likewise, the adoption of phase heads and spell out domains further reduced the role of case in regulating superraising. Compare (53a) with (53b). The nominal *John* is able to move out of the embedded clause to the matrix subject position in (53b), but is unable to do so in (53a).

- (53) a. * $John_i$ is likely (that) t_i will be sick.
 b. $John_i$ is likely t_i to be sick.

The case-dependent explanation for this was that because the nominal *John* receives nominative case from the embedded T in (53a), it is in effect “frozen” and therefore unavailable for further movement to another case position. Modern syntactic theory can rule out examples like (53a) through reference to the phase impenetrability condition (Chomsky, 2000) which disallows the movement of syntactic objects across spell out domains. The idea here is that the merger of a phase head triggers the spell out of its complement, rendering it inaccessible to further syntactic operations, with the exception of syntactic objects in its left edge position. So what disallows the movement of *John* in (53a) is that doing so would involve a movement across phase domains, one that is disallowed by the grammar. Because the embedded clause in (53b) is assumed to be a TP, rather than a CP – and thus not a spell out domain – the PIC does not prevent this movement.

2.3.2 Implications

If this varied set of distribution facts is no longer solely captured through a need for case, then it's reasonable to wonder if there is any role for abstract case to play at all, especially when we additionally consider the default case data discussed earlier. Many researchers have taken on this question, especially those who are inclined to prefer a dependent case model of case valuation (Levin & Preminger, 2015; McFadden, 2004; Preminger, 2014). Because dependent case models have been able to achieve an impressive scope of empirical coverage and their adoption depends

on being able to remove case's role in regulating nominal distribution, the separation of case from licensing has recently seen an increased focus.

Arguably, where we see the biggest impact of case today with respect to nominal distribution is where nominals can't appear, rather than where they can. The primary focus here is on the distribution of nominals in non-finite clauses. While this might be considered the "last bastion" for case theory, it's an arena where case still plays an important role and one in which we've not yet proposed an appropriate replacement. Among those who adopt a dependent case model, Levin (2015) is alone in maintaining that nominals still have a licensing requirement and that failing to meet that requirement is fatal to a derivation. He even maintains that case plays a role in this, albeit indirectly. He replaces the standard case filter shown in (54a) with the proposed alternative in (54b).

(54) a. *Standard Case Filter*

A nominal is licensed if and only if its unvalued case feature has received a value at spell out.

b. *Proposed Case Filter*

Noun Phrases must be KPs.

Levin argues that what dictates the ability of a nominal to appear in a particular position is its size: all nominals must be of size KP; they must include a K projection in order to be licit in the structure. He ties this to a grammatical requirement that all phrases include their maximal projections, arguing that the maximal projection for nominals is KP. He ties this idea to the original notion that case plays a role by arguing that this K position is the position that houses case features. In this way nominals need case not because they need case directly, but because they are required to be a big enough size where they include the projection that houses those features. He includes two additional "escape hatches" where nominals are licit, despite not being generated size KP: (i) the ability of nominals to late adjoin a K head and (ii) the ability of some nominals to adjoin to other nominal elements that include that maximal projection KP. The details of these escape hatches are

tangential to what I consider a minor criticism, so I direct the reader to Levin (2015) for a thorough discussion. What is theoretically unattractive about this style of approach is that it doesn't appear explanatory in the same way that a more standard case account might be. It more or less argues that the reason nominals are licensed is that they are simply generated licensed. It does however pave the way for us to adopt dependent case, which we've discussed at length in section 2.2.

A more direct criticism would be that this sort of proposal does not address the data in (55)-(56) which remains squarely in case's domain and Levin must follow others who offer independent reasons unrelated to nominal size or case to explain these distribution facts. For this discussion, I'd like to focus on two of these examples that do in fact have reasonable alternative explanations for their ungrammaticality. Under the traditional case story, what rules out both (55) and (56) is that the non-finite subject DPs *him* and *her* have failed to receive a case value, since non-finite T is not a case assigner and there is no other available source. Both McFadden (2004) and Levin (2015) have proposed similar alternative explanations for the ungrammaticality we see here.

(55) *John hoped him to win the lottery.

(56) *It is likely her to leave the party early.

The alternative explanation offered for (55) is that the ungrammaticality is not due to the inability of *him* to appear as a non-finite subject, but rather due to the inability of the complementizer *for* to be unpronounced. This solution is intended to be an extension of the Empty Category Principle and draws its inspiration from a similarity with the *that-trace* effect (Chomsky, 1981; Perlmutter, 1971; Stowell, 1981), shown in (57). The *that-trace* effect describes a generalization that the complementizer *that* is unable to appear overtly when it is followed by a trace. This was extended more broadly to be a generalization that the complementizer *that* must be dropped when followed by phonetically null subjects.

- (57) a. Who_i do you think t_i kissed Mary?
b. *Who_i do you think that t_i kissed Mary?

Citing a similar distribution to *that* McFadden and Levin suggest that a similar treatment can be extended to the complementizer *for*.

(58) **Complementizer optionality**

- a. I would like (for) him to buy the book.
- b. I believe (that) he bought the book.

(59) **Obligatoriness in CP subjects**

- a. [*(For) him to buy the book] would be preferable.
- b. [*(That) he bought the book] would be preferable.

(60) **C⁰-trace effects**

- a. Who_i do you think (*that) t_i bought the book?
- b. What_i do you think that he bought t_i?
- c. Who_i would you like (*for) t_i to buy the book?
- d. What_i would you like for him to buy t_i?

(Levin, 2015)

The idea is that like *that*, *for* is also banned from appearing overtly when it is followed by a phonetically null subject. So while the standard case-based account would rule out (61a) on the grounds that PRO is unable to receive null case from an empty complementizer, the ECP version would argue that what explains the ungrammaticality of (61a) is the failure of the complementizer *for* to be dropped when preceding a phonetically null nominal, PRO as required.

- (61)
- a. *John hopes for PRO to leave.
 - b. John hopes for him to leave.
 - c. *John hopes him to leave.
 - d. John hopes PRO to leave.

While this particular explanation can directly account for the distribution of *for* in examples like

(61a), where one needs to understand why the null version of *for* is required, it says nothing about examples where we need to understand why the overt version of *for* is required, as shown in (61b)-(61c). (61c) cannot therefore be ruled out due to an ECP violation, but instead must be ruled out through other means. What one would have to argue here is that [him to leave] in (61c) is a TP and that the verb *hope* is the kind of verb unable to take TP complements. Note, this also requires assuming that [PRO to leave] in (61d) is a CP. This of course is possible, but given that the tenability of this account depends on assuming a particular structure for the ungrammatical sentence in (61c) (and also for the grammatical sentence in (61d)) it wouldn't be unreasonable to lay an ad-hoc criticism against this approach. For one, it's not clear that we couldn't instead assume, as is more standard, that [him to leave] in (61c) is a CP with a null complementizer, especially when one assumes exactly that structure for (61d).

This sort of ECP extension also requires a less modern understanding of complementizer-trace effects and it is not clear that it could even be extended to Minimalist frameworks in the way McFadden and Levin intend. As Pesetsky (2017) notes, modern understandings of the mechanisms behind complementizer-trace effects do not focus on whether the complementizer is overt or not, as GB era versions did (and as the explanation above requires), but instead are about whether T to C movement is possible and/or obligatory. Pesetsky and Torrego (2001) offer a proposal of complementizer-trace effects that argues that syntactic objects like *that* and *for* actually originate in T and eventually move to C if they are attracted by tense feature probes on C. They assume C can have two probing features: one that attracts tense features and another that attracts wh-features. In a sentence where the wh-phrase is not in subject position, these probes will find matching goals on different syntactic objects – the tense head and the wh-phrase respectively. The tense feature probe will find a goal in the T head, thus triggering the movement of T to C, as it does in (62a). If *that* or *for* is what occupies that position, then *that* or *for* will be able to move to C and thus occupy a position to the left of the subject. The wh-feature probe will agree with and trigger the movement of the wh-phrase the question targets.

- (62) a. [CP Who_i do you think [C that_j [TP Sue t_j met t_i]]]?
 b. [CP Who_i do you think [C [TP t_i met Sue]]]?
 c. *[CP Who_i do you think [C that_j [TP t_i t_j met Sue]]]?

However, Pesetsky and Torrego (2001) argue that if the *wh*-phrase occupies the subject position, it is capable of valuing both the tense feature of C *and* the *wh*-feature of C, given its position in SPEC TP. Because the subject position constitutes a more local goal than T itself, this has the result of blocking the T head from being attracted by the probe and thus prevents T-to-C movement (62c). Since they assume that *that* and *for* originally occupy T, this will capture the inability of *that* or *for* to proceed a subject trace. (See Pesetsky & Torrego, 2001, for the details of their approach).

Both McFadden (2004) and Levin (2015) admit that the rules and principles that govern when complementizers can be overt or must be null are still very poorly understood and they don't offer many details beyond the comparison with *that* for how the complementizer effect might work. What's important about the proposal in Pesetsky and Torrego (2001) and the overview presented in Pesetsky (2017) is that they show that the ECP is not maintained in Minimalism as it was originally formulated. The basic patterns it intended to capture are argued to be better captured via the relationships between probes and goals and the sorts of constraints we assume there to be on movement rather than a type of filter that captures whether or not complementizers should be overt. This makes an ECP dependent explanation of the non-finite data discussed in this section quite untenable, at least under Minimalist assumptions.

Furthermore, the data in (63) also seems to suggest that the comparison between *that* and *for* that McFadden and Levin center their hopes for an ECP extension on isn't as strong as one would need if one wants to place the locus of explanation on similar behavioral patterns. What (63b) and (63d) show is that the null versions of *that* and *for* appear to have different behavior. There are two possibilities here: either both (63b) and (63d) involve null complementizers and we need to understand how and why they behave differently or the embedded clause in (63b) is a CP, while the one in (63d) is a TP and we need to understand why the two have different structures. The first possibility is unattractive in part because it is not at all obvious why two null complementizers

should have such different requirements. Furthermore, if one depends on those differences to be able to abandon a long-held central tenet of the grammar like case, then we should have a much better understanding of what those differences are before committing to doing so. The second possibility is also unattractive, as we've discussed earlier in this section, because it is essentially a stipulation – one that appears odd given the similar behavior and structures in (63a)-(63c). Notice that a case-based explanation here can single out (63d) as the unique member quite easily; it is the only example in (63) where the embedded subject is unable to receive case.

- (63) a. it's possible [_{CP} that he left].
 b. it's possible [_{CP} $\emptyset$ he left].
 c. it's possible [_{CP} for him to leave].
 d. *it's possible [_{CP} $\emptyset$ him to leave].

For these reasons, I argue that an ECP account isn't a tenable replacement for case theoretical explanations of nominal distribution in non-finite clauses. The ECP isn't really extendable in the intended way to Minimalist frameworks because we've since reframed how to conceptualize complementizer-trace data, it doesn't explain the obligatory presence of *for* in structures like (63c)-(63d), and it leaves questions about either differences in null complementizers or stipulated structures unanswered. I think this validates some real skepticism about whether abandoning case's role in regulating nominal distribution is a reasonable departure.

Moving to (56), repeated below as (64), the standard approach rules out this example on grounds that the DP *her* is unable to receive case from the embedded non-finite T, also signaling a preference for move over merge (Shima, 2000). The alternative that McFadden (2004) and Levin (2015) propose is that (64) is instead a violation of requirements on what sorts of things are qualified to serve as an associate of the expletive *it*. Levin (2015) explicitly proposes that the embedded clause in (64) must be a TP and argues that TPs are unable to serve as associates of the expletive. Since the only potential associate for *it* in (64) is the TP [*her to leave the party*], as shown below in (65), the resulting sentence is ungrammatical.

- (64) *It is likely her to leave the party early.
- (65) a. It is likely [for her to leave the party early.]
 b. *It is likely [her to leave the party early.]
 c. [CP for her to leave the party early] is likely.
 d. *[TP her to leave the part early] is likely.

Like the ECP argument outlined above, an appeal to expletive association requires assuming a particular structure for the ungrammatical sentence shown in (65b), one that is not obviously correct. There is an equally plausible alternative structure for this sentence that includes a null complementizer in the C position that is inconsistent with the proposal offered. Without clear motivation for selecting the TP approach, this explanation reduces to stipulation. When coupled with the ECP arguments above, I argue that we have not yet proposed a true viable alternative to the broad theoretical insights classical case theory has offered and thus its abandonment is premature.

2.4 Conclusions

With this discussion in our rear-view, it's important to take time to summarize where we are. In this chapter, I hope I have provided enough reasons for the reader to be more skeptical of adopting a dependent case model and by extension, its required precursor – the separation of case from licensing. My aim has been a modest one. With a modern and incredibly detailed account of how dependent case could operate in a grammar that is consistent with a traditional Minimalist framework, it is time to ask what the conceptual and theoretical implications of adopting this proposal are. I have provided some arguments, both empirical and conceptual, that suggest that the exploration of these issues gives us reason to pause and either go back and adjust the dependent case model to address the issues raised or to reject the system altogether and attempt to address the issues raised with a more standard approach. In this thesis, I intend to pursue the latter option, but hope the former is also taken upon by those more inclined.

CHAPTER 3

OBLIGATORY OPERATIONS

3.1 Introduction

This chapter is similar to chapter 2 in that it also attempts to explore the empirical and conceptual implications that adopting an alternative approach to failed valuation has on our understanding of the grammar. This chapter focuses on this issue in the domain of φ -agreement. As discussed in chapter 1, the existence of default agreement raises some interesting problems related to the valuation of φ -features and the relationship between agreement and grammaticality. As with case, the crux of the issue is that the existence of default agreement raises questions about how the functional heads that fail to establish a φ -agreement relationship survive to be spelled out without causing the derivation to crash. A perfectly reasonable way to address this issue is to modify our assumptions in a way that allows for the failure of agreement, meaning we must completely recast the grammatical conditions on φ -agreement. Preminger (2014) does exactly this by providing an alternative model of the grammar which encodes grammaticality requirements not by their success, but by their initiation. If an operation is triggered in the derivation, the grammaticality requirements are met. The grammar treats failed agreement as a reasonable outcome of the operation, so long as it was initiated upon the creation of its structural description. This model is quite radical in that it upends a large set of standard theoretical assumptions largely held by mainstream syntactic theory since Chomsky (2000, 2001) about what drives derivations. This chapter will detail an overview of what the grammar might look like if one adopts an obligatory operations model and will evaluate the implications of the proposal.

3.2 Obligatory Operations

The data used to frame the obligatory operations approach is from Hebrew and is shown below in (1), what Preminger (2014) calls *gratuitous nonagreement*. What (1) taken together shows is

that φ -agreement is the kind of operation that if it can apply, must. A general explanation for the ungrammaticality in (1b) is that it is a reflex of the failure of the operation behind φ -agreement to have applied, reinforcing the characterization that φ -agreement as an operation is obligatory. Had it applied in (1b), it would have caused the ‘correct’ φ -features to surface as they do in (1a).

- (1) a. ha-necig-im dibr-u
 the-representative-PL spoke-3PL
 ‘the representatives spoke.’
 b. *ha-necig-im diber
 the-representative-PL spoke(3SG.MASC)

(Preminger, 2014)

This explanation is consistent with a number of frameworks. The question becomes: what specifically is bearing the primary theoretical burden of enforcing the obligatory nature of φ -agreement. In a framework that uses the inability of uninterpretable features to survive to the semantic interface to drive derivations, the failure of the agreement operation would cause those features to remain unvalued and therefore a derivation crash would be expected. Preminger calls these unvalued features *derivational time bombs* and this is how he characterizes the modern standard approach that came out of the work first advanced in Chomsky (2000, 2001). It’s important to clarify here that it is entirely possible, and is quite common actually, to assume that even within a derivational time-bombs approach, probes immediately begin their search upon merge into the derivation. The point Preminger makes is that it is not the time-bomb nature of unvalued features at the interfaces that is driving the distinction in (1), it is the immediate and automatic probing.

Preminger argues that the best way to model the obligatory nature of φ -agreement, and potentially by extension other syntactic phenomena, is to instead propose that there are syntactic operations that are automatically, obligatorily, and immediately triggered upon the creation of the respective operation’s structural description. We can view the obligatory operations proposal as an attempt to reduce the theoretical complexity of the grammar by attempting to place the entire burden on the immediate and obligatory triggering of the operation behind φ -agreement, without reference to the time bombs themselves. In a standard model where both the immediate probing

and the presence of uninterpretable features is used to explain grammaticality distinctions, it would be useful, if possible, to reduce the theoretical burden so that it only depends on one of those.

Data like the default agreement data from Hindi-Urdu discussed in the introduction to this thesis (2) and failed agreement data that we will discuss shortly is used to push forward the alternative. What will unify these examples is that each involves the failure of the agreement operation to successfully transfer φ -features from goal to probe and the subsequent insertion of default feature values. What this data shows is that despite the failure to value the relevant uninterpretable features, the resulting sentence is perfectly grammatical.

- (2)
- | | | |
|----|--|-------------------|
| a. | Mona amruud khaa-tii thii
Mona.F guava.F eat.HAB.F be.PRF.F.SG
'Mona used to eat guava' | SUBJECT AGREEMENT |
| b. | Ram-ne imlil khaa-yii thii
Ram.M.ERG tamarind.F eat.PFV.F be.PST.F.SG
'Ram had eaten tamarind' | OBJECT AGREEMENT |
| c. | Mona-ne is kitaab-ko parh-aa thaa
Mona.F.ERG this book.F.ACC read.PFV.M.SG be.PST.M.SG
'Mona had read this book' | DEFAULT AGREEMENT |

(Bhatt, 2005)

Preminger concludes that this data shows that it is not the inability for unvalued features to survive that drives the obligatory nature of φ -agreement. Instead he proposes that what's behind both the grammaticality of (1a) and the ungrammaticality of (1b) is the existence of obligatory operations that do the actual transferring of features. It is the failure to be triggered when an operation's structural description is met that causes ungrammaticality. The grammar completely tolerates an operation that is triggered, but subsequently fails to culminate successfully.

First we'll survey the data used to advance the obligatory operations approach, with detailed explanations to follow. The primary data that Preminger uses to argue for obligatory operations comes from Kichean, a member of the Mayan language family that exhibits ergative-absolutive agreement alignment. Intransitive subjects show agreement with the verb and use an absolutive

marker to do so (3). Transitive subjects agree with verbs and use an ergative marker (italicized), while transitive objects use the same (bolded) absolutive marker used in intransitives clauses (4).¹

- (3) a. **ri achin** x-*Ø*-uk'lun
the man COM-**3SG.ABS**-arrive
 'the man arrived.'
- b. **rat** x-*at*-uk'lun
you(sg) COM-**2SG.ABS**-arrive
 'you(sg) arrived.'
- (4) a. *rat* x-*Ø-aw*-ax-aj **ri achin**
you(sg) COM-**3SG.ABS-2SG.ERG**-hear-act **the man**
 'you(sg) heard the man.'
- b. *ri achin* x-*a-r*-ax-aj **rat**
the man COM-**2SG.ABS-3SG.ERG**-hear-act **you(sg)**
 'the man heard you(sg).'

Of special concern for the obligatory operations proposal is a construction called *Agent Focus*, an example shown in (5). Agent Focus clauses are similar to transitive ones in that they have two arguments, but are similar to intransitives in that they only have one agreement slot. The two arguments therefore compete for agreement, obeying a φ -feature hierarchy (6) with 1st and 2nd person arguments being preferred over 3rd person arguments, and 3rd person plural arguments being preferred over 3rd person singular ones.

- (5) a. ja **rat** x-*at*/**Ø*-ax-an ri achin
 FOC **you(sg)** COM-**2SG.ABS**/***3SG.ABS**-hear-AF the man
 'it was you that heard the man'
- b. ja ri achin x-*at*/**Ø*-ax-an **rat**
 FOC the man COM-**2SG.ABS**/***3SG.ABS**-hear-AF **you(sg)**
 'it was the man that heard you(sg)'

- (6) 1st/2nd person > 3rd person plural > 3rd person singular

¹Unless otherwise noted, all Kichean data comes from Preminger (2014).

Further constraining the appearance of person features is a language-family-wide constraint (7) that bars two [participant] bearing arguments from co-occurring in this construction (8).

(7) *The AF person restriction*

In the Kichean AF construction, at most one of the two core arguments can be 1st/2nd person.

- (8) a. *ja rat x-**in/at/ø**-ax-an yin
 FOC you(sg) COM-**1SG.ABS/2SG.ABS/3SG.ABS**-hear-AF me
 Intended: ‘It was you(sg) that heard me.’
- b. *ja yin x-**in/at/ø**-ax-an rat
 FOC me COM-**1SG.ABS/2SG.ABS/3SG.ABS**-hear-AF you(sg)
 Intended: ‘It was me that heard you(sg)’

Another piece that will be relevant to accounting for the agreement patterns exhibited in Kichean AF constructions is the idea that the first and second absolutive agreement markers aren’t true agreement markers in Kichean, but are instead clitics: reduced, determiner-less versions of the strong pronouns. Table 3.1 below shows the similarities between the agreement markers on the left and the strong pronouns in the middle used to argue in part for this conclusion. Notice that these similarities do not exist in the ergative agreement marker paradigm, shown on the right. The table also shows that these similarities disappear for the 3rd person agreement markers, leading Preminger to conclude that in Kichean, only 1st and 2nd person absolutive markers are clitics; 3rd person agreement markers are instead the full expression of person and number.

Table 3.1: Kichean Agreement Markers

	ABS agreement marker	strong pronoun	ERG agreement marker
1sg	i(n)-	yin	n/w-
1pl	oj-	roj	q(a)-
2sg	a(t)-	rat	a(w)-
2pl	ix-	rix	i(w)-
3sg	ø-	rja’	r(u)/u-
3pl	e-	rje’	k(i)-

One final note about the language is that Preminger suggests we treat the [e-] morpheme as a plural morpheme. If true, we can understand the 1st and 2nd person markers/clitics as expressing person and number suppletively, while 3rd person markers do not share this property. This is one more piece of evidence that could motivate treating the [participant]-bearing morphemes different from the non[participant]-bearing morphemes, which will later be used to derive the AF person restriction which only involves the 1st and 2nd person arguments.

There are two parts to proposing an obligatory operations approach to φ -agreement in Kichean AF constructions: deriving the agreement paradigm itself and deriving the Agent-Focus person restriction that bars two [participant] bearing arguments from co-occurring in AF constructions. I'll discuss each in turn. First, there's an operation responsible for φ -agreement called `FIND`, shown below in (9). This operation is triggered automatically, obligatorily, and immediately upon the creation of `FIND`'s structural description. In this particular case, this essentially means that as soon as an unvalued feature f probe is merged into the structure, `FIND` will be triggered. One defining feature of this approach is that if the operation `FIND` fails, the grammatical requirements are considered met and there is no ungrammatical consequence because what is required is the attempt at agreement, not a particular result. We'll see how this failure helps to capture the default agreement data mentioned in the previous section (and throughout this thesis).

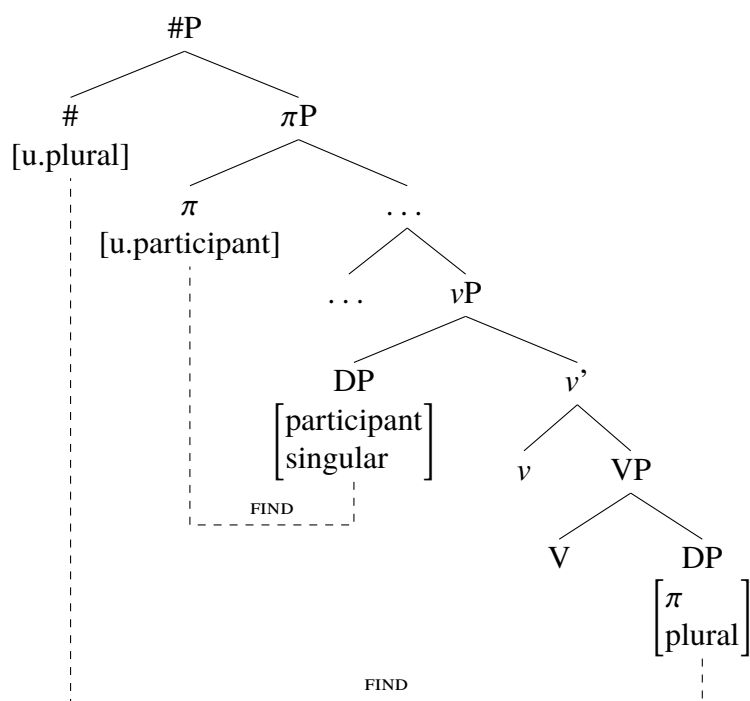
(9) `FIND( $f$ )`

Given an unvalued feature f on a head H^0 , look for an XP bearing a valued instance of f and assign that value to H^0 .

With respect to φ -features, Preminger assumes that φ -probes are separated onto two functional heads, a person head and a number head, that are each relativized and probe independently (10). In Kichean, the person head is located lower than the number head and therefore will probe first. The person head is relativized to probe for the featural specification [participant], encoding the language's preference for agreement with 1st/2nd person arguments, while the number head probes for [plural], encoding the preference for plural over singular arguments. According to `FIND`, the

person probe will search for an argument bearing a [participant] and will ignore arguments that do not bear this feature. Likewise, the number probe will search for arguments bearing [plural] and will ignore arguments that do not bear this feature.² In the tree below in (10), the dashed lines show that each probe is ‘looking’ for valued instances of each head’s relative probing φ -features in their respective c-command domains. Because the external argument is specified with a participant feature, the probe will ‘see’ it and establish an agreement relationship via FIND. Notice, however that the external argument in our example is *not* specified with a [plural] feature. This means that while visible to the person probe, the external argument is essentially invisible to the number probe. It therefore can bypass the external argument and FIND the plural feature on the internal argument instead.

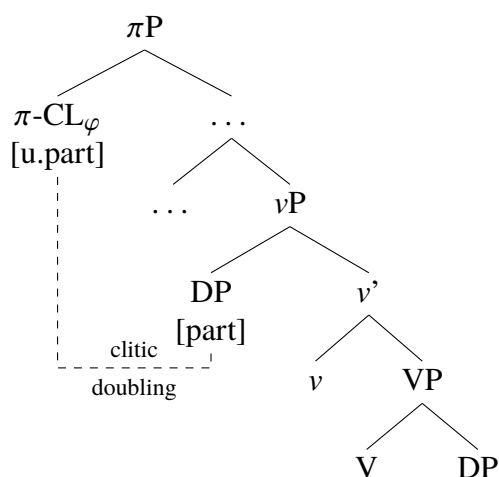
(10) *Kichean AF Probes*



²Two quick notes on notation: I’m using the form u.feature to represent that the feature on the probe is unvalued. However, under Preminger’s analysis, this feature is not of the same status as the typical uninterpretable features that cause derivation crashes. I’ll represent this distinction using the traditional form *u*feature when illustrating models that assume standard feature assumptions and the unitalicized u.FEATURE, with the added period for models where they are simply unvalued but aren’t assumed to cause derivation crashing.

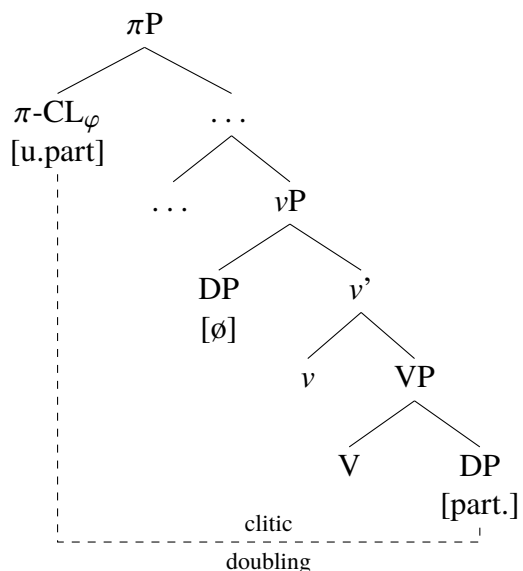
Being successfully probed by the person head triggers clitic doubling of the probed argument. Because the probe is relativized to look for [participant] features, this circumstance will arise when an argument bears a [participant] feature. If an argument is successfully probed by person, its entire φ -feature set is copied, not just the feature value for the person category, so only when the person head fails to probe an argument will the number morphology on the verb be an exponent of agreement. This is illustrated on the tree below in (11) and (12) via the CL_φ attached to the person probe head. This signals that all the φ -features for clausal agreement have been satisfied by the probing of person. If the subject is specified for [participant], the person probe, bearing a [participant] feature, will successfully probe the subject, triggering clitic doubling of the subject (11). If however the subject does not bear this [participant] feature, the person probe bearing [participant] will skip over this argument and continue to probe (12).

(11) *Clitic Doubling Triggered by [part] Probe*



If instead the object of the clause bears the relevant [participant] feature that the person probe is looking for, clitic doubling with the object will be triggered, as it was with the subject in the previous example.

(12) *Clitic Doubling Triggered by [part] Probe*



The clitic doubling assumption is crucial for the analysis of how agreement patterns are produced in Kichean AF because the effects of clitic doubling impact the environment in which the number head probes. This is because the person probe probes first – its outcome conditioning the environment the number probe will probe – and the result of clitic doubling is the valuation of the entire φ -feature set of the agreed-with argument. In this way, successful clitic doubling bleeds number probing because the [participant]-bearing argument has already valued the number features. To account for the Kichean AF Person Constraint, Preminger extends Béjar & Rezac’s Person Licensing Condition (Béjar & Rezac, 2003) which dictates that 1st and 2nd person features, namely the [participant] feature, must enter into an agree relation in order to be licensed (13).³

(13) *Person Licensing Condition (Béjar & Rezac, 2003)*

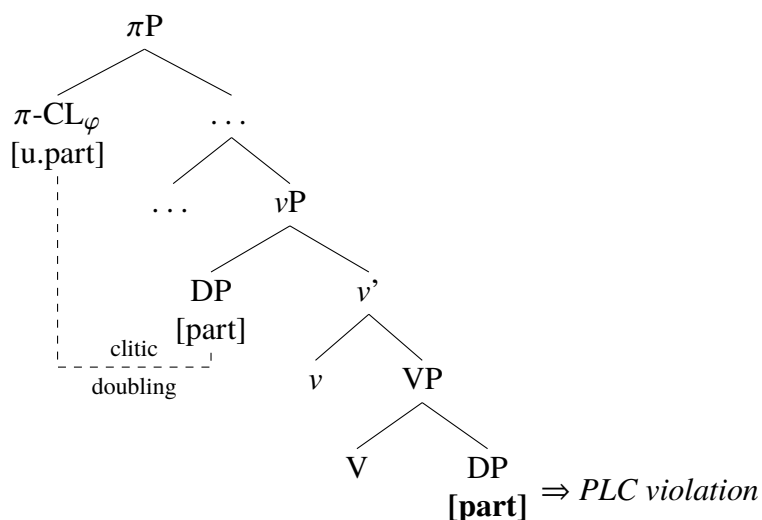
Interpretable 1st/2nd person features must be licensed by entering into an Agree relation with an appropriate functional category.

If both the subject and the object are specified with a [participant] feature, as they are in (14), the

³In one of the following sections in this chapter, we’ll address the observation that the PLC appears to employ a similar mechanics/derivational logic to derivational time-bombs and how Preminger handles this apparent similarity.

person probe probes into the higher argument, the subject, triggering clitic doubling as it did in (11). However, the derivation would be ruled ungrammatical when it is spelled out due to a violation of the Person Licensing Constraint because the [participant] feature of the object (bolded in (14)) would remain un-agreed with since only one agreement slot is available. In this way, Preminger accounts for both the preference for 1st/2nd person arguments and also the restriction that bars two of them from appearing in the same clause.

(14) *PLC Violation*



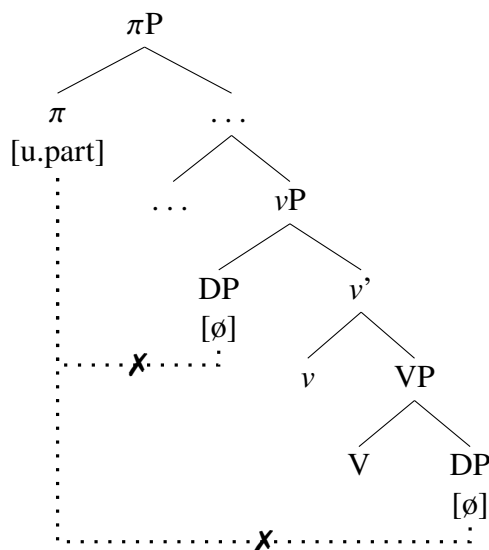
Two assumptions stand out as bearing the theoretical weight of explanation: the relativization of probes and the Person Licensing Condition. The relativization of probes is what derives the preference for 1st/2nd person arguments over third and (as we'll see below) the preference for plural arguments over singular ones. The PLC is what accounts for why two arguments in the same structure bearing [participant] are illicit.

If neither the subject nor the object is specified with a [participant] feature – namely both arguments are third person – then the person probe is unable to successfully probe either argument and clitic doubling isn't triggered at all. Importantly, these sentences, shown in (15) are perfectly acceptable. Preminger's analysis of this data requires a model of the grammar that allows for the agreement operation to fail without causing the derivation to crash. In (16) we observe that

the person probe that is searching for an argument bearing [participant] will be unsuccessful in a sentence where both arguments are 3rd person. What's central to this approach is that despite the failure of the [participant] probe to find an argument to value its uninterpretable feature, the derivation does not crash and the resultant sentence is grammatical. It is important to note here that under this *FIND* approach, the result of the derivation in (16) is not successful φ -agreement with either of the 3rd person arguments, but rather complete agreement failure and the assumed insertion of a default morphological form, here a null morpheme.

- (15) a. ja ri tz'i' x- $\emptyset$ -etzel-an ri sian
 FOC the dog COM-**3SG.ABS**-hate-AF the cat
 'it was the dog that hated the cat.'
- b. ja ri xoq x- $\emptyset$ -tz'et-o ri achin
 FOC the woman COM-**3SG.ABS**-see-AF the man
 'it was the woman who saw the man.'

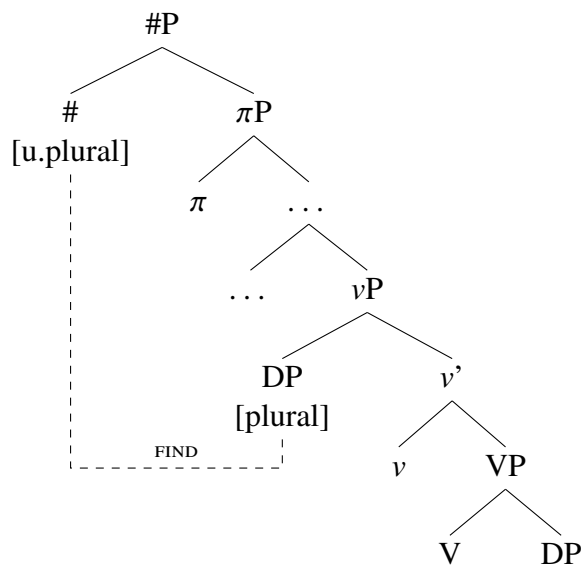
(16) *Failure of [π] Probe*



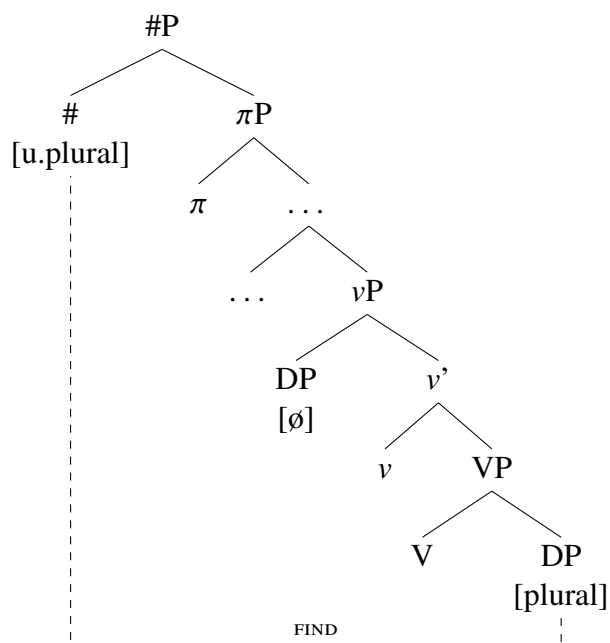
With respect to the number probe, we see a similar type of process. The φ -feature on the number head will only act as a probe when person agreement is unsuccessful. This is because successful person agreement triggers clitic doubling, which values the entire φ -feature set, both person and number. Only when number remains unvalued will it trigger *FIND*. Since the number probe is

similarly relativized, in this case for [plural], it is able to ignore or skip arguments that do not come specified for this feature. If the subject is specified with the [plural] feature, the number probe will successfully agree with it as in (17). If instead, the subject is not specified with [plural], the number probe will skip it and continue its search, eventually reaching the object. If this object has a [plural] feature, then the number probe will agree with the object instead (18).

(17) *Number Agreement with Subject*

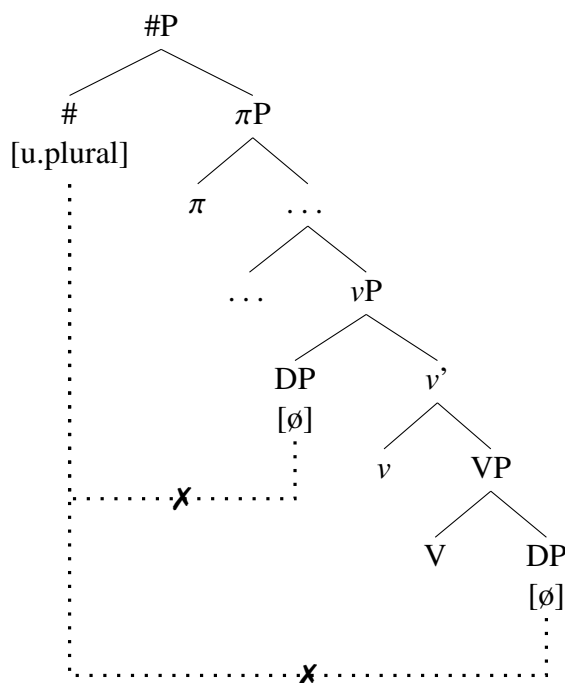


(18) *Number Agreement with Object*



However, if neither the subject nor the object have a [plural] feature as in (15), then the probe will have failed to agree with either argument (19). Under the assumptions provided by obligatory operations, this failure is tolerated by the grammar since the operation FIND at least was triggered and attempted to value its unvalued number feature. As with the person features, the failure of the number probe triggers the insertion of a default singular morpheme – once again the null morpheme – and is not reflective of a successful agreement relationship. It's also worth noting here that because there is not a similar Number Licensing Constraint, nothing would bar two [plural]-bearing arguments from existing in the same clause.

(19) *Failure of [#] Probe*



To summarize how obligatory operations accounts for agreement patterns, including failed agreement, in Kichean AF constructions, here's a quick review of the main assumptions and the theoretical work each does. The [participant] specification on the person probe is what derives the preference for agreement with first and second person arguments over third. The [plural] specification on the number probe is what derives the preference for plural arguments over singular ones. The assumption that clitic doubling triggers the exponence of the whole φ -feature set is what explains why number agreement is only truly distinguished in the third person. The requirement that [participant] features be agreed with in order to be licensed coupled with the availability of only one agreement slot is what accounts for the barring of more than one [participant]-bearing argument in an AF clause. Finally, the grammar's tolerance of failed operations is what produces default failed agreement when neither of the two arguments are able to satisfy a particular probe's relativized featural specification.

With an overview behind us, we can discuss what this failed agreement proposal is intended to mean for our broader understanding of the grammar. Preminger correctly recognizes that the grammar needs some way to address the fact that there are some features whose failure to receive a

value in a canonical way does not cause the derivation to crash. In Kichean AF constructions, these are sentences where neither argument bears a [participant] feature or where neither argument bears a [plural] feature. Approaches that rely on the failure to value features to determine grammaticality appear to constitute an incompatibility with this failed agreement data, as we've discussed in chapter 1 and chapter 2. To reconcile this issue, the obligatory operations approach removes the source of the “explosion” caused by derivational time-bombs surviving to the interfaces. What is crucial to the grammaticality of a certain derivation under the obligatory operations framework is not whether or not an unvalued feature successfully finds a valued counterpart, but rather that the operation applies whenever it can. The central point here is that this particular analysis of Kichean AF data means that whatever operation is responsible for valuing features is allowed to remain unsuccessful without causing the derivation to crash. It therefore cannot be the case that the grammar uses the need to value features to enforce grammaticality requirements alone.

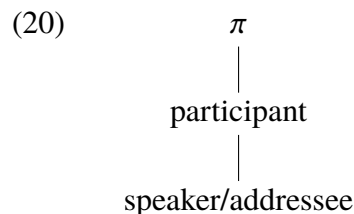
3.3 An alternative

The system proposed in Béjar (2003) similarly recognizes the need to account for data that involves the failure of the agreement operation in some regard. She argues that a solution can be found within a standard framework by decomposing the monolithic *AGREE* operation into two independent, but related, operations: *MATCH* and *VALUE*. Both operations are sensitive to the intrinsic hierarchical relationships held between features. What this does is it essentially expands the number of agreement outcomes from two to three. With a monolithic *AGREE* operation, there are only two possible outcomes: *AGREE* is either successful or it is unsuccessful (this is how *FIND* is assumed to operate.) Once we separate the operation into two suboperations, the application of one dependent on the successful culmination of the other, we expand the number of outcomes to three: *MATCH* can fail (which prevents *VALUE* from applying since successful *MATCH* is a precondition on the application on *VALUE*), *MATCH* can be successful and *VALUE* successful, or *MATCH* can be successful and *VALUE* unsuccessful. Béjar argues that this third outcome, made available only by the decomposition of *AGREE*, is exactly what produces the set of data that appears confounding to

derivational time-bombs approaches. It is in this arena that unusual agreement patterns that appear to involve failure can surface. The existence of an alternative proposal that largely still assumes the standard assumptions about grammatical requirements being enforced by feature valuation means that we no longer *must* adopt a FIND approach on empirical grounds. This reframes the obligatory operations discussion as a comparison between two approaches, rather than an empirically required necessity.

3.3.1 An overview of MATCH/VALUE

Béjar’s system similarly employs privative features that are structured hierarchically. These hierarchical relationships are derived from inherent entailment relations that the features’ semantics require. For example, the feature [speaker] semantically encodes that the bearer of the feature has the semantics of being a speaker in the event represented by the clause/verb; we traditionally call this first person. There’s another feature [participant] which encodes that the bearer of that feature is a participant in some event. Since being a speaker in an event semantically entails that one is also a participant in the event, we can derive a hierarchical relationship (20) based on entailment between the two features. Since speaker semantically entails participant, we assume that the feature [participant] dominates (or is hierarchically “higher”) than the [speaker] feature. (See Harley and Ritter (2002) for a detailed view/proposal of this feature system.)



What’s particularly important about these hierarchical relationships in Béjar’s system is that she argues that the syntactic operations responsible for φ -agreement are actually sensitive to (and depend on) these relationships. Agreement is understood to be composed of two suboperations, independently applying, each with its own set of conditions upon which they succeed. The

operation called `MATCH` is responsible for identifying which among the NPs in a c-command domain is available or visible to the target for agreement. It does not, importantly, decide which argument will actually control agreement; rather it circumscribes the set of arguments which are viable possibilities. A successful match is one in which the features of the goal match the features of the probe. The conditions upon which this is successful are when the features on the probe are a subset of the features on the goal (21). The operation `MATCH` is only evaluated with respect to the root feature. A more colloquial way of explaining this system so far is to say that probes are first looking for a certain *type* of feature category, rather than a particular value and want to first identify which syntactic elements could be potential goals by minimally having the right feature category. By evaluating this operation at the root, we are encoding that at this point, the grammar cares less about the particular value that an argument has and more about whether or not that argument is visible. The subset relationship essentially says that it's okay if the goal has more featural information than the probe (is more highly specified), but it doesn't count as a match if the probe has features that aren't specified on the goal. In table 3.2, we see that the probe will match a goal so long as the goal shares the same root feature of the probe. Whether the goal has more (line 1) or less (line 3) structure than the probe doesn't affect the success of `MATCH` at this stage. What would cause `MATCH` to fail would be if a goal did not share that root feature, either by being specified with a completely different type of feature (line 4) or by being specified with no feature at all (line 5).

(21) `MATCH`

A probe matches a goal if the root feature of the probe is either a subset of or identical to the root feature of a goal.

Table 3.2: MATCH outcomes

Probe	Goal	MATCH Outcome
$[\pi \text{ [part]}]$	$[\pi \text{ [part [speaker]]}]$	success
$[\pi \text{ [part]}]$	$[\pi \text{ [part]}]$	success
$[\pi \text{ [part]}]$	$[\pi]$	success
$[\pi \text{ [part]}]$	$[\# \text{ [plural]}]$	failure
$[\pi \text{ [part]}]$	$[\emptyset]$	failure

The other half of φ -agreement is due to the operation **VALUE**. **VALUE** is the operation responsible for actually establishing the relationship between a probe and a goal: an agreement target and its controller. It has stronger conditions upon which its success is evaluated (22). The subset relationship that defined the condition on **MATCH** is shared by **VALUE**, but where the **VALUE** conditions are stronger are in that this subset condition is evaluated not just at the root, but at the level of the entire feature set. So where the success of **MATCH** wasn't concerned with whether a potential goal has more or less feature structure than a probe, **VALUE** is only successful when a goal has more of the same featural structure than a probe (line 1), not less (line 3). **VALUE** also depends on the successful culmination of **MATCH**; lines 4 and 5 of table 3.3 show that the conditions of **VALUE** aren't even evaluated in the instances where **MATCH** had failed.

(22) **VALUE**

A probe is valued by a goal if the features of the probe are either a subset of or identical to the features of a goal.

Table 3.3: VALUE outcomes

Probe	Goal	VALUE Outcome
$[\pi \text{ [part]}]$	$[\pi \text{ [part [speaker]]}]$	success
$[\pi \text{ [part]}]$	$[\pi \text{ [part]}]$	success
$[\pi \text{ [part]}]$	$[\pi]$	failure
$[\pi \text{ [part]}]$	$[\# \text{ [plural]}]$	N/A
$[\pi \text{ [part]}]$	$[\emptyset]$	N/A

The separation of φ -agreement into two independently applying operations creates a three-way set of outcomes: **MATCH** can fail, both **MATCH** and **VALUE** can succeed, or **MATCH** can succeed, but

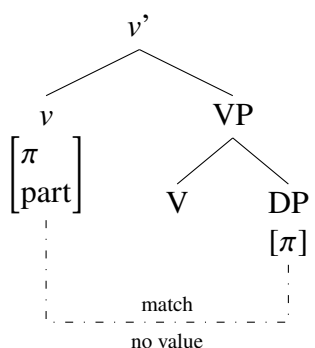
VALUE can fail. These outcomes are summarized in table 3.4 below. This of course does not solve the original default problem – that failure to successfully value unvalued features should produce a derivation crash. To obviate this worry, what she proposes is that the failure to either match or value a potential goal in some given domain marks the offending features for deletion. This is called *partial default agreement*. In a sense, this formally encodes that what the grammar cares about is the attempt, as is done in the obligatory operations approach.

Table 3.4: MATCH and VALUE interactions

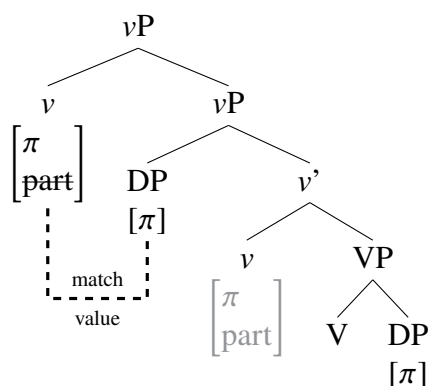
Probe	Goal	MATCH Outcome	VALUE Outcome	Result
$[\pi \text{ [part]}]$	$[\pi \text{ [part [speaker]]}]$	success	success	φ -agreement
$[\pi \text{ [part]}]$	$[\pi \text{ [part]}]$	success	success	φ -agreement
$[\pi \text{ [part]}]$	$[\pi]$	success	failure	probe is stripped 2nd cycle is triggered

When a goal matches, but fails to value a probe, as in (23), the grammar strips the probe of its featural content (minus the root feature) and a second cycle of agree is triggered, where the probe is able to continue its search in the expanded domain created by the merge of the ‘next’ syntactic object (24). Because the probe has been modified upon the second cycle, the properties a goal must have to be considered a successful VALUE are also modified. While a third person argument was not considered a viable agreement controller upon the first cycle of agree (23), it *is* considered a viable controller upon the second cycle of agree (24). If upon the second cycle of agreement, a probe is still unable to find an agreement controller, even with the reduced featural specification on the projection of the probe, the agreement operation is allowed to fail without consequence since partial default agreement already marked the feature for deletion. Total default agreement occurs at the point when this second attempt at agreement is unsuccessful. Essentially, the distinction between the two is that partial default agreement is the result of agreement with an impoverished feature set, while total default agreement is the result of a complete and total failure to agree. This is importantly not a distinction argued for in Preminger (2014) and we’ll return to this point in the next section.

(23) *1st Cycle VALUE Failure*

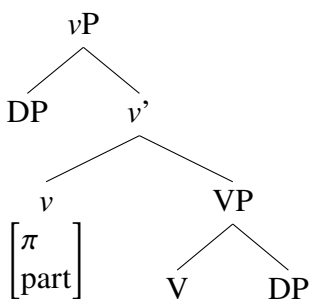


(24) *2nd Cycle VALUE Success*



To see how this system works more explicitly, let's look at an agreement pattern similar to Kichean – φ -agreement in Georgian (Aronson, 1989; Hewitt, 1995). Georgian φ -agreement reflects a general preference for agreement with objects, if those objects are either first or second person. If the object is third person, then agreement shows a preference for a first or second person subject. In a clause with no first or second person arguments, default or third person agreement is observed. Georgian person probes are assumed to be specified for [participant], like Kichean probes, to capture the preference for first and second person arguments and are assumed to be located low on v to capture the preference for agreement with the syntactic object (25).

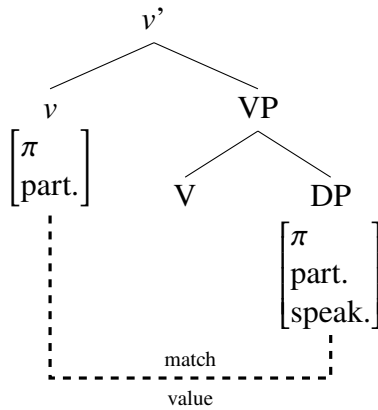
(25) *Georgian [π] Probe*



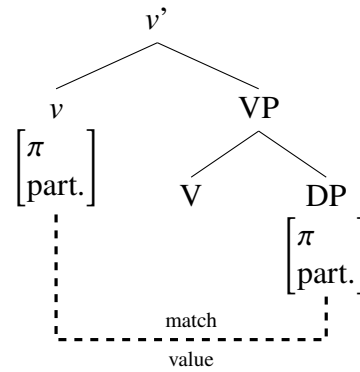
What this means is that the syntactic object will be the first argument a probe will encounter upon its search. If that object bears a [participant] feature, then MATCH will succeed, since both the probe and the goal contain the root [π] feature. VALUE will also succeed since the features on the

probe will constitute a subset of the features specified on the goal. With **VALUE** successful, features on the probe will be valued and φ -agreement will be the outcome, either first person agreement (26) or second person agreement (27).

(26) *1st Person Object Agreement*

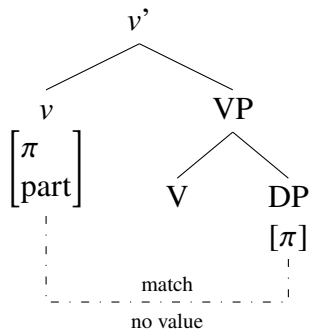


(27) *2nd Person Object Agreement*



If instead the Georgian object is third person, therefore not [participant]-bearing, **MATCH** will still be successful, since both probe and goal contain the root $[\pi]$ feature. Unlike the previous example, however, **VALUE** will not be successful as the probe's featural content is not a subset of the goal's (28). This also illustrates the context where partial default agreement is triggered, reducing the probe's feature set to only bear the root feature.

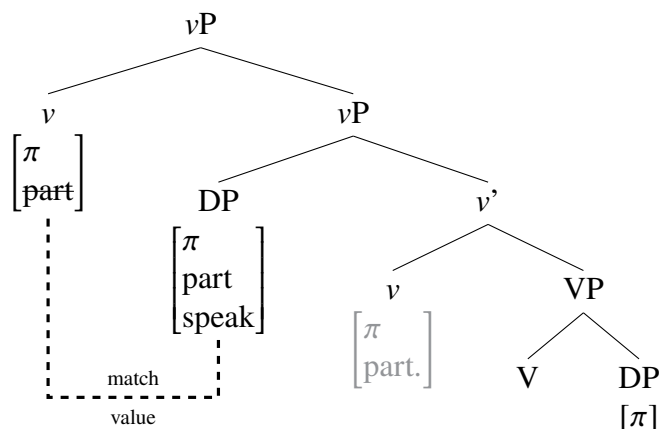
(28) *First Cycle VALUE Failure*



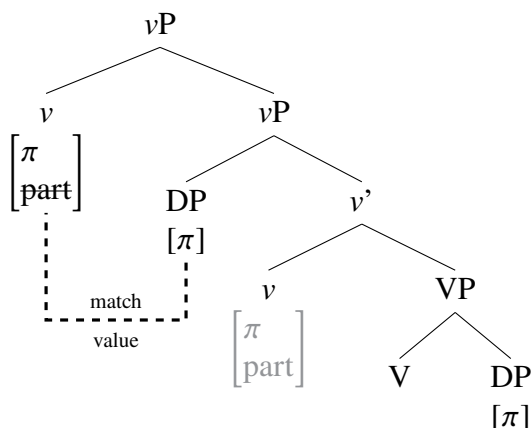
A second cycle of agreement is also triggered upon the projection of the probe's unvalued features. Upon this second cycle, which now has the external argument in its search domain, the

probe searches for a match and a value, as it did on the first. What's different on the second cycle is that, due to the modified featural content of the probe, the set of goals that could successfully *VALUE* the probe is larger. If on this second cycle, the probe finds an external argument with [participant] features, both *MATCH* and *VALUE* will be successful (29), as they were on the first cycle. If however, the external argument is third person, we'll also see successful valuation. Since the probe's featural content is now just [π], *MATCH* will be successful, as it had been upon the first cycle, and *VALUE* this time will also be successful, as the probe's feature specification is now identical to that of the goal (30). What this system allows then is an attempt to agree with a first or second person argument, but if that agreement attempt fails, agreement with the third person is possible. The last resort nature of this third person agreement is captured by it only being possible in situations where the attempt at agreement with a more specified set of features was unsuccessful. Both the regular outcomes of agreement are captured and the default flavor of failed agreement are captured in this system.

(29) *2nd Cycle 1st Person Agreement*



(30) 2nd Cycle 3rd Person Agreement



3.3.2 Accounting for Kichean

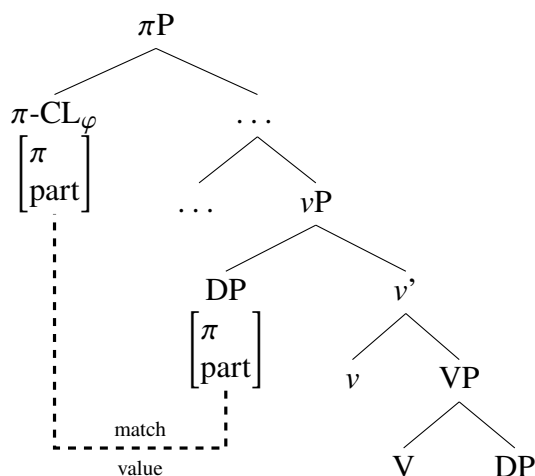
This model can easily account for the Kichean failed agreement data that we talked about in the last section. It's worth noting here that the details of *how* the MATCH/VALUE approach accounts for the Kichean agreement patterns depend on various assumptions one makes about the featural specification of third person arguments and which functional heads are assumed to house which probes. While the *how* details vary however, the conclusion that the MATCH/VALUE is capable of accounting for failed agreement in Kichean through second cycle agreement effects remains the same. I'll do my best to outline how the details may vary, but the bigger takeaway here is that regardless of how one sets these various parameters, the MATCH/VALUE approach is extendable to the Kichean data.

In many ways, the Kichean AF agreement patterns are actually a bit simpler than the types of person hierarchy patterns found in Béjar (2003), making the task of applying the MATCH/VALUE approach to Kichean very straightforward. Kichean has a relatively simple hierarchy, preferring [participant]-bearing arguments and preferring plural arguments over singular ones when number is an exponent of φ -agreement and not clitic doubling. This compares to a more complicated person hierarchy found in Nishnaabemwin which not only prefers second over first person, but also first person over third. This allows the Kichean probe to be less specified than the Nishnaabemwin one.

The details of how Nishnaabemwin agreement work will be discussed in section 3.4.

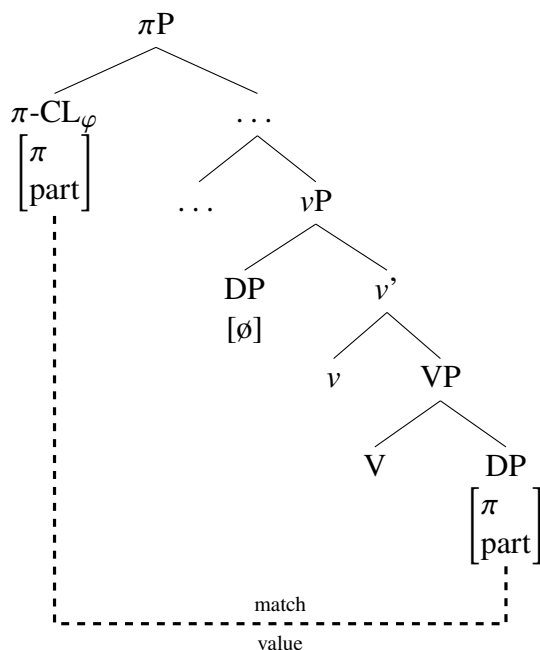
Preminger assumes that probes for both person and number are at least higher than the external argument. For him, this results in only one agreement cycle, which is either successful or not. The way that Béjar's system would work depends on where one assumes those probes are located. First, let's see how the system would handle the data if the probes are both located higher than the external argument, as Preminger does. If a [participant]-bearing argument is in subject position, then agreement is straightforward (31). Both MATCH and VALUE are successful and we can carry over the assumption that agreeing with a person head triggers clitic doubling and the subsequent copying of the entire φ -feature set, rather than the person features alone .

(31) *Kichean* [π] Agreement with Subject



If the [participant]-bearing argument is in the object position, rather than the subject position, the probe is able to skip over the non[participant]-bearing subject and agree with the object, just as it did in Preminger's account. How the probe skips over a non-[participant]-bearing subject depends in part on how one assumes third person features are specified in Kichean. We could assume, as Preminger does, that third person features are not specified at all and if true, the probe would quite simply ignore a third person subject (32), as it does in the FIND approach, subsequently agreeing with the [participant]-bearing object.

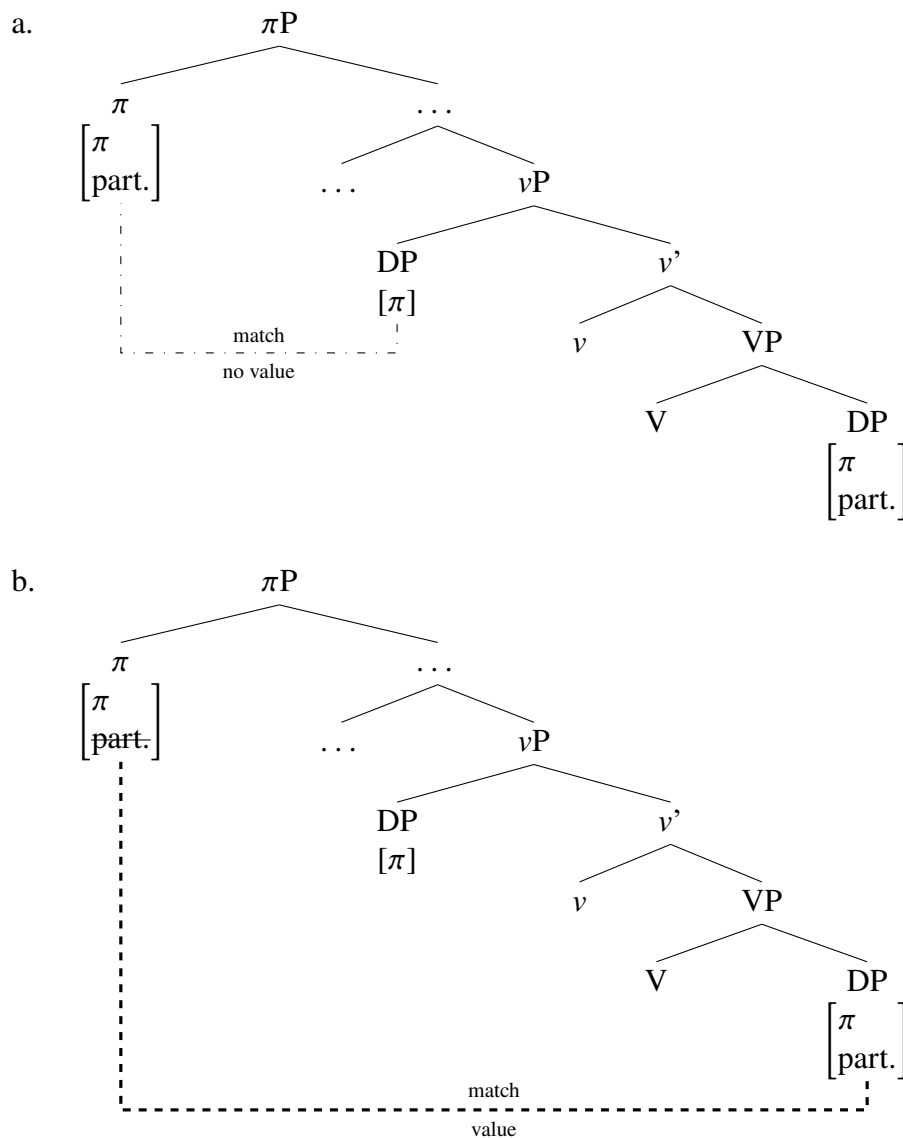
(32) *[π] Probe Skips Subject*



If instead we assume that third person has featural content, a [π] feature, then the subject would MATCH, but not VALUE the probe (33a). When probes are located low, this triggers a second cycle of agreement; however since the probe is high, it has not exhausted its search domain. The ability to MATCH, but not VALUE should trigger partial default agreement, as it always does and the probe could then presumably continue to search, discovering the [participant]-bearing object where both MATCH and VALUE would then be successful (33b).⁴

⁴Béjar doesn't have any examples of languages with a high person probe, so it's not immediately clear what she assumes happens when a probe find a MATCH but no VALUE if there's another potential agreement controller lower in the search domain. It could be the case that the higher intervener needs to move in order for the probe to continue to search. If true, then perhaps the movement of the external argument to subject position would move the external argument "out of the way" of the probe and it could keep searching. Alternatively – and this will be what I assume here – we could just say that the probe continues to search if it finds a MATCH, but no VALUE, as it does when probes are low.

(33) *3rd Subject has Features*



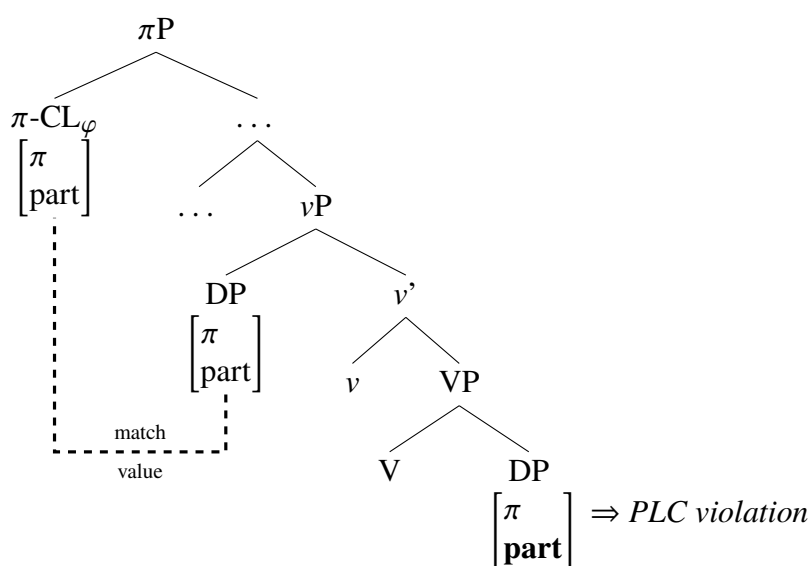
Finally, this approach could adopt the same assumptions Preminger does regarding the AF person restriction which bans two [participant]-bearing argument from co-occurring by adopting the PLC, repeated below in (34). Since the PLC for Kichean was adopted from Béjar and Rezac (2003) who also assume Béjar's MATCH/VALUE model, it's easy to see that it can exist in a MATCH/VALUE system as well. If both the subject and the object are specified with a [participant] feature, the person probe would probe into the higher argument, the subject, triggering clitic doubling as it did in (31). However, the derivation would be ruled ungrammatical when it reached spell out due to a violation

of the Person Licensing Constraint because the [participant] feature of the object, bolded in (35), would remain un-agreed with since only one agreement slot is available.

(34) Person Licensing Condition (Béjar & Rezac, 2003)

Interpretable 1st/2nd person features must be licensed by entering into an Agree relation with an appropriate functional category.

(35) *PLC Violation*

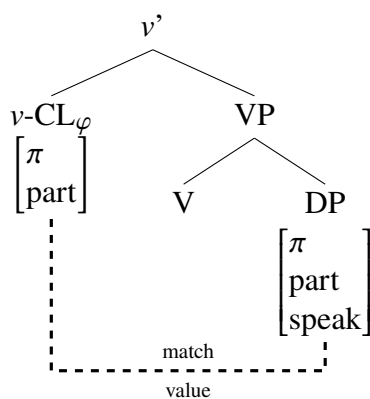


Béjar takes the featural specification of third person arguments to be a point of cross-linguistic variation. For some languages, like Nishnaabemwin, third person arguments have no featural content with respect to person. Others, like Georgian, third person arguments are specified with a minimal featural specification for person. Béjar suggests what decides this is whether or not the third person in the particular language ever exhibits any sort of intervention effects. If so, then the third person needs featural specification to be visible; if not, it seems the null hypothesis is that there's no reason to think that third person comes with any features if it doesn't use them for something. Preminger implicitly assumes that third person features are unspecified, but doesn't explicitly rule out the possibility that they have featural content. This question, is a continual discussion in much of the φ -agreement literature (see Nevins, 2007, for an overview) and while

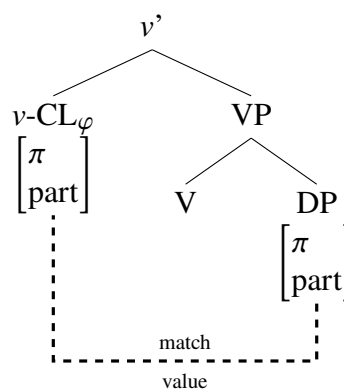
doesn't effect the ability of either approach to account for the failed agreement data in Kichean, does affect how the derivation proceeds at least in the MATCH/VALUE approach.

Finally, I'd like to end with the characterization that Béjar herself would likely adopt and that's an approach that has at least the person probe lower, on v . Preminger himself offers this as a hypothetical possibility, but winds up rejecting it because doing so under his system would render the external argument unable to be reached by the probe – a signal that he intends his FIND operation to *not* trigger a second cycle of agreement upon failure (see next section for a discussion of the implications of this move). A low person probe would encounter the object upon its first cycle, only if agreement with the object was unsuccessful due to a lack of [participant] features. If the object is [participant]-bearing, both MATCH/VALUE would be successful and φ -agreement would result (36)-(37).

(36) *1st Person Agreement*



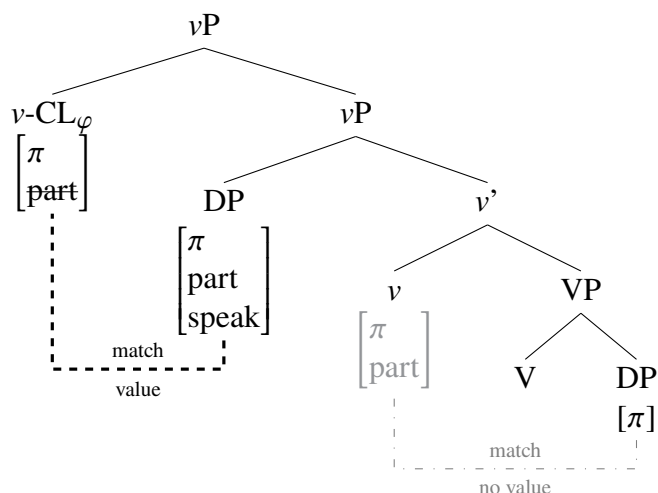
(37) *2nd Person Agreement*



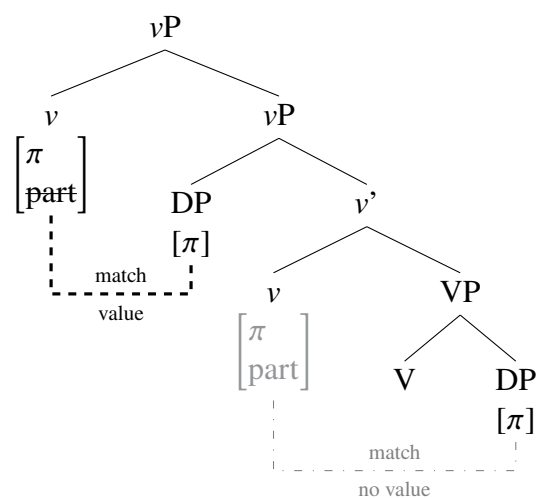
If the object was instead third person, then MATCH would succeed, but VALUE would fail, triggering a second cycle of agreement and again stripping the probe of features to its root. If upon this second cycle the probe encountered a [participant]-bearing subject (38), MATCH/VALUE would be successful and φ -agreement would result with the subject. If the subject was instead third person (39), then the probe would actually agree with the subject as well, because the probe's featural specification would be reduced as the result of a failure to VALUE on that first cycle. If we look back to the discussion of how Georgian is accounted for, the Kichean data would work very much

the same if we assumed Kichean person probes are low which is not surprising since the languages obey largely the same person hierarchy preferences.

(38) *Second Cycle Agreement with 1st Person*



(39) *Second Cycle Agreement with 3rd Person*



One difference between the two approaches is that a **FIND** approach treats third person exponence as a complete failure to agree, while a **MATCH/VALUE** approach treats third person exponence as φ -agreement with a third person, at least if one assumes third person has true featural content. If one instead assumes that third person is featurally empty, then both models characterize third

person exponence as the result of failure. There isn't really a way to decide between these options, at least empirically, but it's important to note there is a derivational distinction.

What the proposal illustrated here shows is that it is possible to model failed agreement within a framework that uses uninterpretable features to enforce grammaticality. This is achieved by exploiting a third outcome of agreement where these default-like patterns appear. To be clear, the existence of this approach does not show that uninterpretable features *must* be behind the obligatory nature of these operations. Recall that while Preminger hints that we might be able to do away with derivational time-bombs entirely and does mention failed agreement being incompatible with the standard derivational time-bombs approach, it's only the *obligatoriness* of φ -agreement that his argument references. His central proposal is more modest: one that says that uninterpretable features are not necessarily to be eliminated from the system, merely that they do not bear the theoretical burden of enforcing why operations like φ -agreement are obligatory. Within a Béjar-style system, one could certainly argue, as Preminger presumably would, that the probes immediately begin their search upon merge into the derivation and this immediate probing is what enforces why φ -agreement is obligatory.

However, with the existence of a possible alternative comes a shift in how the discussion should be framed. No longer does failed agreement *necessitate* obligatory operations, it merely provides a context through which we can judge the fitness of multiple alternatives. This will be the focus of the next three sections, where we examine data that similarly involves failed agreement, but provides a lens through which the two specific proposals may be better differentiated.

The main criticism leveled against the approach advanced in Béjar (2003) is that one can view the partial default agreement that she proposes as a type of diacritic, and thus an understandably unattractive feature of the framework that should be avoided (Preminger, 2014). Preminger argues that since that type of approach would need to track both the attempt and the outcome of agreement, assuming uninterpretable features enforce grammaticality is redundant. Under obligatory operations, nothing arguably needs to be tracked as the attempt is triggered obligatorily and automatically and the outcome is inconsequential for grammaticality. The next section will serve to illustrate that

when we view agreement patterns that are more complicated than the Kichean AF patterns, partial default agreement becomes a necessary feature of the model. There is reason to think that partial default agreement is not only *not* a diacritic in the traditional sense, but also is empirically needed. The resulting conclusion is therefore one that provides further credence to continuing to explore the tenability of a MATCH/VALUE approach to failed agreement.

3.4 Failed Agreement isn't Always Default Agreement

Looking solely at the FIND operation, we might be tempted to think that φ -agreement outcomes are quite simple; φ -agreement either succeeds or it fails and the grammar has the means to handle each. In this section, we'll see that the outcomes of failed agreement are more diverse than this binary distinction and we'll explore what this varied outcome set means for each of the proposals under discussion. In sections 3.4.1.1-3.4.1.2, I discuss data that shows that agreement must be able to continue to try to find a feature value after its initial failure and explore implications for FIND. In section 3.4.1.3 I illustrate how a FIND approach runs into trouble with data that exhibits a higher degree of sensitivity in the hierarchy that governs its person feature preferences. In section 3.4.2, we discuss implications for capturing dative intervention effects, and in section 3.4.3, we investigate failed agreement as a tool to resolve conjunct agreement conflicts.

The data discussed in this section shows two things: (i) that there is an empirical difference between the kind of failed agreement that results in partial default agreement and the kind of agreement failure that results in total default agreement and (ii) that the outcomes of failed agreement aren't uniform. I use these conclusions to argue against an obligatory operations approach on the basis that it does not predict a distinction between the two. I suggest that unvalued features might serve to mediate these more complicated outcomes and thus shouldn't be removed from the system.

3.4.1 Person Hierarchy Effects

We begin our discussion of the outcomes of failed agreement by looking at some data from languages that exhibit *person hierarchy effects*. What defines these languages, a group to which Kichean

(40) a. ja **rat** x-**at**/*-ø-ax-an ri achin
FOC **you(sg)** COM-**2SG.ABS**/*3SG.ABS-hear-AF the man
'it was you that heard the man'

b. ja ri achin x-**at**/*-ø-ax-an **rat**
FOC the man COM-**2SG.ABS**/*3SG.ABS-hear-AF **you(sg)**
'it was the man that heard you(sg)'

Languages with person hierarchy and second cycle effects are especially relevant for discussing instances of failed agreement as failed agreement is assumed to be the mechanism behind agreement

107

displacement (Béjar & Rezac, 2009). The general idea is that in languages where the agreement preference is initially with the internal argument unless it bears a certain set of φ -features, the displacement of agreement to favor the external argument is captured through the ability of the agreement probe to reattempt agreement upon a second cycle of φ -agreement. We'll discuss three types of data in this subsection: (i) languages with a morphological sensitivity to agreement cycles, (ii) languages with a syntactic sensitivity to agreement cycles, and (iii) languages with a complicated hierarchy preference that necessitates probe modification upon the second cycle of agreement. Given that Kichean was characterized as having high φ -agreement probes in Preminger (2014), it's worth understanding what the FIND proposal would mean for languages with lower φ -agreement probes that often trigger additional cycles of the operation behind φ -agreement. The focus will of course be on how the FIND proposal and the MATCH/VALUE proposal handle these sorts of data and what conclusions we can draw from those results. What we will conclude from this discussion is that the FIND operation – without a reasonable mechanism for reapplication nor with the ability to modify the featural specification of probes – is largely unable to sufficiently handle this more complicated data.

3.4.1.1 Morphological Effects

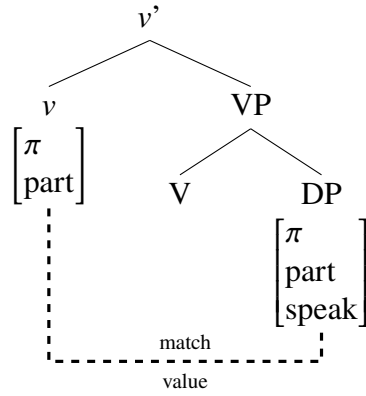
In a number of languages, the morphological affix that is inserted for a certain person value depends on whether the probe received that person value upon the first attempt at agreement or upon the second attempt triggered by failed agreement. Georgian provides an illustration. Georgian person probes are located low, on *v*. They are specified for $[\pi[\text{part}]]$ reflecting a preference for arguments bearing a participant feature, either 1st or 2nd person arguments.

Relevant to second cycle morphological effects, Georgian has a morphological alternation called the *m*-/*v*- alternation (Béjar, 2003) where there are two distinct sets of morphemes: an *m*-set and a *v*-set. These are shown in table 3.5. The *m*-set of morphemes is inserted for the given person value when agreement is successful on the first try (41a) and the *v*-set is inserted when the probe's value came from agreement on the second attempt (41b).

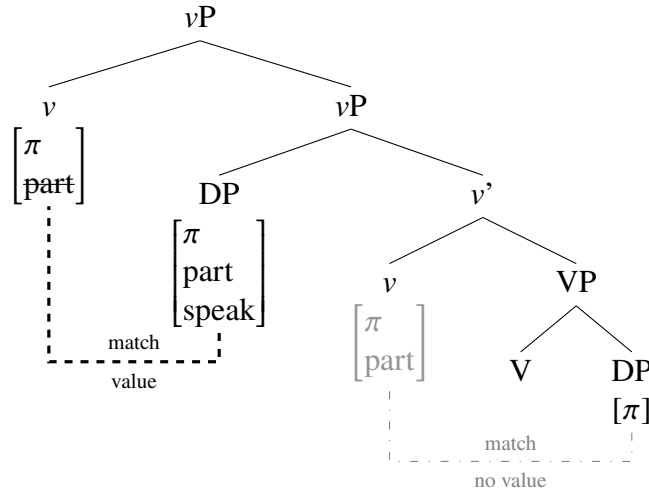
Table 3.5: Georgian Agreement Morphemes

Person	m-set morpheme	v-set morpheme
1st	m-	v-
1st (inclusive)	gv-	N/A
2nd	g-	x-/ø
3rd	N/A	ø

(41) a. *First Cycle (m-set morphemes)*



b. *Second Cycle (v-set morphemes)*



The morphological distinction between the two sets shows a sensitivity not only to the difference between partial and true default agreement (exhibited by a difference between first/second and third person morphemes within each set), but also between more canonical successful agreement and partial default agreement (exhibited by the existence of two morpheme sets). The m-/v- alternation

shows that the grammar is sensitive to a three way distinction – a more complicated set of outcomes than the simple tolerance of failed agreement models.

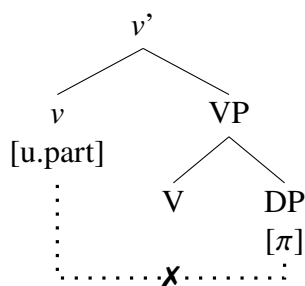
The `MATCH/VALUE` approach quite obviously captures this data as the triggering of a second cycle of agreement is explicitly built into the system as a result of failed first cycle agreement. Along with the triggering of a second cycle, the `MATCH/VALUE` system also assumes that the probe is modified in a way that it becomes less specific about the person features that would qualify to establish a successful agreement relation. This probe modification is not a crucial feature for Georgian agreement patterns, as the Georgian person hierarchy, like Kichean's, is relatively simple. So while not particularly relevant here, it will become so for other languages (see section 3.4.1.3). What *is* crucially important is the fact that failed agreement for many of these languages does not immediately trigger default agreement, suggesting that the outcomes of failed agreement are more complicated than the Kichean AF facts imply. Instead, in languages like Georgian, failed agreement triggers a second cycle of agreement that could result in either true person agreement with an argument in the expanded search domain or it could result in default agreement.⁶

It's not impossible for us to modify the `FIND` operation to capture this data, but doing so does require us to assume some non obvious (and perhaps unattractive) assumptions. Given the need in Georgian for a second cycle of agreement, Preminger's proposed `FIND` operation would need to reapply if it failed to find a value for the probe upon its initial search. Since assuming unvalued features can project higher in the structure, this does not seem like an unreasonable extension. In

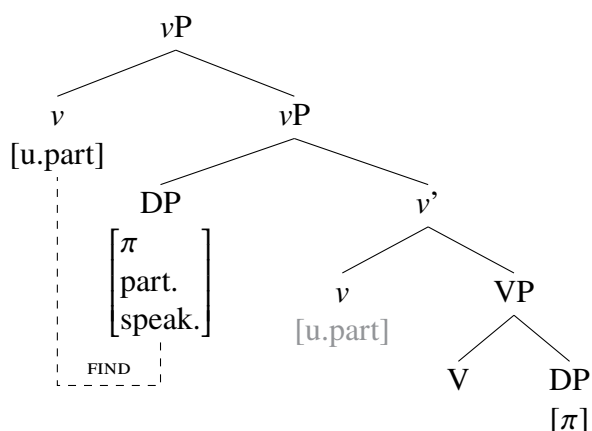
⁶This is a good time to mention that the featural specification of third person has a varied history and there are still many disagreements on how we should assume third person is modeled in the grammar. Some people are proponents of third person having true featural content, something like a $[\pi]$ feature, reflecting that third person is a default, but one that is relatively as opposed to totally underspecified (Ackema & Neeleman, 2017; Nevins, 2007). Others, assume that third person is totally underspecified, lacking person features at all, the reflection of total underspecification (Adger & Harbour, 2007). Still others treat each language individually and assume that the third person specification is something that is cross-linguistically variable (Béjar & Rezac, 2003). Why this is relevant here is that to maintain a distinction between second cycle agreement (partial default agreement) and total default agreement, one has to assume that third person is featurally empty. Otherwise, the exponence of third person is not a default as much as the reflection of third person features.

this view, *FIND* would be triggered automatically upon the merge of an unvalued probe on *v* and attempt to find a [participant] feature with which to agree. If it is unsuccessful in this search, as it is in (42a) when the object is third person, we could assume *FIND* is re-triggered when the unvalued feature projects higher in the structure and has the external argument in the larger search domain (42b). In this way, *FIND* would be capable of accounting for the kind of second cycle morphological effects seen in Georgian. If we did not allow the probe to re-trigger *FIND*, we wouldn't predict ungrammaticality as such a failure is tolerated by the grammar under the obligatory operations model, but we would incorrectly predict the appearance of default third person morphology when *FIND* fails in (42a).

(42) a. *FIND fails*



b. *FIND reapplies*



While it might be reasonable to assume that *FIND* is able to continue its search until spell out so long as the structural description is still met, it's important to point out that adopting this sort

of assumption under an obligatory operations approach does require us to explicitly stipulate it. It's also worth noting that Preminger does not consider this a possibility. When extending the `FIND` operation to non-AF data in Kichean, he laments `FIND` being unable to access the external argument if we located it on v , implying that he does not consider the failure of `FIND` able to continue searching. In a derivational time-bomb approach like the `MATCH/VALUE` approach, this additional assumption comes for free since the need to value an unvalued feature is already what drives the operation to apply in the first place. It's therefore quite obvious why second cycle agreement effects are possible.

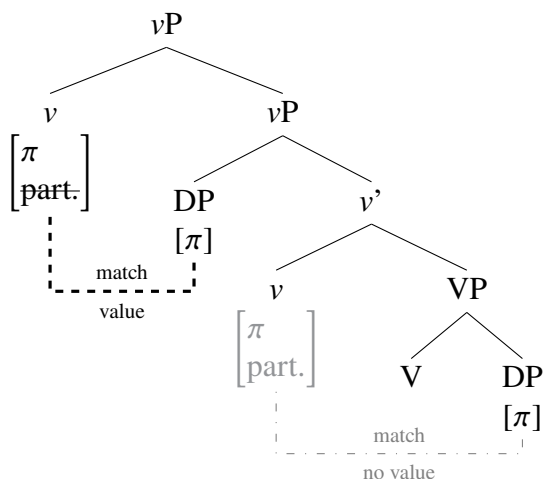
It is not just the stipulative nature about `FIND`'s reapplication that is problematic; more worrisome is that making this assumption goes against the spirit of the approach itself. This is a question of how we encode grammatical requirements and what we assume those grammatical requirements to be. The obligatory operations approach shifts those requirements to the application of a set of rules or operations. It is the attempt – or the triggering of the operation – that is required, not a successful (or any specific) outcome. Once an operation has been triggered, the grammatical requirement that it is intended to encode has been met. It is therefore not obvious why operations in this framework should need to reapply. Conversely, in a more standard derivational time-bombs approach, the grammatical requirements enforced by the grammar do depend on outcomes. It's therefore less radical to assume that the grammar dispatches everything at its disposal to satisfy them. The reapplication of operations – with an understanding of how that application is constrained – isn't as unexpected.

It is also interesting to note that there is one derivational distinction between the models, although we are unable to use it to empirically decide between the two. In a Georgian clause with two third person arguments, both models will be able to capture that the first cycle of agree is allowed to fail without causing the derivation to crash. The outcome of the second cycle however – while morphologically identical – is derivationally distinct. Under the `MATCH/VALUE` approach, upon the failure of the first attempt at agreement, the probe's featural specification is reduced to the root feature. This means that on the second cycle, a modified version of the initial probe is doing

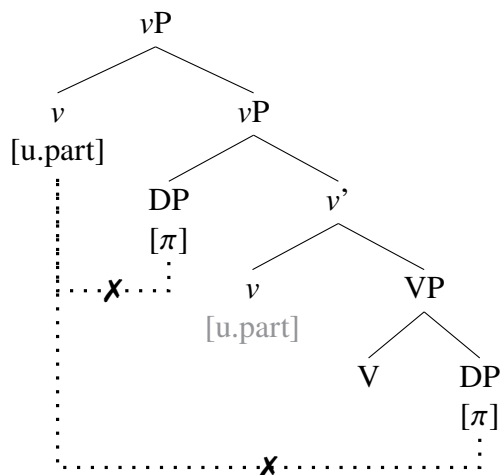
the searching, which in turn means that a different set of arguments now qualifies as a successful controller. To the extent that one assumes third person in Georgian has featural content, the newly reduced $[\pi]$ probe will in fact find a successful agreement controller in a third person argument, but only upon the second cycle of agreement (43). This means that the default $[\emptyset]$ form under this framework would be the exponent of successful agreement with a third person argument.

The picture looks a bit different from the obligatory operations view. Under this view, there is no means for probe modification. Once the `FIND` operation has been triggered and it fails to find a successful controller, the grammatical requirements of the operation have been met. While we can assume that `FIND` might be able to reapply as the result of such a failure, there's no mechanism that strips the probe of featural content upon the second cycle, a feature of the system that Preminger explicitly argues against. What this means is that upon the second cycle of agreement, the [participant] probe would not consider $[\pi]$ a successful controller and agreement would once again fail (44). Unlike the `MATCH/VALUE` approach, the `FIND` approach considers the default $[\emptyset]$ form the exponent of complete failed agreement. At this point, there's no clear way to distinguish these two alternatives empirically, but it's important to point out that they do make some theoretical distinctions. Also of note is the observation that if one understood third person features differently in Georgian, the distinction is removed. If third person is instead assumed to have zero featural content, then the `MATCH/VALUE` approach would fail similarly to the `FIND` approach on its second cycle of agreement.

(43) *Second Cycle 3rd Person Agreement*



(44) *Hypothetical Reapplication of FIND*



Georgian is not unique in morphologically expressing a distinction between agreement cycles. Another illustration comes from Karok, a language spoken in California (Bright, 1957). Karok, like Georgian is a low- φ language, meaning that its person and number probes are both located on the lower agreement functional head v . Karok similarly exhibits separate morphological affixes that are dependent on which cycle of agreement the probe was successfully valued. The paradigm is shown below in table 3.6. The Karok facts we'll discuss here are slightly more complicated because these affixes, unlike what we reported above involve both person and number. The series A morphemes are inserted in singular contexts, where the series B morphemes are inserted in plural

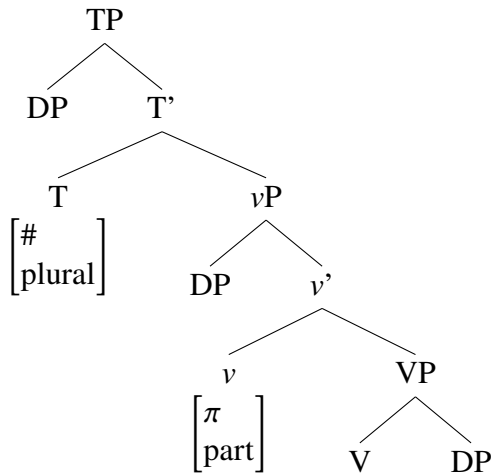
contexts. What's important for the point at hand is that there is a distinction between morphemes inserted on the first cycle of agreement and morphemes inserted on the second cycle of agreement. This, as it did for Georgian, signals that the operation behind φ -agreement must be able to reapply upon failure. Again, both models can account for these morphological patterns in similar ways by assuming that operations are able to continue to apply if they fail to find an agreement controller upon their first search.

Table 3.6: Karok Agreement Morphemes

		First Cycle		Second Cycle	
	A ₁		B ₁	A ₂	B ₂
1st	na-		kin-	ni-	nu
2nd	nu-/ʔi		ki-	ʔi-	ki-
1st		N/A		ʔu-	kun-
pl.		-ap			ka-

Another example comes from Erza Mordvinian a language spoken in Mordovia (Abondolo, 1982). Erza Mordvinian exhibits a slightly different system than the ones we've seen in Georgian and Karok. Erza Mordvinian is what's called a split- φ language, a language whose person probe and number probe are located on different heads. For Mordvinian, the person probe is the lower probe, located on v , and the number probe is located higher, on T (45). Like the other languages described in this chapter, Mordvinian has person hierarchy effects that are encoded through the use of relativized probes. Mordvinian's person probe is specified for [participant], while its number probe is specified for [plural], (45). Like Georgian and Karok, Mordvinian also exhibits second cycle agreement effects that are morphologically realized.

(45) *Split φ -Probes in Mordvinian*



The Mordvinian affix paradigms are shown below in table 3.7. There are two things to clarify in the first cycle affix paradigm. First, like we've seen above, there are no 3rd person first cycle affixes since given the relativized person probe, φ -agreement will never successfully provide a value upon the first cycle if the internal argument is a non-[participant]-bearing third person argument. Second, the alternation in the plural affixes is due to phonological alternations (Béjar, 2003). What's especially interesting about the Mordvinian paradigms is that not only are there morpheme sets for each cycle of agreement, but the morpheme structure is distinct across cycles as well. Mordvinian's second cycle affixes are suppletive for person and number, but its first cycle affixes are not. The second cycle affixes in the first column, **-a**, **-ak**, **-y** are the morphological exponence of second cycle agreement when both arguments are singular. The second cycle affixes in the second column, **-n**, **-t**, **-nze**, are what surface on the second cycle when the direct object is plural and the second cycle affixes in the third column, **-nek**, **--k**, **∅**, are the morphological exponence of second cycle agreement when the subject is plural.

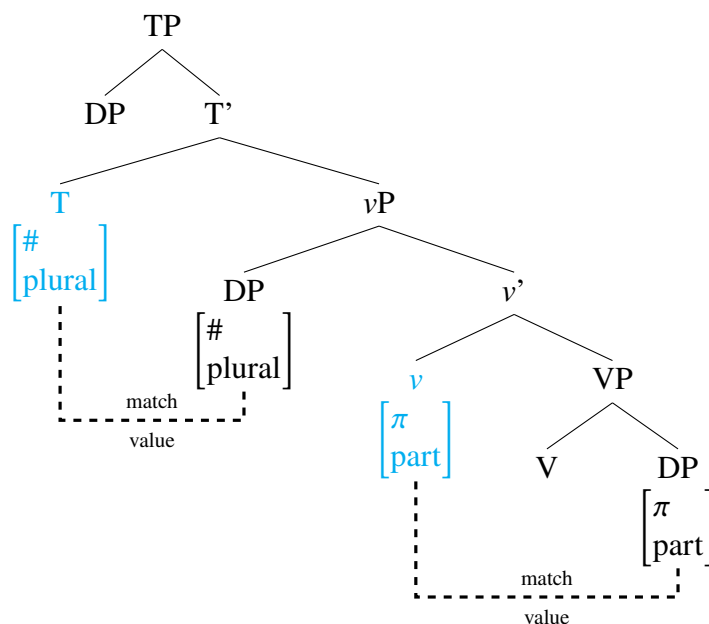
Table 3.7: Mordvinian Agreement Morphemes

	First Cycle	Second Cycle		
1st	-am	-a	-n	-nek
2nd	-ad	-ak	-t	--k
3rd	N/A	-y	-nze	∅
pl.	-yz/-iz/-y		∅	

Béjar (2003)'s MATCH/VALUE approach shows how the triggering of second cycle agreement derives this distinction. Central to this explanation is the assumption that upon a failure to agree, the unvalued feature projects to a higher position in the structure to allow for both the expansion of the search domain and the continued search in this new domain. Independent of this assumption, suppletive morphology is predicted to occur when the two heads that house the person and number features respectively are in an adjacent enough position to encourage the morphology to insert a suppletive form. To see how these two assumptions produce suppletive morphology when paired with the results of failed agreement, we'll work through each of the four-way outcomes of the interaction between the two φ -agreement probes: (i) both succeed, (ii) person succeeds and number fails, (iii) both fail, and (iv) person fails and number succeeds.

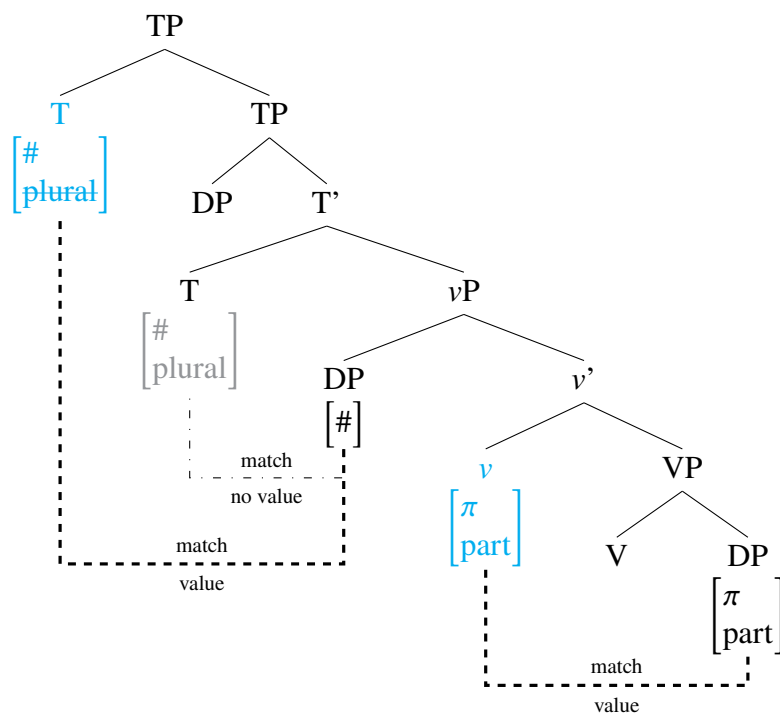
The simplest possibility is that no second cycle effects are triggered by the success of φ -agreement upon the first attempt. The success of φ -agreement will mean that the now valued probes – colored in blue – are each located in their original positions, far enough away from each other to prevent the insertion of any suppletive morphology (46).

(46) *Both Probes Successful on 1st Cycle*



If instead, VALUE with person features is successful, but VALUE with number features is not – thus triggering a second cycle for the number probe – the two heads that house the now valued probes are still far enough from each other to prevent suppletion in the morphological component. This is what we see in (47). When the number probe fails to VALUE on the first attempt, its feature specification is stripped to the root and it projects to reapply.

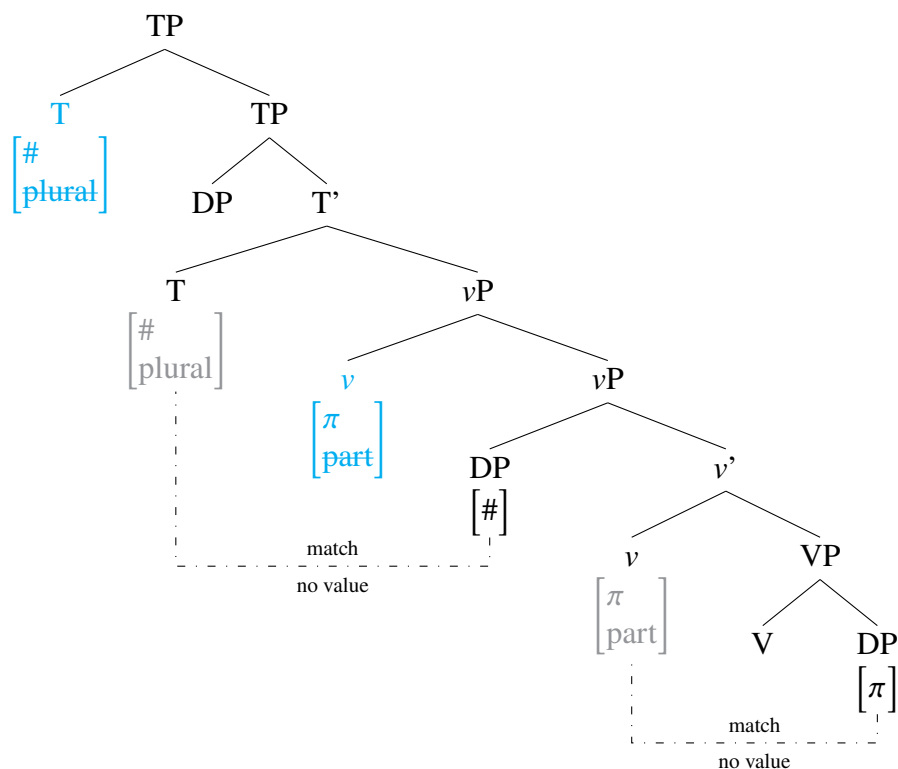
(47) *Person Successful on 1st Cycle, Number Fails*



A similar outcome is found when both person and number fail to agree on the first attempt (48). The person probe on v will have its features stripped and project. It will be in this position where the probe's features are valued either through second cycle agreement with the external argument, the internal argument, or default agreement.⁷ Likewise the number probe on T fails to value and it also is stripped and projected. Both features project higher upon the second cycle and are thus once again too far from each other to induce suppletion.

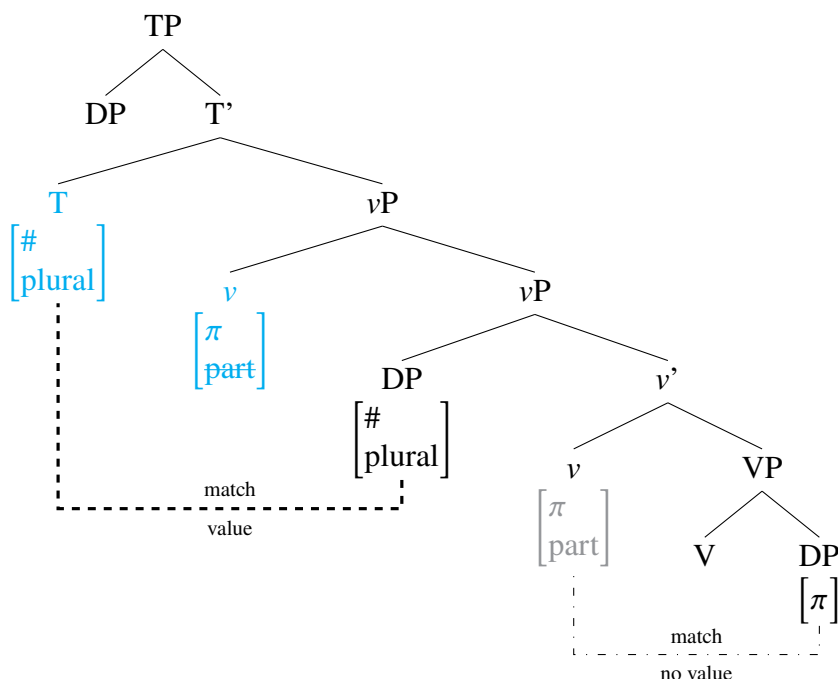
⁷Since the focus of the trees in this section is to show the final positions of the probes, I've remained agnostic here about which DP winds up valuing the person features.

(48) *Both Probes Fail 1st Cycle*



Notice, however what happens when person agreement fails on the first attempt, triggering a second cycle, but number agreement is successful on its first (49). The two valued probes are located adjacent to each other because person projected while number did not. Their adjacent positions can thus trigger the insertion of a suppletive form.

(49) *Person Fails 1st Cycle, Number Succeeds*



What allows the circumstance exhibited in (49) that produces the suppletive morphology is the availability of the higher projection of features indicative of a second cycle of agreement that is triggered by the failure of the operation to succeed on the first attempt. In this way, second cycle agreement effects not only result in distinct morpheme sets, but they also provide an explanation for the kind of morphological type distinction we observe in Ezra Mordvinian.

What the data in this section has shown is that the grammar makes use of a distinction between agreement on the first attempt and agreement on a later attempt. How the morphology uses this information to supply the correct morphemes is an independent question, but there are a few options that have been proposed. The first is to assume that the mechanism behind morphological insertion is sensitive to the features that inherently exist on the probe and the additional featural structure added by the operation *VALUE* Béjar (2003). A solution like this one is tied quite heavily to the assumption made in Béjar (2003) that the probe's feature structure is reduced to the root feature upon the exhaustion of the first cycle. This reduction allows for a distinction in the probe's starting feature set between the first and second cycles of agreement that could then be extended to the vocabulary items themselves. The vocabulary items for first person – first and second cycle

respectively – could be differentiated by what features were on the probe. The second proposal, and the one more amenable to the `FIND` operation, is to assume that the insertion of each of the second cycle affixes is conditioned by some syntactic context specified on the vocabulary item itself (Béjar, 2003; Béjar & Rezac, 2009). For example, we could differentiate the first person first cycle affix from the first person second cycle affix by including a structural description on the latter, something like “is inserted when in the domain of T”. Since the T head will have not yet merged into the structure at the time the first cycle affix has applied, we can use this context to differentiate between the structure that exists at the two cycles.

There are a number of other languages in which we observe similar morphological sensitivities (see Béjar and Rezac (2009) for a thorough survey). To recap, what these morphological sensitivities to cyclic agreement show is that the grammar has a way to recognize and express whether an unvalued probe received a value via establishing a successful agreement relation on its first attempt (first cycle agreement) or on its second attempt (second cycle agreement). At minimum, this data empirically requires that whatever model we adopt, the operations that are responsible for φ -agreement must be able to reapply until the relevant unvalued feature has received a value. The `MATCH/VALUE` approaches accounts for this data quite clearly, while the `FIND` approach would require a stipulation that is at odds with the spirit of the proposal.

3.4.1.2 Syntactic Effects

Not only is there evidence the morphology is sensitive to the distinction between first and second cycle agreement, but the syntax itself seems sensitive to this as well. This section illustrates two places where we see evidence of this: (i) the presence of additional morphology – and thus additional syntactic material – upon agreement on the first cycle in inverse agreement contexts and (ii) the presence of a special case upon agreement on the first cycle. Both of these are described as repair strategies to address deficiencies that arise when agreement is successful with the internal argument upon the first cycle. As with the morphological sensitivities, these syntactic sensitivities are discussed here only to show once again that the operation behind φ -agreement has more

complicated outcomes than simple success and failure and by extension requires a model capable of predicting these varied outcomes.

We first begin with a recognition of the morphological facts, then we'll proceed to discussion about what people have assumed those facts to reflect syntactically. In Mohawk, there is an additional prefix that appears only when agreement has successfully been achieved with the internal argument (Beatty, 1974; Béjar & Rezac, 2009; Postal, 1979). This prefix is in addition to the traditional agreement marker that reflect the φ -features of the agreement controller. In the paradigm shown below in (50), the canonical agreement marker is shown in small capitals, while the additional first cycle agreement prefix is shown underlined.

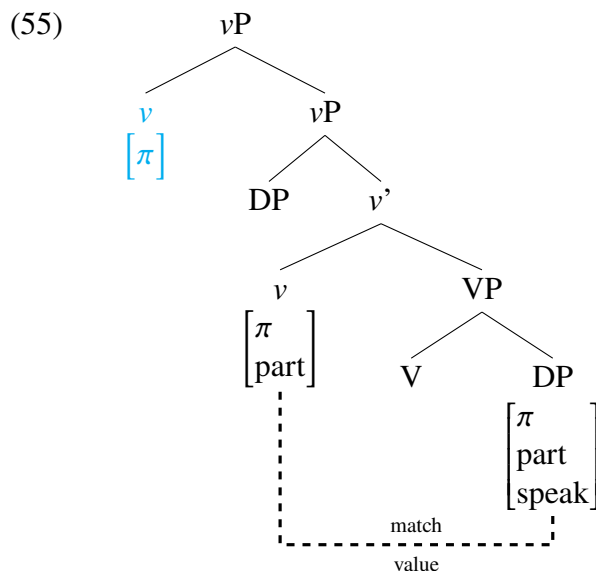
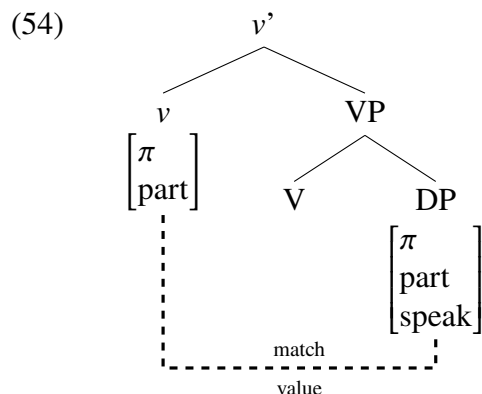
- (50)
- | | | |
|----|---|-------------------------------------|
| a. | <u>KU</u> -see
1/2-see
'I see you' | $1 \rightarrow 2$, <i>external</i> |
| b. | <u>K</u> -see
1-see
'I see him.' | $1 \rightarrow 3$, <i>external</i> |
| c. | <u>hs</u> -see
2-see
'You see him.' | $2 \rightarrow 3$, <i>external</i> |
| d. | <u>(h)s</u> -K-see
2-1-see
'You see me.' | $2 \rightarrow 1$, <i>internal</i> |
| e. | <u>wa</u> -K-see
3.INV-1-see
'He sees me.' | $3 \rightarrow 1$, <i>internal</i> |
| f. | <u>(H)s</u> -(<u>w</u>)a-see
2-3.INV-see
'He sees you.' | $3 \rightarrow 2$, <i>internal</i> |
| g. | hra- <u>wa</u> -see > <u>hra</u> -o-see
3.M.DFLT-see
'It sees him.' | $3 \rightarrow 3$, <i>internal</i> |

What's relevant for the discussion at hand is that this additional morpheme is only present in instances where the internal argument was successfully agreed with – on the first cycle of agreement. If the external argument instead controls agreement – the result of a second cycle of agreement – this additional morpheme is absent. In this way, Mohawk marks a distinction between the first and second cycles of agreement.

Another morphological fact that illustrates the same concept is the appearance of a special kind of case that Béjar and Rezac (2009) call *R-case*. An example comes from Kashmiri, a language spoken in India (Wali & Koul, 1997). In Kashmiri, there is a special case that is morphologically identical to the dative case that only appears when first cycle agreement is successfully established with the internal argument (51) (Mahajan, 1989; Nash, 1996; Woolford, 1997, 2006). This case, while morphologically identical to the dative (53), differs from canonical dative as it – unlike the canonical dative – disappears under passivization (52b). Once again, what's relevant for the current point is that there exists a case whose distribution depends on which cycle of agreement successfully established a relationship with an argument.

- (51) a. bɪ̃ chu-s-ath tsɪ̃ parɪna:va:n
 I.N be.M.SG-1.SG.N-2.SG.E/A you.N teaching
 'I am teaching you' *1 → 2, direct*
- b. tsɪ̃ chu-kh me parɪna:va:n
 you.N be.M.SG-2.SG.N me.D teaching
 'You are teaching me.' *2 → 1, inverse*
- (52) a. su kari-y tse me hava:lɪ̃
 he.N do.FUT-2.SG.D you.D me.D handover
 'He will hand you over to me.' *3 → 2, inverse*
- b. tsɪ̃ yi-kh me hava:lɪ̃ karnɪ̃ təm'sɪ̃ndi dəs'
 you.N come.FUT-2.SG.N me.D handover do.INF.ABL he.GEN by
 'You will be handed over to me by him.'
- (53) mohnas a:yi kəmi:z aslamni zəriyi dinɪ̃
 Mohan.M-D pass.F.SG shirt.F Aslam.M by give
 'Mohan was given the shirt by Aslam.'

What groups the Mohawk and Kashmiri examples together, but distinguishes them from the morphological effects we discussed in the last section, is the fact that these morphological effects are proposed to be the exponent of an additional φ -probe added to the derivation as the result of a successful agreement with the internal argument. They therefore constitute a syntactic, rather than a morphological, sensitivity to second cycle agreement effects (although of course they are still reflected in the morphology as well). Béjar and Rezac (2009) propose that once the probe finds a successful controller in the internal argument, it establishes an agreement relation (54). This agree relation is what allows the grammar to generate an additional probe (in blue) upon the projection of the v head (55) that can access the external argument.

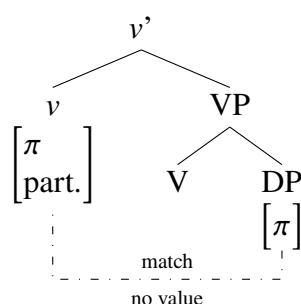


If the original probe instead fails to find a controller in the internal argument (56), the lack of an agree relation blocks the addition of this probe to the projection of v and the original probe continues to search in its new search domain, finding a controller in the external argument (57).⁸ Béjar and Rezac (2009) view this mechanism as a repair strategy intended to resolve PLC violations due to a failure of the external argument to establish an agreement relation Béjar and Rezac (2009). If each

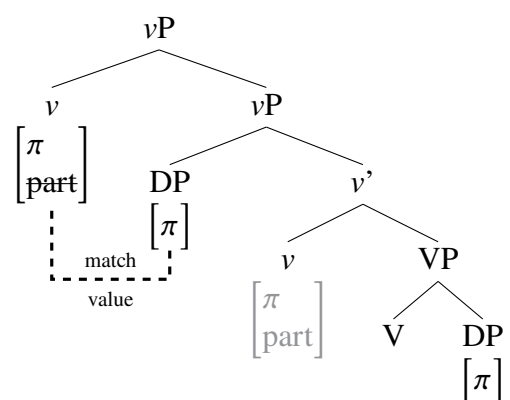
⁸Béjar and Rezac (2009) adopt the MATCH/VALUE approach, but suggest that each individual feature can agree independently. For an example like (56), they assume the $[\pi]$ feature on the probe checks the one on the internal argument and then the $[\text{part}]$ feature is residual and is what probes on the second cycle. These details aren't relevant for the points I'm making here, so I've left them off the trees. (see Béjar & Rezac, 2009, for details).

nominal must enter into some agreement relation to be licensed, then the success of the internal argument to establish such a relation bleeds the ability for the external argument to do. The added probe mechanism provides the external argument an opportunity to be licensed by agreement only in situations where it would otherwise be unable to do so. Mohawk and Kashmiri are understood to be beholden to the same principles related to the added probes, but are assumed to spell out those added probes differently. In Mohawk, the added probe is spelled out quite obviously as an additional morpheme, but in Kashmiri, the case assigning properties of v are modified by the presence of the additional probe on v .

(56)



(57)



One could imagine that Preminger too could account for these types of second cycle effects by proposing an additional operation, one whose application was triggered by the successful culmination of the operation FIND. For this to be possible, the operation would need to be able to do two things. First, it would of course need the ability to add the additional probe to the higher projection of v . I see no issue here, at least none which doesn't also plague the MATCH/VALUE style approach. The operation would also however need to be able to be formulated in such a way where the outcome of agreement could be accessed by the structural description. Preminger has proposed one such operation that we might use for guidance: a movement to subject position rule for non-quirky languages, shown in (58). He argues that movement to subject position is an operation independent of FIND, but one that depends on FIND successfully finding an agreement controller to be triggered. To encode this, the operation includes the outcome of an independent operation in its structural description.

(58) *In a non-quirky-subject language:*

$\text{MtoCSP}_{\text{NQS}} = \text{Move (XP successfully targeted by FIND)}$

What makes this problematic however, is that without derivational tracking or a model driven by the incessant need to value features – which as we’ve discussed Preminger is loathe to do – it seems that this reference must be constrained to rules that modify in some perceptible way the syntactic object they are targeting. Otherwise, it’s not obvious that the grammar can determine that an independent operation has or has not either been triggered or been successful. With respect to the movement to subject rule, we can see that the problem is that the FIND operation wouldn’t modify in any way the argument being targeted for φ -agreement, and subsequent movement. What is modified are the features of the probe, not the goal. It’s not clear how the grammar ‘knows’ whether or not a particular XP goal had been successfully targeted by FIND. With respect to our hypothetical rule, we must be able to propose some operation that can trigger the insertion of an additional probe upon the success of the FIND operation to succeed in valuing a probe’s features. Here, the situation actually appears a bit more optimistic than it does for the movement operation. At the success of FIND, the probe is modified in a way such that its unvalued features receive a feature value. If one proposed an operation that was triggered upon the existence of valued features on v , we might be able to account for the second cycle effects in Mohawk and Kashmiri.

(59) *Hypothetical added probe rule*

Inspect φ -features of v . If valued, insert an additional unvalued probe on projection of v .

This operation could add the same added probe that Béjar and Rezac (2009)’s system does in the case that FIND was successful. A tangential concern, and one that may affect both proposals equally, is the question of what motives or explains why the grammar has the added probe mechanism at all. Under the Béjar and Rezac (2009) approach, the addition of the added probe is one instance of a broader last resort mechanism, in this case employed to alleviate PLC violations. The FIND operation and its hypothetical added probe counterpart could likely invoke reference to the PLC violation as well, especially given its central role in accounting for the Kichean AF data. However,

while not especially problematic, the PLC does seem more amenable to being modeled in a system modeled by feature valuation rather than an obligatory operations approach. We'll discuss this further in section 3.6. In brief, the need for interpretable [participant] features to establish some sort of agreement relation in order to be licit appears more in sync with a model that enforces its principles through generally similar mechanisms rather than one that enforces its principles through the triggering of operations without concern for their outcomes.

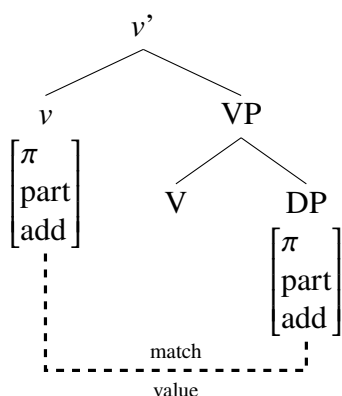
At a minimum, what these syntactic effects do is reinforce the idea that the grammar is sensitive to whether or not the φ -agreement probes are successful on the first or second attempt at agreement. This in turn reinforces the idea that the outcomes of agreement are not simply success or failure, but rather success, failure with interesting effects, or failure that leads to defaults. Whatever model of φ -agreement we adopt, it must be able to account for second cycle effects – and to the extent that we consider Béjar and Rezac (2009)'s treatment of the Mohawk and Kashmiri data reasonable – it must be able to trigger the insertion of an additional φ -probe to obviate violations of the PLC. Once again, these empirical requirements suggest that the outcomes of failure to agree aren't a simple binary set: success (agreement) and failure (default). Instead, the failure to agree results in more complicated outcomes for both the syntax and the morphology that reduce the tenability of the FIND approach.

3.4.1.3 Probe Modification

Finally, we come to the most problematic person hierarchy data for an operation like FIND: the agreement pattern found in Nishnaabemwin. Nishnaabemwin exhibits both person and number agreement (Valentine, 2001). Both the person and number probe are assumed to be located on ν and thus we can characterize Nishnaabemwin as a low φ -language Béjar (2003). There are two facts about Nishnaabemwin that differ from the other languages we've seen so far. First, the feature specification $[\pi[\text{part}]]$ maps to first person in Nishnaabemwin as second person is more specified in the language, adding an addressee feature to its feature set: $[\pi[\text{part}[\text{add}]]]$. Second, third person in Nishnaabemwin is assumed to not be specified at all, $[\emptyset]$.

Like all languages with person hierarchy effects, the choice of agreement controller is dependent not only on the syntactic characteristics of one argument, but of multiple arguments and on their relative positions. Nishnaabemwin prioritizes agreement with 2nd person arguments, then 1st person, then 3rd person. In a clause with a second person object and a non-2nd person subject (either 1st or 3rd person), the probe targets the object first and agrees as in (60).

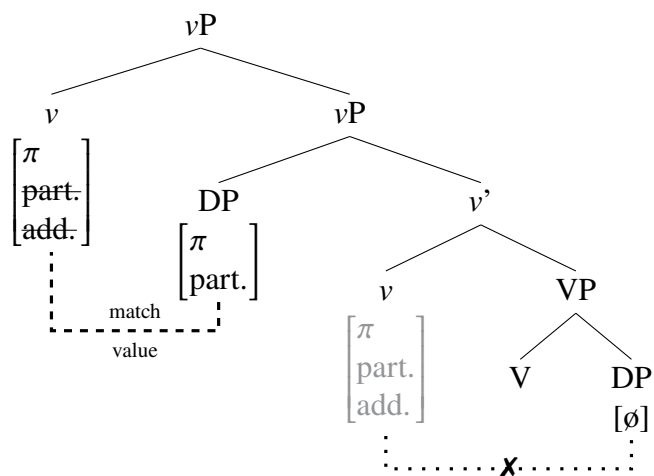
(60) *Agreement with 2nd Person Object*



Relevant to our discussion comparing the FIND operation and Béjar's MATCH/VALUE approach is what happens when the object is not 2nd person, and thus not a viable target for agreement. As (61) shows, the $[\pi[\text{part}[\text{add}]]]$ searches its domain and does not find a DP that can fully value its person feature. If the probe is unable to agree with the object due it being a non-second person argument, the agreement controller *displaces* to the external argument – if that argument is first person. If not, default third person morphology surfaces. What this tells us is that when agreement fails on the first attempt, the probe still cares about finding an agreement controller that respects Nishnaabemwin's person hierarchy of $2 > 1 > 3$; default morphology is not an immediate nor the singular result of failed agreement. This is viewed as evidence that there's a distinction between failed agreement that results in the agreement with something else upon a *second cycle* of agreement and failed agreement that results in the insertion of default morphology. As we already saw in the last section, Béjar (2003) accounts for this pattern through the stripping of the probe's features upon failing to value the internal argument - the outcome she terms *partial default agreement*. The stripping of the

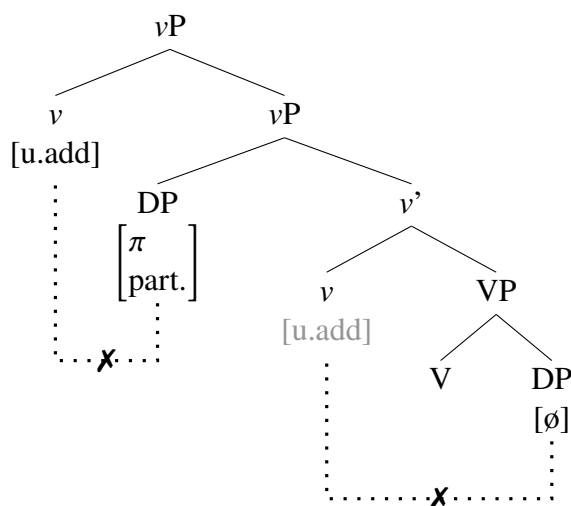
probe modifies the set of person features that would qualify as a successful agreement controller, thus making the way for first person second cycle agreement (61)

(61) *1st Person Agreement on Second Cycle*



What's crucial to accounting for the second cycle agreement patterns found in Nishnaabemwin is the ability to modify the feature specification on probes, a feature that the FIND obligatory operation approach does not share. If we assumed the obligatory operation FIND was triggered immediately upon the merge of the probe, it would attempt to agree with the internal argument and it would fail to do so (62). A integral feature of this proposal is that this failure to agree would not cause a derivation crash, as the operation is allowed to fail without consequences for grammaticality. From here, there are two potential next steps: either (i) the FIND operation is exhausted and can't reapply and the default third person form is wrongly inserted at spell out or (ii) FIND applies again upon a projection of probe features. In section 3.4.1, I presented reasons to be concerned about allowing FIND to reapply, but for the sake of pushing the account, let's assume that it can.

(62) *Reapplication of FIND*



The problem is that without access to a means for probe modification, the probe will still be relativized for second person upon any subsequent agreement cycles. Thus the first person external argument that is in the newly created second cycle agreement domain would still be unavailable for agreement because its feature set doesn't qualify it as a viable agreement controller. Agreement would then fail on this second cycle and at the spell out of v , the default third person features would again be wrongly inserted.

Notice that under either assumption about *FIND*— whether the probe halts its search after the first attempt or is allowed to continue its search through multiple attempts – the outcome is the failure of the probe to agree with an argument, resulting in the insertion of default third person features at spell out. So while second cycle agreement effects of course need the ability to probe a second cycle (which *FIND* can perhaps provide), they also depend on the grammar's ability to modify the probe in a way that broadens the type of argument that the probe could agree with successfully (which *FIND* cannot provide). So although the probe is initially relativized to only consider second person arguments as potential controllers, the probe is able to consider first person arguments as well, but only upon a second attempt. The *FIND* operation does not come with the ability to modify the probe upon failure and as a result predicts third person default features instead of the first person morphology we observe.

It's important to be clear that the pattern shown here reflects true agreement with a first person external argument and is not the result of a more general version of default features. In Nishnaabemwin, there is an empirical difference between first person agreement affixes and third person default agreement affixes, shown in table 3.8. The distinction between the first person affix and third person default affix, both possible as a result of failed agreement tells us that the outcome of failed agreement is not simple tolerance, as the `FIND` operation models. The probe must be able to change what it is looking for when it doesn't find an agreement controller on the first try. The result is either agreement on the second attempt (61) or no agreement at all.

Table 3.8: Nishnaabemwin Agreement Morphemes

Morpheme	
1st	n-
2nd	g-
3rd (default)	w-

Preminger is understandably critical of including special diacritics in the agreement system, especially those whose function appears redundant or solely to ensure that the last resort default mechanism does in fact wait until the last resort. The problem here is that it appears that Béjar's partial default agreement has been mischaracterized as such a diacritic. The fact that partial default agreement has an empirical consequence that differs from total default agreement challenges that characterization. Partial default agreement does more than just mark a time-bomb 'safe', preventing it from crashing the derivation, and its role goes beyond ensuring that the last resort mechanism defaults truly wait until all other options have been exhausted. True to its name, partial default agreement allows for a third outcome – an empirically necessary one – between canonical successful agreement and default agreement. Furthermore, the 'diacritic' itself doesn't share typical diacritic behavior in that it actually modifies the probe's featural specification. In this way, partial default agreement behaves less like a diacritic and more like an additional operation triggered upon the result of a previous one. Given this crucial role in accounting for more complicated agreement patterns that appear impossible to account for without it, I would challenge the idea that partial

default agreement is a redundant diacritic, if even a diacritic at all.

What the person hierarchy data has shown us is that the outcomes of failed agreement are more varied than the `FIND` operation itself can model. While it's tempting to think that partial default agreement is an unnecessary or redundant mechanism intended to ensure that a last resort mechanism did in fact only apply as a last resort, it importantly provides a third outcome of agreement – one that the grammar uses in varied ways.

3.4.2 Dative Intervention

An exciting extension of the obligatory operations approach to failed agreement is its potential to serve as an account for dative intervention. Dative intervention describes the puzzling fact that dative arguments are unable to transfer their own φ -features to a probe (63), but can serve to intervene and thus block agreement with a lower argument (64). What makes this phenomenon puzzling is the question of how to reconcile those two behaviors. On the one hand, in order to block agreement with a lower argument, the dative argument must be visible in some way to this probe. Traditionally, this means needing to have some set of φ -features. Without these features, it's unclear how the probe would be able to “see” and subsequently be halted by the dative argument. On the other hand, if the dative argument does in fact have the φ -features that make it visible to the φ -probe, it's unclear why the dative argument is unable to transfer those feature values to the probe via agreement.

- (63) a. Strákunum leiddist/*leiddust.
boy.thePL.DAT were.bored.3.SG/*3PL
'The boys were bored.'
- b. Strákarnir leiddust/*leiddist.
boy.thePL.NOM walked.hand.in.hand.3PL/*3SG.
'The boys walked hand in hand.'
- (Sigurðsson, 1996)
- (64) Það finnst/*finnast einhverjum stúdent tölvurnar
EXPL find.SG./*PL. some student.SG.DAT computer.the.PL.NOM ugly
'Some student finds the computers ugly.'
- (Holmberg & Hróarsdóttir, 2003)

Once we concede the ability of the agreement operation to fail, we can account for the two-faced nature quite easily (Preminger, 2014). To do so, Preminger first cites an assumption that φ -agreement is sensitive to the assignment of morphological case (Bobaljik, 2008), with each language adhering to the Moravcsik hierarchy, shown in (65). The way to interpret this hierarchy is to say that if a language permits agreement with dependent case marked arguments, it will also permit agreement with unmarked case arguments, but not agreement with lexically marked or oblique arguments. Each language is able to set its own relevant boundary, which accounts for some of the cross-linguistic variation we see in which types of arguments are viable agreement controllers in different languages. The extension of obligatory operations to dative intervention also depends on this assumption in that the FIND operation responsible for φ -agreement is sensitive to case distinctions. Preminger modifies the operation's description to be sensitive to this case discrimination, (66).

(65) *Moravcsik Hierarchy*

unmarked case > dependent case > lexical/oblique case

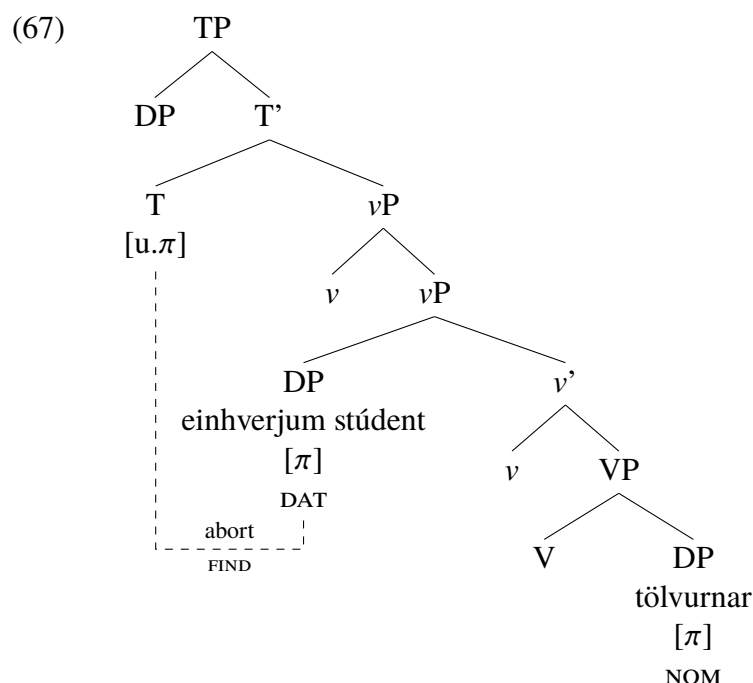
(66) *FIND(f)*

Given an unvalued feature f on a head H^0 , look for an XP bearing a valued instance of f . Upon finding such an XP, check whether its case is acceptable with respect to case discrimination:

- a. *yes* $\rightarrow$ assign the value of f found on XP to H^0
- b. *no* $\rightarrow$ abort FIND

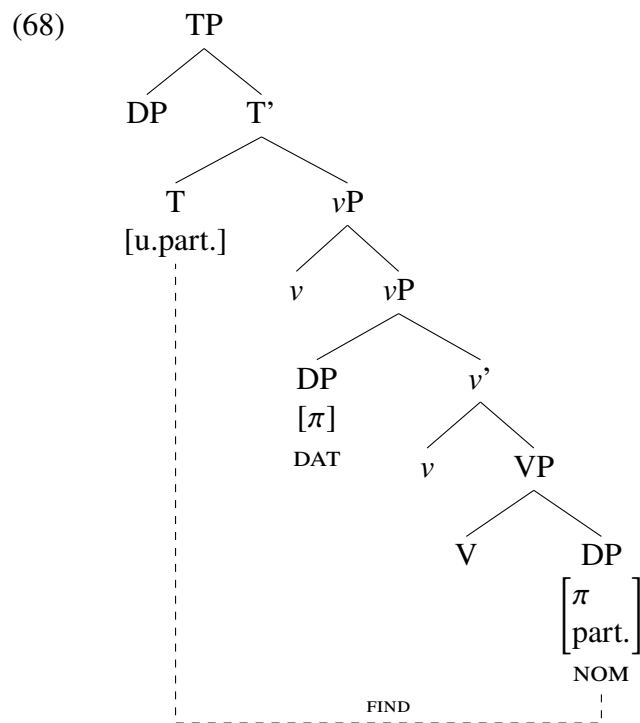
Dative intervention, under this approach, is the result of failed agreement, explaining why we observe default third person in exactly these instances. Observe the rough sketch in (67) for the Icelandic sentence in (64). Icelandic person probes aren't relativized for a particular set of person features, so Preminger assumes the probe is $[\pi]$ and on T. When merged into the structure, FIND is immediately and obligatorily triggered and begins its search for an argument with which to agree. According to the description outlined in (66), it finds such an argument in the dative argument

einhverjum stúdent. The probe then evaluates the acceptability of the dative argument's case, according to where on the hierarchy Icelandic sets the parameter for case discrimination – between dependent and unmarked case. Therefore according to Icelandic's case discrimination settings, dative case is not acceptable for agreement and the FIND operation aborts, thus ending the probe's search. Because operations are allowed to fail without grammatical consequence in the obligatory operations model, the derivation does not crash and when this phase is spelled out, the third person default features are correctly inserted.



The question then turns to how this account of dative intervention would fare when coupled with the fact that not all languages with dative intervention are assumed to have flat $[\pi]$ probes, as Icelandic and French do. In other words, does this account of dative intervention still work in languages whose agreement probes are relativized to search for a particular set of person features, like those sensitive to person hierarchy effects discussed in the previous section? To see why this is an important question, let's observe a prediction that this dative intervention model makes. There are two important features of this account. The first is that the dative argument must be visible to the probe in order to trigger the evaluation for case discrimination. The second is that evaluating

that an argument does not meet the requirements dictated by case discrimination must immediately halt the probe. Otherwise, the probe would be able to continue its search and incorrectly target a lower nondative argument. With relativized probes, a smaller set of feature specifications make an argument visible. This in turn raises the question: what happens when the dative argument is less specified than the probe (and a lower nondative argument), as shown in (68)? According to the modified FIND operation, the probe will begin its search, but being relativized for [participant], for example, would simply ignore a third person dative argument, as it does in all of the canonical failed agreement data discussed for Kichean AF constructions outlined in previous sections. Because the dative argument is ignored, it is never evaluated for case discrimination and therefore never causes the FIND operation to abort. If a lower nondative argument existed, nothing would prevent the probe from finding this argument and establishing a successful agreement relation.



There is data from Georgian that appears to be exactly the kind described above and thus constitutes a data set that the FIND approach to dative intervention cannot capture (Harris, 1981). In (69a), there is both a third person dative argument and a third person nominative argument.

Agreement morphology is third person and is consistent with an approach like the one we've described above where failed agreement would trigger the insertion of third person default features. The FIND proposal does not run into any issues here because even if the probe is unable to 'see' the dative argument, it would likewise be unable to 'see' the third person nominative, resulting in failed agreement and predicting the appearance of third person features. However, data like (69b) cause a problem. Since the probe is relativized in Georgian to search for a [participant] feature, it won't ever be able to investigate whether or not the dative argument's case features respect case discrimination for the language because third person arguments do not bear this feature. Without being able to halt the probe, nothing would prevent the probe from successfully agreeing with the [participant]-bearing nominative argument, which as (69b) shows is ungrammatical. What is missing here is the ability for a third person dative argument to be visible to a more specified relativized probe within a system that otherwise depends on the relativized probe ignoring less specified arguments.

- (69) a. vanom anzori šeacara givis
 Vano-ERG Anzor-NOM he-compares-him-him Givi-DAT
 'Vano compared Anzor to Givi.'
- b. *vanom (šen) šegadara givis.
 Vano-ERG you-NOM he-compared-him-you Givi-DAT
 'Vano compared you to Givi.'
- (Harris, 1981)

An approach like Béjar's MATCH/VALUE approach does not share this problem because the probe evaluates MATCH at the root feature level, ignoring any further featural structure that may exist. This means that third person arguments aren't ignored, but instead mark the probe for partial default agreement, if relevant. The relativized [participant] probe is able to search as it always does and considers the dative third person argument a successful match. From here, if we assumed that a MATCH/VALUE approach to dative intervention uses the same sensitivity to case discrimination that the FIND approach does and we assumed that case discrimination would similarly halt the operation, we could account for dative intervention in the same way that Preminger (2014) does.

At its core, the dative intervention puzzle is one of visibility. When a probe is specified quite

minimally, as it is in Icelandic and French, it's hard to see how any problems could arise because any dative argument will be visible enough to the probe to instigate case discrimination inspection. Of course, as is the theme of this section, failed agreement patterns aren't always this simple and the additional complexity that person hierarchy sensitive languages add raises some issues with respect to the interaction between relativized probes and dative intervention. The `FIND` operation has difficulty in capturing this additional complexity, while the `MATCH/VALUE` approach has the mechanisms available to handle it.

3.4.3 Conjunct Agreement

A final example of an instance where failure to value does not immediately trigger a default comes from analyses of conjunct agreement. Here we'll see that for some languages, a failure to value can result in different mechanisms for calculating conjunct agreement patterns, either resolved agreement (RA) or closest conjunct agreement (CCA). What's important to take away here is the existence of an outcome of the failure to value that is distinct from the insertion of a default form. This signals that the grammar must have some way of distinguishing when to use default mechanisms as the result of failed value and when to use other mechanisms (like CCA) as the result of failed value. The answer to this question still eludes us, but it's reasonable to attempt to propose a syntactic distinction that the morphology or PF can read in a way that would help condition which outcome we observe. To the extent that we can find a syntactic distinction, the `MATCH/VALUE` approach seems able to provide that distinction in a way that a `FIND` approach that only encodes triggering cannot. Derivational time-bombs care about outcomes, obligatory operations care about beginning states. Failed agreement is a non-uniform set of outcomes, needing a model that cares about outcomes more than it cares about beginnings.

Bhatt and Walkow (2013) outline data from Hindi-Urdu that shows a distinction between how conjoined arguments interact with agreement that is dependent on whether those conjoined arguments are in subject position or in object position. When a conjoined argument is in subject position, we get something called *resolved agreement*, or agreement with the entire conjoined

phrase. This is shown below in (70), where number agreement is easiest to illustrate. In the examples in (70), number agreement tracks the entire phrase, not one of the conjuncts. So even when neither argument is plural, we still observe plural agreement when the conjoined argument is in subject position because the agreement probe is accessing the features of the entire ConjP. This of course implies that the grammar has some mechanism available for calculating the resolved features of a conjoined phrase. The details of how this calculation occurs is not relevant here, so I direct you to (Bhatt & Walkow, 2013; Marušič, Nevins, & Badecker, 2015) for more discussion

- (70) a. Ram aur Ramesh gaa [rahe hãĩ / *rahaa hai]
 Ram.M and Ramesh.M sing [PROG.M.PL be.PRS.PL / *PROG.MSG be.PRS.SG]
 ‘Ram and Ramesh are singing.’ M.SG + M.SG: agreement = M.PL
- b. Sita aur Ramesh gaa [rahe hãĩ / *rahaa hai]
 Sita.F and Ramesh.M sing [PROG.M.PL be.PRS.PL / *PROG.MSG be.PRS.SG]
 ‘Sita and Ramesh are singing.’ F.SG + M.SG: agreement = M.PL
- c. Ram aur Sita gaa [rahe hãĩ / *rahii hai]
 Ram.M and Sita.F sing [PROG.M.PL be.PRS.PL / *PROG.F be.PRS.SG]
 ‘Ram and Sita are singing.’ M.SG + F.SG: agreement = M.PL
- d. Mona aur Sita gaa [[rahii / rahe] hãĩ/ *rahaa hai]
 mona.F and sita.F sing [[PROG.F / PROG.M.PL] be.PRS.PL / *PROG.M.SG be.PRS.SG]
 ‘Ram and Sita are singing.’ F.SG + F.SG: agreement = F.PL/M.PL

(Bhatt & Walkow, 2013)

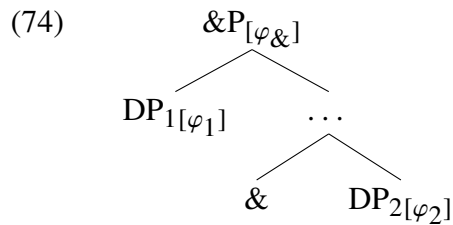
The agreement pattern differs however when conjoined phrases are in object position instead. When a conjoined phrase is in object position, resolved agreement is completely unavailable (73) and instead the probe agrees with the closest conjunct (71)-(72).

- (71) a. Ram-ne ek thailii aur ek badsaa (aaj) uthaa [-yaa/ *-yii/ ??-ye]
 Ram.ERG a bag.F and a box.M (today) lift [-PFV.M.SG/ *-PFV.F/ ??-PFV.M.PL]
 ‘Ram lifted a small bag and a box (today).’ [F.SG + M.SG] ... V.PART.M.SG
- b. Ram-ne kai thailiyã: aur ek badsaa (aaj) uthaa [-yaa/*-yii/ ??-ye]
 ram.ERG many bag.F and a box.M (today) lift [PFV.M.SG/ *PFV.F/ ??PFV.M.PL]
 ‘Ram lifted many small bags and a box (today).’ [F.PL + M.SG] ... V.PART.M.SG

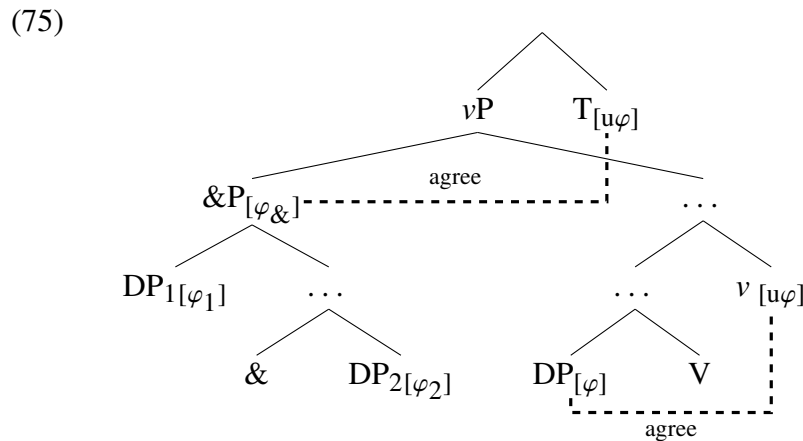
- c. Ram-ne ek thailaa aur ek baksaa (aaj) uthaa [-yaa/ ??-ye]
 Ram.ERG a bag.M and a box.M (today) lift [PFV.M.SG/ ??PFV.M.PL]
 ‘Ram lifted a bag and a box (today).’ [M.SG + M.SG] ... V.PART.M.SG
- d. Ram-ne kai thaile aur ek baksaa (aaj) uthaa [-yaa/ ??-ye]
 Ram.ERG many bags.M and a box.M (today) lift [PFV.M.SG/ ??PFV.M.PL]
 ‘Ram lifted many bags and a box (today).’ [M.PL + M.SG] ... V.PART.M.SG
- (72) a. Ram-ne ek thailii aur ek petii (aaj) [uthaa-yii thii / ??uthaa-yii
 Ram.ERG a bag.F and a box.F (today) [lift-PFV.F be.PST.F.SG / ??lift-PFV.F
 thĩ: / ??uthaa-ye the]
 be.PST.F.PL / ??lift-PFV.M.PL be.PST.M.PL]
 ‘Ram had lifted a small bag and a box (today).’ [F.SG + F.SG] ... V.PART.F AUX[F.SG]
- b. Ram-ne kai thailiyā: aur ek petii (aaj) [uthaa-yii thii / ??uthaa-yii
 Ram.ERG many bags.F and a box.F (today) [lift-PFV.F be.PST.F.SG / ??lift-PFV.F
 thĩ:/ ??uthaa-ye the]
 be.PST.F.PL / ?? lift-PFV.M.PL be.M.PL]
 ‘Ram had lifted many bags and a box (today).’ [F.PL + F.SG] ... V.PART.F AUX[F.SG]
- c. Ram-ne ek thailaa aur ek petii (aaj) [uthaa-yii thii / ??uthaa-yii
 Ram.ERG a bag.M and a box.F (today) [lift-PFV.F be.PST.F.SG / ??lift-PFV.F
 thĩ:/ ?? uthaa-ye the]
 be.PST.F.PL / ??lift-PFV.M.PL be.PST.M.PL]
 ‘Ram had lifted a bag and a box (today).’ [M.SG + F.SG] ... V.PART.F AUX[F.SG]
- d. Ram-ne kai thaile aur ek petii (aaj) [uthaa-yii thii / ??uthaa-yii
 Ram.ERG many bags.M and a box.F (today) [lift-PFV.F be.PST.F.SG / ??lift-PFV.F
 thĩ: / ??uthaa-ye the]
 be.PST.F.PL / ??lift-PFV.M.PL be.PST.M.PL]
 ‘Ram had lifted many bags and a box (today).’ [M.PL + F.SG] ... V.PART.F AUX[F.SG]
- (73) Ram-ne ek phaalvaalii aur ek duudhvaalii [dekhii thii/ ??dekhii thĩ:/ *dekha tha/ *dekhe
 the]
 Ram.ERG a fruit.seller.F and a milk.seller.F.SG [see-PFV.F be.PST.F.SG/ see-PFV.F be.PST.F.PL/
 *see-PFV.M.SG be.PST.M.SG/ *see.PFV.M.PL be.PST.M.PL]
 ‘Ram had seen a fruit seller and a milk seller.’ (Bhatt & Walkow, 2013)

To account for the distinction in behavior between conjoined subjects and objects, Bhatt and Walkow (2013) propose an analyses of conjunct agreement that derives the distinction through accessibility. Conjoined nominals are assumed to have the structure shown in (74), where each individual conjunct

has its own φ -features and the &P is the locus of resolved features, the φ -features that represent the entire conjoined nominal. Since the φ -features of the &P are higher – and therefore closer to the probe – than the φ -features of the individual conjuncts, the probe will interact with them first. This results in resolved agreement being the ‘typical’ outcome, only not obtaining when blocked in some way. Since Hindi-Urdu has one φ -agreement probe, we can view the situation as being another instance of agreement competition, this time between the φ -features on the &P and the φ -features of the individual conjuncts.

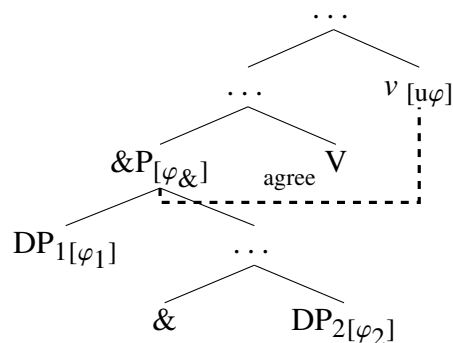


When the conjoined argument is in subject position, the φ -agreement probe on T will first encounter φ -features of the &P (75). Since there is no evidence that Hindi-Urdu obeys the kind of person hierarchy effects observed in previous sections, we can safely assume the probe on T is a flat probe, specified as $[u\varphi]$. Therefore, there is nothing to block agreement with the φ -features on the &P, resulting in resolved agreement.

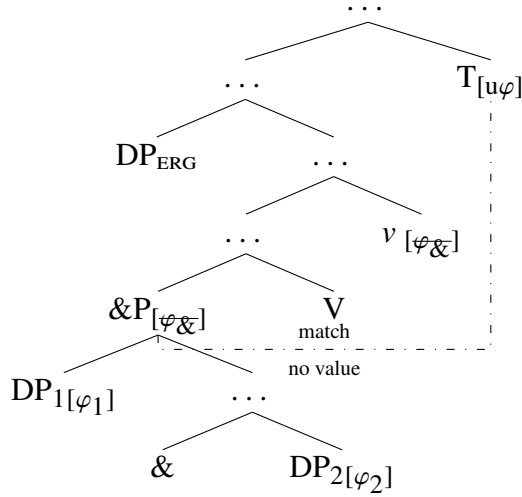


When the conjoined argument is instead in object position, the situation is a bit different. Bhatt and Walkow (2013) assume that when v merges into the structure, it assigns case to and agrees with the coordinated object (76a), rendering its φ -features inaccessible for φ -agreement via the activity condition (Chomsky, 2001). This has the effect of making the φ -features on &P unavailable to value any future probes, but importantly still accessible to matching. When the probe on T reaches the &P object, it cannot agree with the resolved features on the &P itself (76b), explaining why resolved agreement is impossible when the conjoined argument is in object position, (73). Bhatt and Walkow (2013) then propose that a match with the φ -features on &P, but a failure to value the features on the probe will trigger a morphosyntactic algorithm that decides which of the two conjuncts will value the probe: this will result in either first conjunct agreement or last conjunct agreement.

(76) a. Step 1:



b. Step 2:



Once again, we see an instance where the outcomes of failed agreement are more complicated than the insertion of a default. In the Hindi-Urdu conjunct agreement data, we see that the failure to value can result in not a default, but rather in the calculation of closest conjunct agreement. While neither of the two agreement models we've discussed are obviously extendable to this data without modification, the MATCH/VALUE approach certainly appears more amenable since it already has a third outcome of agreement built in via the partial default mechanism. We can imagine that we could modify this approach to include the morphosyntactic algorithm as an additional outcome of a probe succeeding in matching with a goal, but failing to be valued by it. One of the research questions that Bhatt and Walkow (2013) leave for future research is how we distinguish between default agreement as a result of failure to value and closest conjunct agreement as a result of failure to value. In the MATCH/VALUE approach, we can imagine the answer. Closest conjunct agreement is the result of a successful match with a failure to value, while default agreement is the result of a total failure to agree.

Modifying the FIND approach to make this distinction between failed valuation outcomes is more difficult. Inherent to its conceptual basis is that the grammar's processes are driven by the need to trigger operations, not by the need to ensure any particular result. In this system, there is no division between matching and valuing and because FIND is a single operation; it has a binary set

of outcomes: it either succeeds or it fails. It is hard to imagine how to distinguish between when failure to agree triggers a default and when failure to agree triggers a different result.

3.4.4 Interim Summary

The common thread that ties the data in this section together is that the failure to value a set of φ -features leads to a more complex set of outcomes beyond the insertion of a default. The solution that we need to propose therefore needs to be able to encode this nonbinary set of outcomes of failed agreement. The person hierarchy data showed both the need for a second cycle of agreement and the need for probe modification. The dative intervention data illustrated a need to be able to distinguish φ -features on two levels: their value and the category to which they belong. Finally, conjunct agreement data showed that when valuation fails, the outcome is not the immediate insertion of a default, but rather that the grammar can use the failed valuation to trigger a host of other outcomes. We saw how `FIND` was unable to capture some of these more complicated agreement phenomena and how the availability of a `MATCH/VALUE` approach provided a solution, despite cited criticisms of partial default agreement.

At this juncture, one may concede that the details of the `FIND` operation are unable to capture the more complicated types of failed agreement illustrated in this section, but still be concerned that the crux of Preminger's argument has not been rebutted. One can reasonably ask whether we could take the details of the Béjar `MATCH/VALUE` approach that do account for these patterns and combine them with the obligatory operations impetus that drives derivations. I will spend the next two sections providing arguments against the conceptual impetus behind obligatory operations to argue that a model that encodes grammatical requirements in the standard way, via the need to value features, should be preferred. The relevant implication for the broader discussion in this chapter is that obligatory operations, which care only about whether or not an operation has been triggered, and do not encode any sort of grammatical requirement in the outcome of the operation are ill-suited to handle this non-binary set of failure outcomes.

3.5 The Premature Overapplication of Defaults

Fallible operations move the impetus of various derivation operations away from their outcomes towards the contexts in which they are triggered. However, their ability to fail introduces a few timing issues because if operations are allowed to fail without grammatical consequence, then we need to ensure that they do not fail too early. To begin, we ask: what does it really mean to say that operations are obligatory? If the goal here is to better account for why φ -agreement as a phenomenon is obligatory, then of course we should hope to discover a better answer than ‘ φ -agreement is obligatory because the operation responsible for it is obligatory’. What is missing from this hypothetical response is an explanation for what exactly is *responsible* for the operation’s inherent obligatory nature; why must the operation apply? Without this understanding, we reduce inherent obligatoriness to stipulation. Preminger of course sympathizes with this need for explanation and he provides some insight towards more satisfying answers.

To encode obligatoriness in a substantive, non-stipulative way, we need two properties: automation and the immediacy it implies. The basic `FIND` operation provides a nice illustration of why these properties are necessary. Take once again, Kichean AF agreement; all that’s needed to account for why φ -agreement must happen is to say that once the operation’s structural description is met, the operation immediately and automatically is triggered. For `FIND`, this means that upon the merger of an unvalued feature f on a head H , the operation proceeds. If that unvalued feature f finds a successful match, then the operation’s result will be the transfer of φ -features from goal to probe. If it does not find a successful match, then the operation can fail without consequence for grammaticality and third person singular default features will surface. Importantly, what allows us to adopt this explanation for the operation’s obligatory nature without stipulation is that it automatically applies once its structural description has been met.

Inherent to this account are two concepts that are at odds with one another: the need for immediacy and the need for delay until the creation of the relevant structural description. While there’s a tension between the two, they are certainly not incompatible; the basic `FIND` operation is a great illustration of this. However, because their natures are in constant tension, there at least exists

the possibility they can interact in problematic ways.

Where we don't see problematic interaction is where the structural description for a particular operation is quite simply the presence of a particular syntactic object. In these instances, of which basic *FIND* – repeated below in (77) – is an example, merging a single syntactic object creates the structural description and in this way the two conflicting natures are easily reconciled in one step. Upon merge of an unvalued feature *f* on a functional head, the probe may begin probing.⁹ Here we have no real tension between the automatic triggering of the operation and any need for delay.

(77) *FIND*(*f*)

Given an unvalued feature *f* on a head H^0 , look for an XP bearing a valued instance of *f* and assign that value to H^0 .

Where we might see problematic interaction between the need for immediacy and the need for delay is when either the structural descriptions for operations are more complicated or when the rule that is triggered relies on the application of an independent operation. Preminger's extension of the *FIND* operation as an account of dative intervention provides an illustration of this point. Preminger follows Bobaljik (2008) in arguing that, due to φ -agreement's sensitivity to case discrimination, the application of φ -agreement must follow the valuation of case features. This sort of delay is exactly the kind of situation that proves problematic for encoding obligatoriness in the automatic triggering of operations. Essentially, what case discrimination means for the timing of φ -agreement is that *FIND* must wait not only until its structural description has been met, but also until the case features of the relevant arguments are assigned before it can proceed. This gap between the creation of the structural description and when it needs to be triggered is probably widest if one assumes, as Preminger does, a dependent case model of case assignment.

Relevant to our current discussion is the fact that dependent case is a configurational model of case valuation which means that in order to assign case features, all relevant competitors must be present in the derivation before their case features can receive their respective values. If φ -

⁹Whether or not the probe can continue probing was discussed in a prior section. While possible, it would be done via stipulation and would be at odds with the spirit of what drives operations.

agreement is dependent on the valuation of these case features, then *FIND* needs to wait not only until the structural description below has been met upon the merger of an unvalued feature *f*, but it also needs to wait until all arguments are merged into the structure and have been assigned case. Otherwise, it could be triggered too early and fail without consequence. The result would be an overapplication of default morphology in instances where we should observe true agreement.

We can see a similar tension in the operations proposed to handle object shift. Object shift describes a phenomena that involves the optional movement of a DP object out the VP it initially occupies (Diesing & Jelinek, 1993; Fox & Pesetsky, 2005; Holmberg, 1986, 1999). The following data is from Icelandic (Thráinsson, 2007). What is obligatory is that if object shift occurs, then a specific interpretation is required (78a) and if object shift does not occur, a specific interpretation is impossible (78b). If however, the reason that the object has not shifted outside the VP is because object shift is blocked or otherwise unavailable, then a specific interpretation for the object DP is still licit (79).

- (78) a. Ég las₁ [þrjár bækur]₂ aldrei [VP t₁ t₁].
 I read(PAST) three books never
 ‘There are three books that I never read’
 (✓ specific reading of ‘three books’, ✗ nonspecific reading)
- b. Ég las₁ aldrei [VP t₁ þrjár bækur].
 I read(PAST) never three books.
 ‘I never read three books.’
 (✓ nonspecific reading of ‘three books’, ? specific reading)
- (79) a. *þau hafa [viðtöl við Blair]₂ alltaf [VP sýnt t₂] klukkan ellefu
 they have interviews with Blair always shown clock eleven
- b. þau hafa alltaf [VP sýnt [viðtöl við Blair]] klukkan ellefu.
 they have always shown interviews with Blair clock eleven.
 ‘They have always shown interviews with Blair at 11 o’clock.’

(Thráinsson, 2007)

To account for this, Preminger proposes the operation shown in (80). Notice that this rule is sensitive to language-specific structural conditions. For example, in Icelandic object shift, this

condition is that the verb must have moved out of the VP. Notice that the structural description for this rule is the existence of an X that is [+specific], but the point at which the operation needs to apply to avoid premature failure is dependent on other syntactic concerns. To account for Icelandic object shift, the operation must be sensitive to whether or not a V has been moved out of its VP. In order to encode obligatoriness in the immediate and automatic triggering of operations, the operation actually needs to be sensitive to much more than the structural description in order to prevent overapplication of defaults (or failure). In this way, it seems that we cannot enforce the grammatical requirements *solely* through obligatory operations. Once we admit this, it's not clear to what degree obligatory operations as a framework is more attractive than the derivational time-bombs approach.

(80) *An obligatory operations model of OS*

$X[+specific] \rightarrow \text{Shift}[X]$

where Shift is the operation that causes a noun phrase to vacate the VP, and is subject to language-particular structural conditions on its successful culmination.

In pursuit of fairness, my intention here is not to pick on the details of the proposed object shift operation, especially because Preminger does not offer it as a significant proposal, but rather a mere illustration that other phenomena share the same logic as φ -agreement. Their shared logic is that if a rule can apply, it must, but the conditions are such that if a rule is unable to apply, the requirement is lifted. What I do intend to show however, is that while other phenomena might be amenable to an obligatory operations approach in logic, we must be especially careful in how we formulate the rules responsible and that some phenomena may not actually be as amenable as it may appear at first blush due to a dependence on other operations successfully applying first.

If all of these operations essentially need to wait until much more structure has been built, then it is worth wondering at what point they are actually triggered. A principled answer to this question could be that they are triggered upon spell out, at the phase level. But we know from our discussion of second cycle effects that – at least for languages that show preference towards

internal arguments (low- φ languages) – the probe must begin its search before the addition of the external argument. What we seem to have to conclude is that in order to derive the obligatoriness of operations beyond stipulative notions, we have to rely on either automation or immediacy, but relying on these introduces a number of timing issues and/or rule formation issues.

On the other hand, a derivational time-bombs approach – at least when coupled with an intermediate ability to fail like what's encoded in the MATCH/VALUE approach – can avoid some of these problems because it doesn't rely so heavily on notions of automation or immediacy. The derivational time-bombs approach derives its obligatory nature from a response to interface conditions, making the exact timing of a varied set of operations less crucial to capturing syntactic phenomena. If we take the MATCH/VALUE approach to dative intervention, for example, there's no inherent problem with φ -agreement waiting until case valuation because the operation behind φ -agreement isn't driven by notions of obligatoriness. What drives the operation in that framework is the need for features to be valued. Waiting until case is assigned is not at odds with the operation's motivation as it is in the obligatory operations framework.

As we discussed in the introduction to this thesis, the overapplication of defaults is just as problematic to the framework as getting the forms to surface in the first place (and in my view is actually the more theoretically interesting piece.) The overapplication puzzle really centers on the sorts of timing issues discussed here: how do we prevent the default mechanism from applying too early? What derivational time-bombs seem to get us is a way to slow down, or otherwise constrain, the unfettered application of syntactic operations. Preminger frames this as redundant, but I think both the timing issues illustrated in this section and the non-binary outcomes of failed agreement in the previous section show that unfettered application causes real problems and that derivational time-bombs are not redundant, but rather perform important moderating functions with respect to derivational timing.

3.6 Some Conceptual Issues

Now that we've discussed the empirical implications of adopting an obligatory operations approach, it's worth considering its conceptual implications. There are two issues in particular that are especially worth discussing: (i) the extendability of the approach and (ii) how it models grammatical requirements. I argue that if we cannot adopt the obligatory operations approach framework-wide, then there isn't much benefit to adopting it in such a small domain of the syntax, especially with the existence of a derivational time-bomb compatible alternative, like the MATCH/VALUE approach.

3.6.1 Framework-wide adoption

There are a few extensions of this framework beyond the narrow scope of Agree in domains that share a similar logic to φ -agreement. Preminger (2014) characterizes these phenomena as being similar to φ -agreement in that each is obligatory if the phenomenon is possible, but that if the conditions aren't such that the operation can apply, there's no consequence for grammaticality. I'll briefly review both the phenomena and their respective obligatory operations logic below.

First is the suggestion that we model object shift with an obligatory operation that is triggered immediately when the structural conditions for it are met. See the previous section for details. What Preminger wants to capture is the idea that if the conditions for object shift are impossible, the typically obligatory covariation between specificity and movement is lifted. In other words, the specific reading of shifted objects is obligatory as is the nonspecific reading of an unshifted object. However, if the conditions for object shift are not present and the reason the object stayed in its original position was because it was prevented from doing so, then both interpretations are possible. This mimics the logic of φ -agreement, as characterized by Preminger: if φ -agreement is possible, it's obligatory, but if there's a situation where φ -agreement is impossible, the obligatory requirement is lifted.

Similarly, Preminger offers an obligatory operation to handle the definiteness effect. The definiteness effect is a phenomena which typically bars definite arguments from staying in situ.

This asymmetrically affects definite, rather than indefinite arguments (81). Like φ -agreement and object shift, if the conditions are such that the definite argument cannot move to subject position, it is allowed to stay in situ (82). Once again, if the context is unavailable, the requirement is lifted. Preminger proposes an operation that is triggered obligatorily to account for this data (83).

- (81) a. **The boy/A boy** seems to be playing in the garden.
 b. There seems to be **a boy/*the boy** playing in the garden.
- (82) a. The boy/A boy seems to **the girls** to be playing in the garden.
 b. There seems to **the girls** to be a boy playing in the garden.
- (83) *An obligatory operations model of the DE*
- a. $X[+definite] \rightarrow \text{MtoCSP}[x]$ (universal)
 b. $X[\text{Ext. Arg.}] \rightarrow \text{MtoCSP}[x]$ (parameterized)

Finally, Preminger suggest one last extension, long-distance wh-movement. The issue Preminger raises for long-distant wh-movement is that if we model it as being the result of an unvalued [wh] feature attracting and triggering movement of a valued [wh]-bearing XP, we are forced into proposing two versions of the non interrogative complementizer: a [wh] bearing one that would attract the wh-phrase in (84a) and one that does not bear this feature to handle (84b). The need for two versions is due to in part to the derivational time-bomb nature of unvalued features. If instead there was an unvalued [wh] in (84b), it would remain unvalued throughout the derivation as there's not a [wh]-bearing XP for it to attract.

- (84) a. What did Mary say [t [_C that] John wanted what]?
 b. Mary said [[_C that] John wanted an armadillo].

Preminger instead proposes an obligatory operation *displace wh* that is shown in (85). Upon the merge of any complementizer, the operation is triggered to displace a c-commanding wh-bearing XP. If the clause contains such XP, as in (84a), the wh-phrase will move. If the clause however does not contain a wh-bearing XP, the operation will fail without grammatical consequence.

(85) *An obligatory operations approach to wh-movement*

$C \rightarrow \text{Displace}(\text{wh})$

Preminger tempers his argument with these extensions, offering them merely as suggestions that show an extension to other domains is at least possible, rather arguing they constitute a more attractive alternative. As I briefly mentioned in section 3.5, we do have to be careful about what sorts of rules we are comfortable with proposing. In order to maintain that obligatoriness is derived and not stipulated, the operations proposed in the system must be able to immediately apply upon the creation of the structural description. Some rules, like the definiteness operation or long-distance wh-movement, arguably seem better suited to this goal. Others however, like the object shift rule or the revised FIND operation to handle dative intervention, are far more difficult and need to be reframed in a way that their obligatoriness truly is enforceable by automatic and immediate triggering. As discussed in section 3.5, these latter rules suffer from a timing issue that is a result of a gap between the creation of its structural description and independent constraints on its application. Furthermore, as we saw with the second cycle agreement data, we must also be careful to consider the potential outcome of operation failure, as it is not often the case that the grammar simply tolerates failure in a simple way. At a minimum, given these concerns, extension is much more problematic than we are led to believe. If the existence of tolerated failure necessitated an obligatory operations approach, we might be more willing to tolerate these concerns; but with the availability of a MATCH/VALUE approach that also handles grammatical failures, I think the concerns become more serious. To the extent that these considerations constitute a counterargument, I offer them here.

Where we might find more convincing counterarguments are in extensions to phenomena that more canonically cause ungrammaticality through failure to value features. Three in particular are potentially problematic for a framework-wide extension of obligatory operations: the EPP, case licensing, and the PLC discussed in the Kichean AF data overview. The EPP is standardly accounted for through the proposal of a strong unvalued D feature that triggers the movement of the highest argument to the subject position or triggers the insertion of an expletive. The need

for the subject position to be filled is a strict requirement and is one that is modeled well under a derivational time-bomb style approach; a derivation crashes if this D feature remains unvalued either through movement of a DP or through the insertion of a DP expletive.

Another grammatical requirement that is especially amenable to filter-like systems and therefore something that would be quite difficult to handle in a framework where failure is widely tolerated is case theory. Standardly, nominals are assumed to need syntactic licensing and the assignment of case values to nominals is what can perform this licensing function. If a nominal fails to receive case throughout the course of the derivation, as it does in (86), the derivation crashes.

(86) *It is likely her to leave the party early.

Finally the PLC, repeated below in (87), offers another filter-like phenomenon that would be especially difficult to capture with an operation that was allowed to fail.

(87) *Person Licensing Condition (Béjar & Rezac, 2003)*

Interpretable 1st/2nd person features must be licensed by entering into an Agree relation with an appropriate functional category.

Preminger speculates that we might move what's responsible for these filter-like phenomena to a more amenable grammatical component where derivation crashing isn't relevant – removing them from the syntax (see Preminger, 2014, for more details). I raise the issue here as a reminder of the scope of the grammar and the models we have under consideration. The fact that problematic phenomena are addressed by moving the requirement they enforce out of the syntax and into a different component of the grammar is quite telling. It reinforces that each truly provides great difficulty for an obligatory operations approach. Any evidence that shows that these phenomena belong rightly in the syntax proper introduces a huge problem for framework wide adoption. We've seen that there are some operations whose obligatoriness is difficult to enforce, like object shift and case discriminating *FIND* and other phenomena that appear categorically incompatible with obligatory operations themselves and we therefore should be quite pessimistic about its extension,

especially in light of an alternative and more attractive way to handle failed agreement data.

A final related point, and one that I won't be able to fully address here, is a consideration of other syntactic phenomena that do not fit under the φ -agreement umbrella as narrowly defined in Preminger (2014), but nonetheless are standardly treated as being accounted for through the AGREE operation. In order to motivate more strongly the existence of agreement failures, Preminger restricts what he considers agreement to a very narrow understanding of morphological co-occurrence of features. He recognizes that in modern frameworks however, what's considered an outcome of a more general AGREE operation is significantly less narrow. AGREE as an operation has been used to account for negative concord (Zeijlstra, 2004), noun-modifier concord (Baker, 2008b), modal concord (Zeijlstra, 2008), and the binding theory (Reuland, 2011), among many other things. The availability for us to extend FIND to these other agreement-like phenomenon isn't addressed explicitly, but it is an important question to consider. Either we must be able to extend FIND to those other phenomena – which means we should find evidence that they are allowed to fail – or we cannot extend FIND. Being unable to extend FIND has an unattractive outcome in that we lose the ability both to treat these phenomenon as a set and to reduce the number of mechanics we propose to account for them. With the existence of a MATCH/VALUE approach that can also account for failed agreement, it becomes less clear what the advantages are of adopting the obligatory operations approach and more clear what we risk to lose. Furthermore, the timing issues raised in the previous section signal that it is possible that uninterpretable features may serve a timing-regulating function. If uninterpretable features are needed in the grammar more generally, once again we ask what we gain and what we lose by removing their role in a very narrow set of circumstances.

3.6.2 What are probes?

One conceptual result of adopting an obligatory operations framework is that it makes it harder to understand why syntactic objects with unvalued features are the things that probe. It's quite standard to assume that the defining characteristic of probes is their unvalued nature (Carstens, 2016). The motivation for the probe's search is an intrinsic need to get a value. The reason that

probes are, by definition, things with unvalued features is because the state of having an unvalued feature is generally (with some limitations) not tolerated by the grammar. This is not necessarily true in a framework that encodes requirements through obligatory triggering. The concern is that if we're not careful, what defines a probe under obligatory operations is more stipulative than it is under an approach that enforces grammatical requirements through the valuation of unvalued features. The need to receive a value is not only explicitly denied, but is also largely dependent on proposing a rule that would require it.

Features are what signal that a syntactic object is eligible to establish a relationship, they are the mechanism through which that relationship is established, and the transfer of their values is what signals that relationship to other components of the grammar. What does it mean for us to assume that despite this central role, they aren't what enforce the requirements of the grammar?

What the data in this chapter has shown is that the grammar cares very much about unvalued syntactic features receiving feature values. The grammar appears to have at least two different ways to receive a feature value: either through establishing a syntactic dependency with a valued feature bearing object or through failing to do so and receiving one via some default mechanism that supplies features as a last resort. The ability for φ -agreement operations to keep continuing to apply cyclically until a value is found encourages a characterization that this need for valuation is quite strong. This is more or less expected under an approach that frames grammatical requirements as being largely imposed by interface conditions, as is true of the standard derivational time-bombs approach. It is less obviously expected under an approach that says grammatical requirements are encoded component-internal, through an obligatory triggering of operations responsible for valuing features. The grammatical requirement `FIND` encodes is importantly not that an agreement probe receive a value, it's that the `FIND` operation is triggered when its structural description is met. The grammatical requirement – what the grammar cares about – is the application of the rule, not the valuing of the feature.

One of two things is likely true under such an approach: either (i) there are a number of obligatory operations, some of which happen to value features as an outcome or (ii) all operations

involve this need to value features and they all apply obligatorily. If the first is true, then it would mean that the fact that there's a strong need for feature valuation is merely a consequence of the particular rules that have been proposed. Feature valuation only appears to have such a central role by coincidence; we would not have a deeper understanding of why feature valuation is so central to producing grammatical sentences. Our generalizations would also be more vulnerable to the particular rules proposed by various people who work on these phenomena. If the second alternative were true, that all operations happen to involve an unvalued feature, then we would similarly miss a generalization that all operations share the same motivation without an explanation for what is behind that motivation. If we instead maintain the standard approach, that operations are driven by the need to value features, we directly encode the “correct” grammatical requirement, the one that seems empirically motivated and we provide an explanation of sorts for why it is that the syntax provides these great many avenues for feature valuation – it's required by an interface condition, a very minimalist assumption.

3.7 Conclusions

In this chapter I showed that there is an alternative to obligatory operations that allows us to capture how defaults are produced in the grammar while maintaining the assumption that uninterpretable features can still induce derivation crashes. This shows that the obligatory operations model is not *necessitated* by the existence of failed agreement, but is rather one option. We looked at a wide range of failed agreement data that showed the outcomes of AGREE are far from the simple binary distinction between success and failure. The FIND approach is ill equipped to handle this more complicated set of outcomes as it predicts only two. I also introduced arguments that claimed that obligatory operations raise a number of timing issues that, if we're not careful, will overgenerate the distribution of defaults. Finally, we looked at what it means to model grammatical requirements on the basis of operation triggering rather than feature valuation.

I suggest that what failed agreement shows is not that feature valuation is not a requirement of the grammar, but rather the opposite: that feature valuation is such a strong requirement

that the grammar has multiple ways of providing those features to unvalued syntactic objects. The existence of a default mechanism and of second cycle agreement effects strengthens this characterization. The way forward, then, is not to eliminate the role of derivational time-bombs in enforcing obligatoriness of grammatical requirements, but rather seek to better understand this default mechanism, importantly while maintaining the general framework assumptions. The MATCH/VALUE approach does exactly this and therefore should be the framework upon which we build an understanding of the default mechanism that supplies unvalued features in a narrow set of circumstances.

CHAPTER 4

AGREE-BASED CASE

4.1 Introduction

With an understanding of why the dependent case and obligatory operations approach to defaults are problematic, I'd like to turn our focus now towards a solution to the defaults problem that is more in line with standard assumptions. This chapter has two primary goals. The first is to show that default case can in fact be accounted for without rejecting case's role in regulating nominal licensing and doesn't require a dependent case theory of case valuation. This will be achieved by extending the MATCH/VALUE approach of φ -agreement to the domain of case assignment and will also involve understanding case features in a novel way. The second goal of this chapter will be to argue that in light of the serious theoretical concerns about the nature of the framework that adopting the proposals in chapter 2 and chapter 3 require, this new proposal offers a more attractive way to model how defaults interact with the basic tenets of the syntactic framework and should thus be preferred over the alternatives.

4.1.1 Revisiting the Problem of Default Case

As a reminder, a syntactic default in the arena of case is especially interesting because Case has historically had a central role in regulating the distribution of DPs. We do not expect something which can rule out derivations to have access to a default because that access by definition undermines any requirements that are encoded. According to traditional assumptions, the failure to value a Case feature will cause a derivation to crash. What rules out a sentence like (1) therefore is that the DP *her* is unable to value its unvalued Case feature because non-finite T does not have Case features to assign.

- (1) *It is likely **her** to leave the party early.

However, we've seen that there are a number of structures which like (1) also contain a DP that has failed to receive a Case value, but *unlike* (1) produce a perfectly grammatical sentence. A few of these are repeated below in (2):

(2) **Default Case in English:**

- a. *Hanging Topic/Left-Dislocation*
What?! **Him** wear a tuxedo?!
- b. *Gapping*
She will eat cake, **him** brownies.
- c. *Coordination*
Me and **him** will go to the store.
- d. *Modified Pronouns*
Lucky **me** has to clean all the toilets.

The two crucial properties that the environments in (2) share are: (i) they each lack a case assigner that could be the source of the accusative features on the bolded DPs and (ii) the morphological case that surfaces in every one of these instances is consistent within a language, but it varies cross-linguistically. Essentially, cross-linguistic default case data is indicative of a default case mechanism that is able to explain how different morphological cases systematically appear in the same positions cross-linguistically.

This data raises three important questions about both the nature and the role of case in the grammar and also about the nature of defaults and how they can be included in a system that rules derivations out when requirements are not met.

- (i) If we understand default case to be the failure to receive a case value, how can these forms surface in a system that rules such failures impossible?
- (ii) How does such a system distinguish between when it is acceptable to not get a case value and when it is unacceptable?

- (iii) What can an understanding of default case tell us about the features responsible for the distribution and pronunciation of DPs?

Essentially, (i) and (ii) are intended to explain the distinction we see in (1) and (2): how are the examples in (2) grammatical in the first place and once we have a solution, how do we prevent that solution from in turn applying to instances like (1). Question (iii) addresses the hope that once we have a better understanding of the default mechanism itself, we might be able to improve our understanding of how both abstract Case and morphological case features should be modeled.

We saw in chapter 2 that what some researchers have done to address these issues is remove the licensing offense itself, adopting an approach that removes Case's role in ruling out derivations and proposing a model of case valuation that builds the appearance of defaults directly into the system. These proposals address issue (i) by arguing that failing to value a case feature is not fatal to the derivation. They address issue (ii) by framing default and unmarked case as the last resort type of case features that are assigned by the grammar when the grammar is unable to assign one of the dependent cases. Through the discussion in that chapter, I raised some concerns with adopting both a configurational case system and eliminating case's licensing role. In this section, I will outline what others, while trying to maintain more closely the larger set of standard assumptions regarding case and licensing, have proposed for default case.

4.1.2 Previous Agree-based Approaches

As we've discussed at length, one of the first problems that the existence of a default case poses for our model of the grammar is how default case is able to surface at all, given that the failure to value a Case feature is presumed to be fatal to the derivation. If one wants to maintain the standard function of case, we must figure out a way to reconcile how that can happen, despite the grammar appearing to disallow it. Because the locus of the crash is the remaining unvalued Case feature itself, one way to solve this problem is to remove the feature entirely. By removing the feature, one removes the source of the crash. The logic of this approach is essentially: you can't fail to value something that isn't there. This approach centers on differentiating two types of DPs – DPs that

surface with default case forms and those that don't – and encoding this difference in the featural specification of each DP type. DPs that can surface with the default case form will be generated without any Case features and therefore without the ability to cause a derivation crash, at least due to Case. DPs that do not surface with default case are generated with the traditional Case feature set and must therefore receive a Case value or the derivation will be ruled ungrammatical.

Even among the analyses that propose a solution in this vein, we find some variety in how this removal is implemented. We could try to identify some property that unifies the set of either type of DP as Legate (2008) does. She couches the identifying property in a DP's merge position. On her account, DPs that are going to merge in an argument position are subject to a licensing requirement and are thus generated with the expected unvalued Case feature that enforces this requirement. DPs that won't be merged into an argument position are instead not generated with an unvalued Case feature and can thus survive to PF regardless of whether or not they agreed with any prototypical Case assigner.

This approach is successful in a few nice ways: (i) it maintains the standard set of assumptions regarding both the roles and the relationship between morphological case and licensing and therefore inherits the benefits of doing so, and (ii) it appeals to the elsewhere condition discussed in chapter 1 to insert a default form, essentially aligning default case with other instances of morphological defaults more generally.

Despite these successes, the way Legate implements this approach makes a few wrong empirical predictions. Since Legate identifies that the property that distinguishes whether or not a DP is generated with Case features is dependent on whether or not it merges in argument position, we predict default DPs to not appear in argument positions. The data in (3) shows two DPs (in bold) that are arguments of the gapped verb *drink* and the tenseless verb *wear*, respectively. While it's true that neither of these verbs has the ability to canonically assign accusative, it is extremely unlikely that the DPs here are not arguments of their verbs. This data therefore constitutes a counterargument to Legate's proposal.

- (3) a. We can't drink champagne and **him** dollar store wine.
 b. What?! **Him** wear a tuxedo!? Never!

This problem is largely due to the identification of some property that the generation of Case features is grounded in. If we were to remove that property and more or less randomly generate unvalued morphological case features¹ on DPs, as Schütze (2001) does, we could avoid the two issues outlined above. We do however run into a different problem if we pursue that option – overgeneration. Imagine a derivation like that in (4). If DPs are generated randomly in the numeration either with morphological case features [DP_[ucase]] or without [DP], then without an explicit understanding of what governs their selection, it appears that the grammar is equally able to select either DP type. If the derivation selects the cased version, (5a), all goes as expected: the DP is unable to receive any morphological case value from non-finite T in the embedded clause, but is able to receive nominative features when it moves to the SPEC TP of the matrix clause. At spell-out, this produces the sentence in (5b).

- (4) []_i is likely []_i to win the race.

- (5) a. [DP_[ucase]]_i is likely t_i to win the race. (6) a. [DP]_i is likely t_i to win the race.
 b. *Her_i is likely t_i to win the race.
 b. She_i is likely t_i to win the race.

If instead the derivation in (4) selects the caseless version, as in (6a), the DP once again is unable to receive a morphological case value from non-finite T. When this DP type moves to the SPEC TP position of the matrix clause, the presence of nominative features is irrelevant because there is nothing to “receive” the feature values. Without an unvalued morphological case feature, the DP surfaces exactly as it was generated. Because the resulting sentence (6b) is ungrammatical, this proposal overgenerates default case forms in positions where they are unattested. It's important to take a minute to be clear that Schütze's account does not overgenerate the actual distribution

¹It's important to clarify that unlike Legate, Schütze does not address licensing which is why I've switched to morphological features here.

of DPs more generally because Schütze assumes that a separate set of features are responsible for DP licensing. For him, what would rule out (7a) would presumably be that whatever licensing feature that is responsible for DP distribution and is generated on all DPs was not satisfied. By maintaining a strong separation between the features responsible for licensing and those responsible for morphological case, he models default case a purely morphological phenomenon.

- (7) a. Jane hopes [DP] to eat all the honey.
b. *Jane hopes **she/her** to eat all the honey.

It is entirely possible that we might be able to suggest modifications to either of these proposals that will avoid the issues that they are confronted with. However, I argue that despite these hypothetical modifications, we actually have a bigger conceptual issue here that supports abandoning this type of approach altogether, regardless of whether or not we could get the details to work. Case isn't an individual property; it's the reflection of a relationship. Whether or not a DP is a default case DP is about how it is *integrated* into a particular structure, not about individual properties of the DP itself. By modeling the distinction between DPs that end up with default case and those that end up with traditional case through a variation in feature specification on DPs, we put the locus of that distinction on the DP itself, rather than on the relationship that that DP and a particular functional head share. That distinction, I think, should be located on differences in the environments that produce defaults, rather than on the DPs themselves.

These arguments, coupled with the arguments against dependent case and the separation of case from licensing, support the proposal of another way to address the default case issues. What we're looking for then, is an AGREE-based system (contra dependent case theory) that is capable of both producing and constraining default forms that models the distinction between default forms and others as the reflection of environmental distinctions, rather than DP focused ones (contra previous AGREE-based approaches). In the next section, I'm going to argue that we can do just that by extending the decomposition of AGREE that is sensitive to the hierarchical relationships between the features relevant for case and licensing.

4.2 A New Approach

Before moving on to the details of the alternative approach I would like to propose here, I offer a quick overview of its basic components. I argue that we:

- (i) adopt a decomposition of AGREE into two independent operations: MATCH and VALUE. (Béjar, 2003; Béjar & Rezac, 2009; Rezac, 2011)
- (ii) propose that all DPs enter the derivation with identical morphological case and licensing requirements
- (iii) propose that the features responsible for morphological case and those responsible for licensing are independent, but related via entailment
- (iv) use the featural specifications allowed by this new relationship to expand the number of possible outcomes of AGREE, producing exactly the right circumstances to allow defaults to surface where (and only where) they are attested.

This section will detail both a novel understanding of case features and how those features behave in a system that assumes the kind of hierarchical sensitivity that Béjar's MATCH/VALUE approach requires.

4.2.1 Case Feature Systems

The MATCH/VALUE approach to defaults works well in the φ -agreement domain in large part because of the inherent feature structure that φ -features exhibit. It therefore bears asking: what other feature systems contain these hierarchical relationships and if Béjar's approach to AGREE is correct, how would those operations act upon them? In this section, I'm going to follow others who've come before me to argue that case features are similarly organized, with hierarchical internal structure. We'll examine conclusions from the morphological literature that supports this claim and I'll propose a novel system of case features intended to reflect the intuitions that have come from

literature on case syncretism patterns. The expectation is that with a hierarchically organized system of case features, the MATCH/VALUE theory of φ -agreement can be extended and can account for default case while being able to maintain case's role in nominal licensing.

4.2.1.1 Preliminary Concerns

First, we need to address some preliminary concerns about our understanding of the features responsible for licensing and morphological case. Although case is one of the most discussed domains in the syntactic literature, the features behind the relevant phenomena are some of the least understood. The field has still yet to arrive at a consensus regarding an accepted system of case features. This section will outline both what a theory of case features should look like and the issues that make it quite difficult to propose one.

In a very general way, any proposal of any syntactic phenomenon within a framework that depends on the valuation of features to enforce the syntactic requirements that one assumes are active or relevant for a phenomenon really needs to take seriously how those features are motivated, structured, and organized. This is not about simply outlining assumptions in a way that makes a MATCH/VALUE-style extension possible, but rather about a more general goal of understanding the features that play such a crucial role in this type of syntactic framework. Our current syntactic model uses features as the actual mechanism by which all of these syntactic processes operate and all grammatical requirements are enforced. In this way, our framework isn't just a feature-centric one, it's a *feature-driven* one. Because of this, it's crucial that our features are well motivated and well grounded. A flaw in our understanding of these features could have drastic effect for the success of whatever proposal it is that one is making.

For some types of features such a thorough discussion at this stage is unnecessary because some features, like number, are semantically intuitive and their distribution and any internal structure is well understood. Case features do not share this status and because of this, discussions on both the nature of case features and the relationships between them and a discussion of the issues that case feature proposals face is warranted. The complexity of some of these issues is often overlooked,

at least in the more syntactic-focused literature, and so an explicit exploration is worthwhile. Our understanding of the features relevant for case valuation should include the following: a well grounded motivation for both feature existence and distribution, a well grounded understanding of any hierarchical relationships that may hold between those features, and an understanding of how underspecification is modeled.

With respect to the first of these needs, an understanding of how features are motivated and distributed, we can turn to McFadden (2007) for guidance on what proper grounding should arguably look like. McFadden focuses on how poorly grounded features can have detrimental consequences for any proposal that employs them and provides some guidelines for how these features should be grounded in a way that avoids such repercussions. He frames this discussion through an examination of the decomposed features that make up the standard case categories. Decomposing these categories into individual features is an oft-employed strategy for accounting for case syncretic patterns and is widely accepted throughout the morphological case literature (see McFadden, 2007; Müller, 2004b, 2005, for a few examples)

While this strategy successfully models how syncretic patterns arise in the various languages in which they are observed, McFadden cautions that without a set of principles to constrain them, there is nothing to rule out potential patterns that turn out to be unobserved. In this way, McFadden argues that properly grounding case features is essential to proposing a system that has explanatory value, rather than simple descriptive adequacy. He proposes a Morphological Feature Constraint, shown in (8), that attempts to provide these necessary constraints.

(8) *Morphological Feature Constraint:*

The positing of a particular feature to handle patterns of morphological form must be accompanied by an explicit theory of its distribution in syntactic/semantic terms.

(McFadden, 2007)

This constraint essentially requires that features need to be grounded in a way that is independent of the primary function they are to perform in the grammar. Once we have a properly grounded and

well understood set of case features, we expect any intrinsic relationships that might hold between them to become quite obvious. In addition to understanding how the features we've proposed are distributed among the syntactic objects in a particular derivation, we need to also make sure to encode those relationships in our model of the case feature system.

We face a number of issues when proposing case feature systems, most of which are actually unique to case features specifically: (i) case is inherently difficult to ground, (ii) case is also fairly difficult to model, and perhaps unsurprisingly, (iii) the vast differences in basic theoretical assumptions about the nature and role of case introduce further difficulties in proposing a case feature system that is widely accepted. Throughout this discussion, it's important to note that I'm not suggesting that the issues raised here are insurmountable (or even unresolved in some instances), but rather that the task is more complicated than one typically assumes and that there is benefit to being explicit about these difficulties.

Case features themselves are inherently more difficult to ground given their lack of semantic content. It is well known that the case a DP receives does not correspond to a consistent semantic role, at least for the structural cases. On its own, this does not necessarily make grounding case features any more difficult than any other morphosyntactic feature; however, when coupled with standards for independent feature grounding and a dual role in multiple components of the grammar, this lack of semantic content creates a seemingly impossible task: we must ground case features independent of their function, while simultaneously needing to ground them in at least one of them.

To illustrate how a lack of semantic content can cause issues for feature grounding, McFadden (2007) contrasts case features with a feature with full semantic meaning: *person*. He points out that while we can certainly debate the specifics of how first person is represented featurally, there is a limit to the possibilities we can pursue because it is quite easy to determine whether a nominal is first person or not. In this way, the existence of semantic content constrains the nature of the potential features involved, thus limiting the range of possibilities which provides us with greater explanatory value. Semantic content can be viewed as a sort of scaffold onto which one can frame a particular feature's grounding and distribution.

The lack of semantic content with case removes these important constraints and therefore makes the task of independently grounding features that much more crucial. When you pair this fact with the assumption that case serves functions in multiple parts of the grammar, the task of proposing a well defined set of case features that will provide explanatory value is made much more difficult. As discussed above, McFadden also calls us to ground features independent of their primary functions. For McFadden, this is less of a problem because he assumes case does not play a role in the syntax and can independently ground these features there. However, if we maintain the traditional assumptions regarding the dual role of case in the grammar, this means that we must ground case independent of both its morphological function and its syntactic one in order to satisfy grounding best practices. The semantic component is therefore the most independent place to ground these features and unfortunately with case, it is also one that is unavailable to us.

I suggest that this difficulty is why case feature grounding has never felt truly satisfying and it explains why stipulation critiques have gone largely unaddressed – to some extent they are unavoidable. I suggest that the stipulative nature of case feature proposals is not the result of failure to capture case accurately, but is rather a natural artifact of case’s intrinsic nature. It is unsurprising to recall that in Chomsky (2001) case features are the only ones features assumed to be uninterpretable on both DPs *and* functional heads. In this way, they can be viewed as the only set of purely formal features, with no interpretable component. In the proposal that follows, I provide motivations for both the existence and distribution of the features I assume to be responsible for case’s dual functions, but it’s important to keep in mind why they don’t feel as well motivated as we might expect for other features.

In addition to being difficult to ground, case features are also actually quite difficult to model. Most of the syntactic literature on case discusses case categories, like nominative or accusative, but rarely discusses the details of the features that make up these categories. While understandable given the focus of that research, it is important to understand why something as simple as “T assigns NOM to a DP with which it agrees” isn’t actually a simple operation at all. In some sense, the concept of nominative doesn’t even quite exist – it’s a label we’ve given to more easily discuss

a set of features. As Pesetsky (2013) points out, case categories are a sort of ‘middleman’ between the actual features and the morphological forms.

We’ve just outlined how a lack of independent semantics makes case features more difficult to motivate, and now we turn to the large number of choices that we face when modeling those features and how each of these choices presents its own set of challenges for the researcher. It is not my intention to make the claim that case features are impossible to model or even that previous attempts to have done so are unattractive, but rather to be very clear about why choosing a particular set of assumptions to hold is not a simple task and why case valuing operations are not as straightforward as we often assume.

One aspect of case features that makes them particularly tricky to model is their status as valued/unvalued or interpretable/uninterpretable. Recall Chomsky’s 2001 proposal that the features responsible for case both on functional heads and on DPs are to be understood as uninterpretable. The motivation for this is that uninterpretable features are defined as being unable to be interpreted by the semantic component. As we’ve seen, the features responsible for case do seem to lack an inherent semantic meaning and by this definition it is reasonable to assume that case is uniquely uninterpretable on both types of syntactic objects. This assumption does not come without a cost, however. By modeling case features as uninterpretable on functional heads, we are implicitly encoding a grammatical requirement that functional heads with case features must assign case. While this alone is not reason to reject such a characterization, it *is* at odds with how we traditionally understand case, a requirement the grammar imposes on DPs solely. It also introduces a further question: why would these traditional case assigning heads need to assign case in the first place? Unlike with DPs, there is no evidence that case performs any additional function for functional heads that selection can’t account for. Despite this consequence, we clearly cannot maintain the alternative – that case features are interpretable on either functional heads or DPs given case’s complete lack of semantic content, at least not without drastically modifying what it means to be interpretable. More recent research has moved towards differentiating the partner versions of features along a different dimension: whether they are inherently specified with a value or

whether they must receive a value through establishing relationships in the derivation (Adger, 2003; Pesetsky & Torrego, 2007). By this definition, it is no longer unreasonable to assume that case features come in two flavors. Syntactic objects with needs, such as DPs that seem to need licensing beyond selection and instructions for determining form, should be specified with unvalued versions of the relevant feature. Syntactic objects that supply or fulfill these needs, are specified with the valued version of the feature. This move also provides an additional simplicity of model benefit. For verbs that optionally take an internal argument, assuming case on functional heads is uninterpretable required us to propose two flavors of every verb that shares this property: one with an uninterpretable case feature, and one without. Instead, if we propose that case is valued on functional heads, we can maintain one flavor for these verbs. If that verb fails to find a DP to license, this failure is of no consequence to the derivation: no unvalued features remain at the interfaces. I will expand on the assumptions regarding the nature of case features more explicitly when outlining the exact details about the case feature proposal I offer later in this section.

Case feature modeling also runs into issues with respect to the degree of specificity case features should encode and the number of the features themselves. The syntactic component only requires that the features responsible for case represent a binary distinction, to reflect the binary nature of licensing – DPs are either licensed or they are not. Any degree of specificity greater than this is superfluous for this particular function. The morphological component, however, requires a larger degree of specificity – an X-way distinction where X is the number of morphological cases observed in a language. The morphological component requires that case features reflect a greater number of distinctions than the syntactic component does. Clearly, we must prioritize the needs of the morphological component, as it would be impossible to model an X-way case system without an appropriately large inventory of case features, but it's important to remember that the syntactic component needs to simultaneously be able to interpret this X-way distinction in a binary way. This is certainly not impossible and is largely achieved through some of the standard models of case features in the syntactic literature, where [case] is a feature that has X possible values, one for each of the morphological cases present in the language in question. The syntactic component

is able to identify a binary distinction in whether or not the feature [case] has received a value. The morphological component is then able to look at the actual values that the feature has, which produces the required X-way distinction. What we need to be cautious of, however, is that while this sort of model works well for understanding how the needs of both grammatical components are met, it is unclear how this can be easily extended to a system that does not treat the traditional case categories as monolithic units. If we decompose the case categories into independent features, it's unclear how we reconcile that choice with the syntactic component's need to be able to identify a binary distinction. Again, this is not an insurmountable issue, but is one that needs to be remembered when proposing a workable system of case features.

4.2.1.2 The Hierarchical Nature of Case Features

With these issues in mind, we can begin to work through what we know about case features and use this information to propose a system that can capture the right empirical facts. Because there is a division in labor over the morphological and syntactic functions of case, there winds up being an understandable lack of consensus around a unified approach between researchers focused on each of these two components. Morphological case research tends to focus on patterns of case syncretism, whether case is expressed affixally, and capturing the varied morphological case patterns we observe across the world's languages. Syntactic case research has seen a more recent uptick in debate, with old disagreements reemerging and long-held assumptions being reexamined, as we've seen in earlier chapters of this thesis. Rarely do these aims converge and as a result, discoveries made in one arena rarely inform discoveries made in the other. One of the goals of this chapter is to bring together conclusions drawn from each of the various fields and propose a system of case features that can address the independent concerns of each.

There are two important observations about case that will guide our proposal: (i) the observation that case categories are implicationally hierarchical and (ii) the observation that case categories are not atomic units, but are instead composed of a number of smaller individual case features.

Case categories appear to involve implicational relationships between one another. Blake (2001)

proposes that there is a universal implicational hierarchy involving the inventory of case categories a particular language has. This hierarchy is shown in (9). The way to interpret this hierarchy is to say that if a language has a case category X, it will also have the case categories, Y, Z, etc that appear to the left of X in the hierarchy below. If a language has dative case, then it is also true that the language will have nominative and accusative case as well.

(9) NOMINATIVE > ACCUSATIVE > GENITIVE > DATIVE > INSTRUMENTAL > COMITATIVE

Evidence for this implicational hierarchy is also supported through acquisition data involving case categories. Austin (2012) provides evidence that children learning Basque acquire absolutive verbal agreement before they acquire ergative agreement, which is then followed by the acquisition of dative agreement. This relationship mimics the pattern shown in Blake's hierarchy above.

In addition to evidence that suggests case categories bear implicational hierarchical relationships to one another, there is also evidence that case categories themselves are not made up of atomic features, but are instead compositional categories composed of a number of individual case features. The primary evidence for this conclusion comes from a large body of work on case syncretism (see Baerman, Brown, & Corbett, 2005, for an overview). Syncretism is a specific type of homophony that we find in inflectional paradigms. Formally, it is understood as the grammar failing to make some sort of morphosyntactic distinction that under normal circumstances is made. It is a systematic phenomenon and in this way is different from the kind of accidental homophony that might result from the application of independent phonological rules. Take the following example from Russian:

Table 4.1: Accidental Homophony in Russian

a. stem-stress 'place'			b. end-stress 'wine'	
	orthographic	phonetic	orthographic	phonetic
NOM/ACC SG	mesto	'mʲe.stə	vino	vʲi.'no
GEN SG	mesta	'mʲe.stə	vina	vʲi.'na

(Baerman et al., 2005)

Here, while the genitive and nominative/accusative singular forms are homophonous for the lexical item ‘mesto’, they are not for the lexical item ‘vino’. Russian has an independent phonological rule whereby the distinction between /a/ and /o/ is only observed when in a stressed syllable. Since the homophony in table 4.1 can be explained in phonological terms and is not systematic to the case paradigm, it would not be considered an example of syncretism of case.

Table 4.2 shows syncretism in the case system. We can see below that Greek differentiates between 4 cases (NOM, ACC, GEN, and DAT) and also makes a distinction between masculine and neuter gender. In each cell is the morphological form for the adjective *wise*, showing how it varies with respect to how Greek expresses case and gender features. An adjective that is nominative and masculine will be expressed as *soph-os*, while the same adjective will take a different form if it is specified with nominative and neuter features. What the framed cells show is that in the neuter paradigm, the distinction between nominative case and accusative that is normally marked in the masculine is neutralized – the nominative and accusative cases both share the same form *soph-on*. When two normally distinct categories are expressed via the same form, as they are here, we call them *syncretic*. What distinguishes the syncretic type of homophony in table 4.2 from the homophony in table 4.1 is that the homophony in Greek is systematic for the entire neuter paradigm and isn’t reducible to any sort of independent phonological processes. This means that for all adjectives, not just *wise*, the grammar does not make a morphological distinction between nominative forms and accusative forms, even though it does make those distinctions in other gender paradigms, like masculine.

Table 4.2: Greek Adjective ‘wise’

	NEUTER	MASCULINE
NOM SG	soph-on	soph-os
ACC SG	soph-on	soph-on
GEN SG	soph-ou	soph-ou
DAT SG	soph-oi	soph-oi

Syncretism as a process involves the neutralization of distinctions and is viewed as evidence

that broader morphosyntactic categories are comprised of more individual features because the syncretic patterns are indications that the grammar forms natural classes from the categories in question. In order for the grammar to do this, there must simultaneously be a set of features that the natural class can reference and a set of features that maintains the distinction between disparate categories. Analyses of syncretic patterns hope to propose formal ways of encoding those natural classes, defined by morphological behavior. So for two case categories, X and Y, syncretism between the two would involve neutralizing their differences in such a way that the properties they share are the only ones remaining. Typical syncretism accounts involve proposing a number of post-syntactic operations that modify the feature specifications in ways that eliminate any feature overlaps. I've illustrated a brief example of how this works below in (10). Categories X and Y share the feature set {+a, +c}, but are distinguished through having different values for feature {b}. If we wanted to neutralize the distinction between the two categories, ie: make them syncretic, we could eliminate or otherwise delete the feature that encodes the distinction between them, {b}. The result of doing so is that both categories would then be identically specified with only the features they share and could invite the insertion of the same vocabulary item.

- (10) a. X: {+a, -b, +c}
 b. Y: {+a, +b, +c}

Case syncretism makes the task of proposing a unified system of case features even more difficult than we hinted at in the previous section because of the additional constraints it imposes on hypothetical feature specifications. Not only do we need to propose features in a way that is grounded independent from primary function, but we also must make sure to propose a system that allows us to model the correct potential natural classes that their morphological behavior suggests.²

²It's important here to understand that while the original featural specifications are important for correctly predicting the observed syncretic patterns, not all languages exhibit identical patterns and it is not considered a problem. The bulk of the theoretical work is borne by the post-syntactic operations that modify the original featural specifications. So while the proposed feature system must be decomposed enough to be able to capture natural classes between the cases, it doesn't (and shouldn't) need to be modeled after particular examples of syncretism.

Since we see syncretism in between the nominative and accusative cases in Greek, we need to decompose the case categories in a way that we can identify at least one feature that they share and at least one feature that distinguishes them. This is how syncretism provides evidence that feature categories aren't monolithic units, but are rather made up of a set of more individual features.

Informally, we can capture some intuitive natural class behaviors that group various case categories. First, we make a distinction between structural and nonstructural cases. Nominative, accusative, genitive, and dative are considered the structural cases, dependent on syntax rather than semantics. The case categories like instrumental, ablative, and locative, to name a few, are considered nonstructural and are assumed to have a semantic component to their interpretation and assignment. Among the structural cases, we can further distinguish between the core and non core cases. Core cases include nominative and accusative and the noncore cases include dative and genitive cases. Baerman et al. (2005) outlines some general tendencies for syncretic patterns that use these distinctions that I'd like to outline here quickly. Languages typically exhibit one of three possible syncretic patterns. First, there can be syncretism between the two core cases, like the syncretism between nominative and accusative case that we saw in the neuter paradigm in table 4.2. Second, there can be syncretism between a core case (accusative) and one of the noncore, but structural cases like genitive or dative. This is what we see in table 4.3 where the accusative and genitive cases are syncretic with nouns, but not pronouns. Interestingly, syncretism of this type is almost always restricted to the 'marked' core case (accusative or ergative). Finally, there can be syncretism within the noncore cases, which is the pattern we see in table 4.4 where the dative and the illative are syncretic in singular definite nouns, but not in indefinite ones.

Table 4.3: Finnish Syncretism Core/Non-core

	noun 'lock'	pronoun 'I'
NOM	lukko	minä
ACC	luko-n	minu-t
GEN	luko-n	minu-n

Table 4.4: Erzja Mordvin Syncretism Non-core Cases

	‘the house’	‘(a) house’
NOM	kudos’	kudo
GEN	kudont’	kudon’
DAT	kudonten’	kudonen’
ILL	kudonten’	kudos
ABL	kudodont’	kudodo

The observation that case syncretism suggests a decomposition of case categories coupled with evidence that case categories themselves have a particular invariant order motivates Caha (2009) to propose a theory of syncretism that depends on the notion of contiguity. His proposal is shown below in (11):

(11) *Universal Case Contiguity*

- a. Non-accidental case syncretism targets contiguous regions in a sequence invariant across languages.
- b. the case sequence:

NOMINATIVE - ACCUSATIVE - GENITIVE - DATIVE - INSTRUMENTAL - COMITATIVE

He argues that the syncretic patterns informally outlined above can be explained by proposing that syncretism can only target categories that are contiguous on the case category hierarchy. This would allow syncretism between nominative and accusative cases, for example, but would bar syncretism between nominative and genitive cases, unless the accusative is also involved. More important for our purposes is his conclusion that universal case contiguity is only possible if there is a universal system of case features with invariant hierarchical organization. Syncretism offers a way for us to ground case features independent of their primary function in that its behavior, while morphological in nature, is independent from the actual assignment of the case features themselves in the syntax or their more general distribution.

4.2.2 A Proposed Feature System

To summarize before moving forward with the proposal: we have evidence that case categories have implicational hierarchical structure. We also have evidence from syncretism that case categories are comprised of individual case features. Furthermore, case syncretism patterns are constrained to only involve those categories that are contiguous along the hierarchy. These facts all together suggest that case features, like φ -features, have some internal hierarchical structure. It will be the focus of this next section to propose what those features are, ground them appropriately independent of their primary functions, and argue for a novel system of case features that researchers working in both morphological and syntactic domains can employ. This proposal has two main parts that reflect the dual function of case: its role in regulating nominal licensing and its role in morphologically distinguishing various case categories. We will therefore discuss the features responsible for the various case category distinctions we observe and the features responsible for nominal licensing and how those two disparate sets are related to one another.

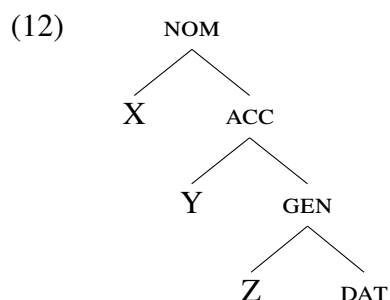
4.2.2.1 Morphological Functions

While novel in its details and its mechanics, this proposal draws heavily from a number of intuitions made by others about the nature of case categories and their features. We first begin with Caha (2009) as this will be the scaffold upon which the proposal is built. Caha (2009) concludes that the only plausible way to account for a universal case contiguity would be if the individual features that make up case categories are sub-classified rather than cross-classified. The primary reason for this is that the two types of relations make different predictions about adjacency/contiguity. He shows that a set of features that is cross-classified, shown in table 4.5 creates a system whereby a larger number of adjacent relationships are formed because there are both vertical and horizontal relationships available. Sub-classification, by contrast, requires a more linear set of adjacency relationships because the horizontal relationship is unavailable (12). This outlines the first characteristic of our proposed case feature system: case features are sub-classified. Perhaps unsurprisingly, the sub-classification nature of case features mimics one that is familiar to us throughout this thesis:

φ -features are also assumed to be sub-classified/hierarchical.

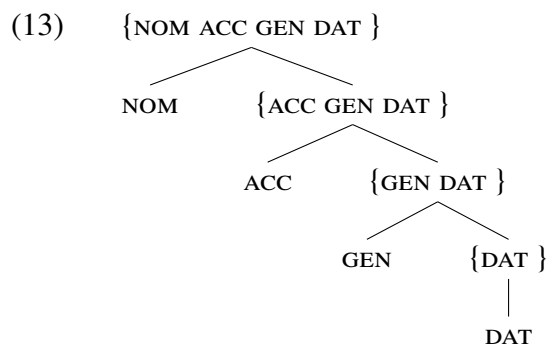
Table 4.5: Cross-Classification of Features

	+Y	-Y
+X	NOM	ACC
-X	GEN	DAT

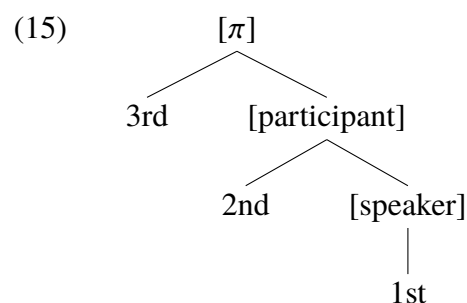
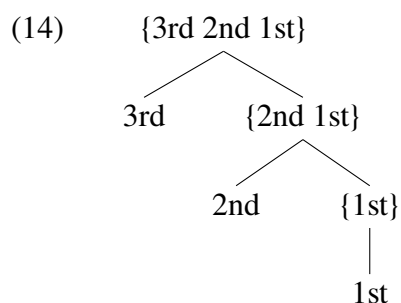


The logic of sub-classification would mean that as we proceed down the structure, we eliminate a case category with each level. Translating what this means for the features themselves, we can imagine that progressing down the structure could mean either removing – if the lowest member is the least specified – or adding – if the highest member is the least specified – a feature unique to the eliminated category. Following intuitions that nominative is the least specified case category (Baker, 2015; Levin & Preminger, 2015; McFadden, 2004), we’ll want to build a system where as you move ‘down’ the structure, each lower level will involve the addition of a feature that groups the remaining categories to the exclusion of the removed category. For an abstract example, consider the representation shown below in (13). We begin at the top of the structure, the set that includes all case categories. If we remove what we assume to be the least specified category, nominative, we are left with the set {accusative, genitive, dative}. To remove nominative of course, we must identify a feature A for which nominative is the only case category that does not bear that feature. Moving to the next level of the universal hierarchy, we want to remove the case category accusative. We must then identify a feature B that genitive and dative include, but not accusative. Finally, we

are left with a paired set. We must therefore identify some feature C that distinguishes genitive from dative.



Note that we can compare this to the already familiar internal structure of φ -features, shown in (14). All φ -features contain the feature $[\pi]$ that represents the whole set {3rd, 2nd, 1st} and as we move down the hierarchy towards more specified person categories, we add features that distinguish the remaining set from the removed category (15). We remove third person from the set and only second and first person remain. We therefore can identify a feature that categorizes the smaller set to the exclusion of the removed category, [participant]. Likewise, as we're left with the paired set {2nd, 1st}, we identify a feature that distinguishes them, [speaker].³



While we take the intuitions here about the need for sub-classification from Caha (2009), we are unable to adopt his system for our purposes because he does not ground the features themselves

³Of course, we could also identify a feature [addressee] that would similarly distinguish between the members of the same set. This would reverse the markedness relationship between them, making the spell out of [participant] first person rather than second. Some languages do in fact choose to express second person as the more specified member of the set (Béjar, 2003).

and continues to use ‘placeholders’ A, B, etc throughout his work. This is not a weakness as he’s operating in an entirely different framework, proposing that those features A, B, C are functional heads rather than independent features in the way we’ve been considering them throughout this thesis. So where there’s room for contribution is in how we ground the features that are responsible for distinguishing each of the eliminated case categories.

From here, we can turn to those who have focused their work on accounting for the specifics of various case syncretism patterns. No singular work can be adopted entirely, but I argue that we can combine insights⁴ from a number of different works, as we did with Caha (2009) to produce a system that both morphologists and syntacticians can be happy with. For clarity, here’s a reminder of the features we need to define:

- (16) a. a feature that encompasses all categories
- b. a feature that is common to {ACC, GEN, DAT } to the exclusion of NOM
- c. a feature that is common to {GEN, DAT } to the exclusion of ACC
- d. a feature that distinguishes GEN from DAT

I see no reason not to propose a feature [case] that is common to all case categories. It’s φ -feature counterpart [π] similarly signals category type in that it intuitively indicates all further specifications belong to the same common set.

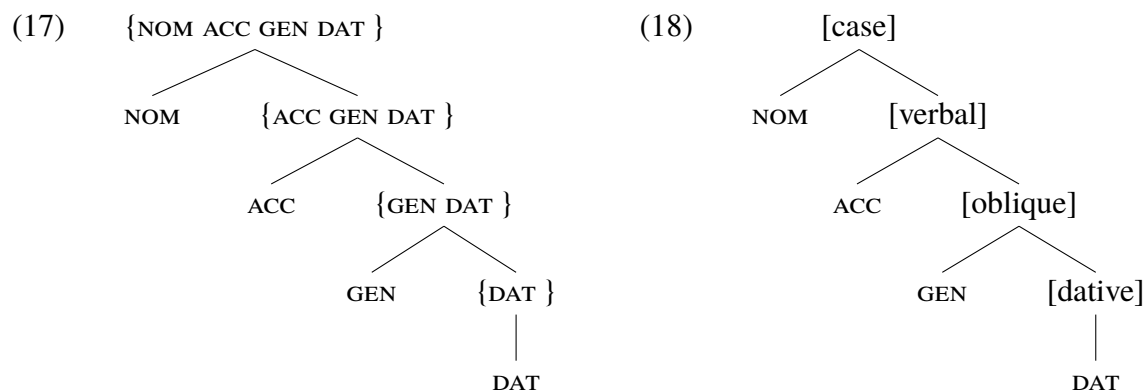
Next we need a feature common to {ACC, GEN, DAT } to the exclusion of NOM. I propose we adopt the intuition that we can group this set to the exclusion of nominative through reference to their ability to be expressed on objects of a verb (Calabrese, 1998; Müller, 2004b, 2005).⁵ I propose a feature [verbal] that is grounded in its ability to be assigned by verbs to their objects.

⁴I use the words ‘insights’ and ‘intuitions’ intentionally here. I will adopt none of the features actually proposed in the works I cite here. However, I recognize that what we call the feature and its motivation are superficially distinct and that it’s the motivation and the intuition behind the proposal that is actually important.

⁵Each of these sources does this in slightly different ways; most notably different is Calabrese (1998) who instead makes negative characterization that the relevant set is ‘not the subject of predication’. While the basic generalization is similar, I wish to note this here to not misrepresent his work.

The next feature should be common to {GEN, DAT } to the exclusion of ACC and I suggest we follow the fairly standard intuition that what divides these case categories is their status as oblique. This justifies the feature [oblique]. Finally, we must propose a feature that distinguishes genitive from dative. While superficially simpler due to the set only containing two members, this is arguably a more difficult task whose difficulty stems from the greater range of options available. Among the intuitions already offered in the literature, two appear most promising, but I'll refrain from adopting them nonetheless for reasons I'll discuss in a minute. McFadden (2007) disagrees that dative is the more specified member of the set and therefore his proposal of a [+genitive] feature to distinguish them reflects this assumption. Müller (2005) does not maintain that assumption, but still singles out genitive as the category with a defining property that a feature can reference. He proposes a feature [+n] that is assigned by nominals and uniquely identifies the genitive case to the exclusion of all others.

Because my intention is to capture the universal case contiguity argued for in Caha (2009) I'd prefer to propose a feature uniquely identifying the dative case category, rather than the genitive. We also cannot do this through negative reference to the property proposed by Müller (2005) because it does not uniquely distinguish genitive from dative; arguably accusative and nominative are also not assigned by nouns or in nominal environments. What I will propose is a feature [dative] that singles out the dative case. This makes the genitive case the default spell out of [oblique]. A summary of all proposed features and their internal structure is shown in (18)



However, I suggest that perhaps this disagreement might be informative. As we've just seen,

one concern is the status of the relationship between dative and genitive cases. The worry is that for some languages, dative seems less specified than genitive and for others, the opposite is true. I believe this is a place where one of the apparent weaknesses of my proposal might turn out to be a strength. The relationship between first and second person is similarly variable. Béjar (2003) notes that while the feature [participant] will always dominate the feature that distinguishes between the members of that set, grammars have a choice whether to adopt the feature [speaker] to be the distinguishing feature or to adopt the feature [addressee] instead. This allows the grammar to exercise some optionality in markedness. If the grammar singles out first person as more marked, it would use the feature [speaker] to mark first person. Under those assumptions second person would be the spell out of the less specified set [participant]. If instead the grammar singles out second person as the more marked category, it would use the feature [addressee]. Under those assumptions the first person would be the spell out of the less specified set [participant] and the second person would be marked with the [addressee] feature in addition to the rest of the hierarchy. If the relationship between genitive and dative is similar to the relationship between first and second person, languages might be able to select how they differentiate between genitive and dative cases by exercising a choice in which sister-feature they select to differentiate between the members of that paired set, either [genitive] or [dative].

What we've done is adopted the argument that constraints on possible syncretism patterns supports the adoption of a universal case contiguity that is captured through a case feature system that is made up of hierarchically organized independently grounded case features that comprise the various case categories. The adoption of the universal case contiguity hypothesis and some of the intuitions about what properties distinguish the various case categories are taken from those who've come before me, but their combination is novel.

Before moving to the syntactic function of case features, it is worthwhile to show how the morphological case system proposed here can account for some of the types of syncretic patterns introduced at the beginning of this section. To capture the syncretism is Greek between the nominative and the accusative, the grammar needs to neutralize the distinction between the two.

Since nominative and accusative case in this proposed system share the feature [case], but do not share the feature [verbal], the morphological component could delete the [verbal] feature in the contexts where syncretism shows up (19). After deletion, both case categories correspond only to the feature [case], allowing for the same vocabulary item to be inserted into either instance.

- (19) a. NOM = [case]
 b. ACC = [case [verbal]]

Likewise, we saw syncretism in Finnish between the accusative case and the genitive case. In order to get this syncretism to surface, the grammar needs to neutralize their distinction in a way that allows for one form to be inserted into either category. The logic of the approach is the same as it was for the Greek syncretism. The [oblique] feature is the feature that distinguishes them, so if the morphology deleted that feature (20), their featural specification would then be identical ([case [verbal]]), paving the way for the insertion of the same vocabulary item. I've included the nominative here to show that by deleting only the [oblique] feature, accusative and genitive will become syncretic, but they will still be distinct from nominative unless additional features are removed.

- (20) a. NOM = [case]
 b. ACC = [case [verbal]]
 c. GEN = [case [verbal [oblique]]]

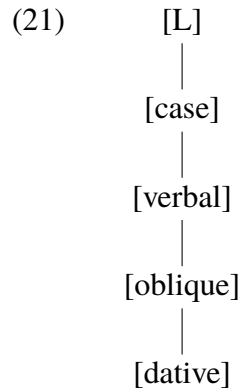
4.2.2.2 Syntactic Function

The proposed system is capable of making the 4-way distinction needed to differentiate between the structural case categories. We've yet to address how we understand nominal licensing to be handled in this system. This is the focus of the current section.

The general problem that previous Agree-based approaches to default case ran into was that they proposed a distinction between default case DPs and DPs that received a typical case. These approaches handled this data by arguing that each type of DP was generated with a different set of

case features. The differentiation between the two groups was grounded in different ways: Legate (2008) grounded the difference through the distinction between argument and nonargument DPs while Schütze (2001) made the distinction through random generation of case features. I argued in section 4.2.1.1 that regardless of the specifics surrounding how one grounds the distinction, modeling default case through a difference in DP type was both conceptually and empirically unsatisfying.

Instead I propose that in order to derive the intuition that default case is a product of its environment rather than its DP type, we should assume that all DPs are generated with the same featural specification, at least with respect to case. We've discussed that DPs have two needs: (i) they need confirmation that they appear in a position that is licit (licensing) and (ii) they, like other syntactic objects, need to receive instructions for how they are to be pronounced (morphological case). I propose that these two needs actually allow us to set up an entailment relationship that we could then use to motivate hierarchical relationships between the features that are responsible for each of these functions. If a DP has received instructions for pronunciation (ie: has a morphological form), then we know that that DP was licensed (ie: licit in the position it occupies). The opposite, however is not necessarily true. Default DPs are exactly the example that shows us that DPs can be licit in a particular position without having received instructions (from the syntax, via *AGREE*) for how they should be pronounced. So in other words, DPs that have morphological case must have been licensed, but DPs that are licensed do not necessarily have to have received morphological case. Following the logic set up in Béjar (2003), we can use this to motivate the proposal that the features responsible for licensing dominate those responsible for morphological case, discussed in the last section, (21)



Like morphological case features, it's important that we also ground the licensing feature [L]. Unfortunately, the simple concept of licensing itself is not something that is very well understood. Licensing seems to be a concept that essentially says "yes, you can appear here". This is hardly a theory of anything and makes its grounding incredibly difficult. It winds up meaning that we know something is a licenser if it allows something else to appear in some determined location or distance from it. I'm going to suggest that there are three types of functional heads with respect to satisfying the needs of DPs:

1. there are functional heads whose primary job is to fully integrate DPs.
2. there are functional heads that *can* integrate DPs, but whose job is not *primarily* this.
3. there are functional heads that simply cannot integrate DPs on their own.

Functional heads of the first type are going to be the canonical case assigners: functional heads like finite T, *v*, P, etc. These functional heads all primarily serve to help integrate DPs into the derivation they are a part of. Because their focus is on the DPs primarily, I suggest that it's not unreasonable to assume that these functional heads are capable of fulfilling both needs of the DP: licensing *and* pronunciation. How this observation is encoded in the featural specification proposed in this thesis is that these types of functional heads are specified with a licensing feature [L] and the relevant amount of morphological case feature structure unique to that particular head. Essentially, these functional heads come with the entire licensing/morphological case feature bundle proposed in (21).

Functional heads of the second type are those that can certainly integrate DPs into a structure, but aren't focused primarily on doing so. These functional heads are capable of integrating a wide range of category types and because of this, I argue that it is not unreasonable to assume that they do not fulfill all of the requirements of the DP, just the distributional one. In more traditional terms, what I mean by this is that these types of functional heads are not morphological case assigners, but they can play a role in licensing DPs. More detailed examples of what functional heads I have in mind will be further explained in the next section, but a quick example would be a coordinating head, like *and*. Coordinating heads surely can coordinate (and therefore integrate) DPs, but they can also coordinate a range of other category types:

- (22) a. [DP Jim] and [DP John] will go to the store.
 b. Jim will [vP go to the store] and [vP rent a movie].
 c. The store is [pp around the corner] and [pp down the street].

Because this function is not restricted to just DPs, we can assume that they do not come supplied with any DP-specific type features, like morphological case. The featural specification I will assume for functional heads of this type is:

- (23) [L]

Finally, to address the third type of functional heads, those that cannot integrate DPs on their own. Non-finite T is traditionally assumed to lack case features and this is what explains why DPs are typically unable to appear in the subject position of a non-finite clause. My assumptions for the featural specification for non-finite T essentially capture the same thing: I assume that non-finite T is unable to license DPs, and therefore is also unable to supply them with morphological case. The featural specification for non-finite T would therefore be:

- (24) [∅]

With focus on the case domain, we're tempted to assume that non-finite T is unique in this specification. However, the way we've grounded the three-way distinction between functional

heads is not case-specific. There are plenty of functional heads that do not license DPs. Functional heads within the DP and aspectual heads in the clause offer two quick examples, but any functional head that cannot license a DP in a local position would be assumed to have this (null) specification.

We have to be very careful about how we talk about the role of non-finite T and what it means to integrate DPs into a derivation. An intuitive understanding is relating this integration to something like a selectional feature, or an EPP feature. We clearly don't want to assume that non-finite T does not have an EPP feature; ECM clauses illustrate that non-finite T is still capable of triggering the movement of the external argument to its specifier. However, the DP cannot stay there unless it is licensed by something else, namely the ν of the embedding verb. In this way I intend to say that non-finite T cannot integrate DPs on its own, and therefore is not specified with *any* of the features related to the requirements of DPs.

A quick comment about the feature [case] and its comparison to [L]: in the following section we'll see that I propose that default case is the morphological spell out of the [L] feature, not the underspecified [case] feature. Conceptually, this is intended to capture the observation that what a default case DP is is one that has been licensed, but has not received any morphological case. However, it would be quite intuitive to instead assume that default case is the physical representation of the "root" [case] feature in the morphological case feature geometry – the least specified feature in the paradigm. As far as I can tell, we have two ways to model what this underspecification would look like.

First, we could assume that for a given language, the least specified morphological case feature in a particular paradigm is the exact feature that spells out whichever case category happens to be the selected default case category in that language. The benefit to modeling case underspecification in this way is that the default case category for a particular language is more or less derived, rather than stipulated. Furthermore, as McFadden (2007) argues, we could also see a reduction in the complexity of the system as we would now have a way to align canonical nominative case valuation with default case, treating the former as a subset of the latter. The problem here, however is that because languages do not universally select the same case category as its default, we are forced into

proposing an entirely separate case feature system for each language with a different default. While I think it's entirely reasonable – and traditional, assuming the Borer-Chomsky Conjecture (Baker, 2008a; Borer, 1984; Chomsky, 1995) – to assume that languages can vary in the feature inventory they select for case, I think it's much less reasonable to assume that any hierarchical relationships between the various case features should be different – and that's exactly what this system type would require.

Instead, we could propose an underspecified abstract type of case feature that dominated all other morphological case features. This case feature could then be sort of syncretic with whatever the default case category for a particular language would be. There are two primary benefits to this. First, unlike the first option, this type of model would allow us to maintain the same case feature system cross-linguistically. We lose the ability to derive which language was the default for a given language, but I would question whether or not this should be a goal in the first place. Given that the default case category can in fact vary cross-linguistically, I would suggest that deriving which case category was the default in a language isn't the goal. The goal should instead be to derive not the specific case category, but rather the set of possible categories from which languages are allowed to select a default. Proposing a [case] feature helps to preserve a universal system of case features and also captures the intuition that what a DP probes for is not a particular case, but rather a more or less abstract version of that requirement and also ties that requirement to its need for licensing.

4.2.2.3 A Summary

Before moving on to the discussion about how this feature system interacts with the MATCH/VALUE system, I'd like to refocus our attention back to some of the preliminary concerns introduced at the beginning of this section. As we mentioned in section 4.2.1.1, one of the biggest issues in grounding case features was that it appears to ask an impossible task: ground the feature independent of any primary functions, while needing be grounded in one of them. One could argue that this new system doesn't make any radical progress here, and to some extent I would agree at least with respect to some of the particular properties we've proposed like 'assigned by verbs'. However, I do think that

we've been able to get around some of the issues with respect to grounding the features independent of their purpose. While it is impossible to ground case features outside the component in which they operate, grounding their internal structure in their syncretic behavior is at least independent of both their nominal licensing function and their primary morphological marking function. There is still room of course to debate about whether or not we've identified the correct properties that single out each of the various excluded categories as we move down the feature tree, but the hierarchical order itself has allowed us to make progress in proposing at least the right scaffold for the system.

This system, when paired with a decomposition of *AGREE*, also allows us to reconcile the issue raised in section 4.2.1.1 about the different grammatical components requiring different degrees of specificity in how case features are modeled. If *MATCH* is only sensitive to the root feature, we can capture the binary need of the syntactic component, while *VALUE* being sensitive to the entire feature set captures the need for an X-way distinction that the morphological component requires.

We also, I think, gain some insight into one of the oldest case puzzles: the relationship between abstract and morphological case. Data like default case, quirky/inherent case, and other instances of unexpected case led researchers to conclude that these must be the result of mismatches between abstract case features that operate primarily in the syntax and morphological ones that primarily operate in the morphology (McFadden, 2004; Schütze, 1997). Once multiple kinds of case features were proposed, the question became what relationship holds between them: are morphological case features the direct spell out of abstract ones with some leeway that allows for the mismatches or are morphological and abstract case features completely independent from one another? What made this debate a difficult one is that if one assumes they are completely independent, we lose the overwhelming amount of redundancy where the two do align properly. If one instead assumes the morphological features are the spell out of the abstract features, we lose the ability to account for the instances where they do not align. I think this proposal provides some novel understanding into the relationship. Morphological case features, the ones responsible for dictating morphological form, are *dominated* by abstract case features, the ones responsible for regulating nominal licensing. This hierarchical relationship between the two allows us to maintain the intuition that these two

functions are independent from one another while at the same time, they are related.

4.2.3 Accounting for Default Case

Now that we've argued that case features are hierarchically organized, as φ -features are, an obvious question emerges: what would the application of MATCH/VALUE mean for the system of case assignment and would this allow us to capture default case in a way that allows us to maintain case's role in regulating nominal licensing while also avoiding the adoption of a problematic configurational case approach.

I borrow three well-argued for assumptions from Carstens (2016): (i) unvalued case/licensing features are probes, (ii) if a probe's feature remains unvalued after probing its c-command domain, it may continue to probe in the search space until it is sent to transfer at spell out, and (iii) unvalued features on heads may project to the phrasal level and continue their search from there if they are unable to find a value in their original c-command domain. While a few of these assumptions are unfamiliar, they fit in quite well with the standard assumptions about the basic architecture and so I feel comfortable adopting them as minor modifications to the framework. I'll discuss their motivations briefly here, but direct the reader to (Carstens, 2016) for further explication.

One feature of Chomsky (2000, 2001) that was arguably stipulative was that only φ -features probed. Despite case features also entering the derivation unvalued, they weren't assumed to probe on their own; in this way they were considered purely 'goal features'. Carstens (2016) argues that all unvalued features have the capability to probe, since this capability is grounded in the need to fulfill those requirements encoded by the presence of those unvalued features. I follow that assumption and assume that all DPs enter the derivation specified with identical feature specification: [uL[case]] and that this [uL[case]] feature is a probe. This captures the intuition that all nominals need both confirmation that they end up in a licit position (nominal licensing) and instructions for pronunciation (morphological spell out). The second need, the need for instructions for pronunciation, is only relevant if this first need is met and is handled by the features that the [L] feature dominates.

A quick comment on the exact location of these features: for shorthand throughout this explanation you'll see that I've located the case features on the category DP. It's quite standard to assume that case features actually exist on the D head itself, or in some independent K layer. With Carsten's delayed valuation proposal, features on a head X can project to category XP if they fail to match in their initial search domain. The motivation for this assumption comes from adjectival concord. Adjectival concord describes the existence of agreement morphology on adjectives that matches that of an agreeing noun. The data from Swahili in (25) shows that the adjectives 'good' and 'heavy' agree with the nouns they modify 'book' and 'load' in nominal class. With respect to adjectival concord Carstens assumes two things: that concord is handled by agreement and that adjectives head the AP adjuncts that participate in concord. The idea is that without assuming the agreement features on A can project up to AP, there is no way for them to participate at the phrasal level. The features on A would probe their c-command domain and never find anything with which to value; concord would be impossible, contrary to fact (25)

- (25) a. kitabu [_{AP} **kizuri** sana]
 7book 7good very
 'a very good book'
- b. mzigu [_{AP} **mzito** mno]
 3load 3heavy too
 'too heavy a load'

(Carstens, 2016)

I follow that assumption and since it will always be true that an unvalued case feature on a D head (or K head) will fail to find a case assigning functional head within its own projection, it will always project to the DP level before interacting with the rest of the structure. Since this interaction is what's relevant for the question at hand, I see no need to illustrate this in the derivations.

With respect to the directionality of probing, Carstens argues that its apparent downward-only nature is actually a simple consequence of the way the structures are built – bottom up – and is not an inherent characteristic of agreement. She argues that because the only available search domain for a probe upon merger into a structure is its c-command domain, the probe has no other choice but

to search ‘down’, at least at first. In many languages, complementizers can participate in agreement. What is interesting about complementizer agreement is that it’s not universally downward; some languages appear to exhibit upwards complementizer agreement where the complementizer agrees with a nominal in the higher clause, rather than in the embedded clause. Data from the African language Lubukusu illustrates this upwards direction of agreement (Diercks, 2013). What we see in (26) is the complementizer agreeing with the subject of a higher clause, rather than the one in its own clause. (The SA in the gloss indicates subject agreement.)

- (26) Khw-aulile [CP **khu**-li/*ba-li ba-limi ba-funa ka-ma-indi].
 1.PL.SA-heard 1PL-that/2-that 2-farmers 2SA-harvested 6-6-maize
 ‘We heard that the farmers harvested the maize.’

Carstens assumes the upwards direction of agreement is constrained in the following two ways: it is only possible when a particular probe has exhausted its c-command domain without finding a value and it must obey locality constraints enforced through transfer to spell out. What this means is that a probe is allowed to probe upwards, but *only* after probing first into its c-command domain and only until it is transferred to spell out. Upon that point, any remaining unvalued features are assumed to cause a derivation crash. Adopting this assumption allows us to abandon an agreement directionality parameter (Baker, 2008b; Diercks, 2011) and since adopting her assumptions about directionality fit in with the general architecture and allow us to simplify the system, I suggest they are reasonable modifications to make.

With those assumptions in place, we can begin to explore how the MATCH/VALUE system interacts with the novel case feature system proposed in the previous section. For reference, table 4.6 outlines the possible outcomes for various featural specifications with respect to the licensing/morphological case bundle I’ve proposed in this chapter. The ‘probe’ column is specified with the same featural specification, since we propose that all DPs enter the derivation with the same needs. The ‘goal’ column lists the possible featural specifications we’ve laid out for various functional heads. The MATCH and VALUE columns each list the outcome of that particular individual operation, given the featural content of the probe and goal. Finally, the ‘morphological outcome’ column gives us the

morphological result of the two operations: a case category, default case, or ungrammaticality.

Table 4.6: Decomposition of Cases:

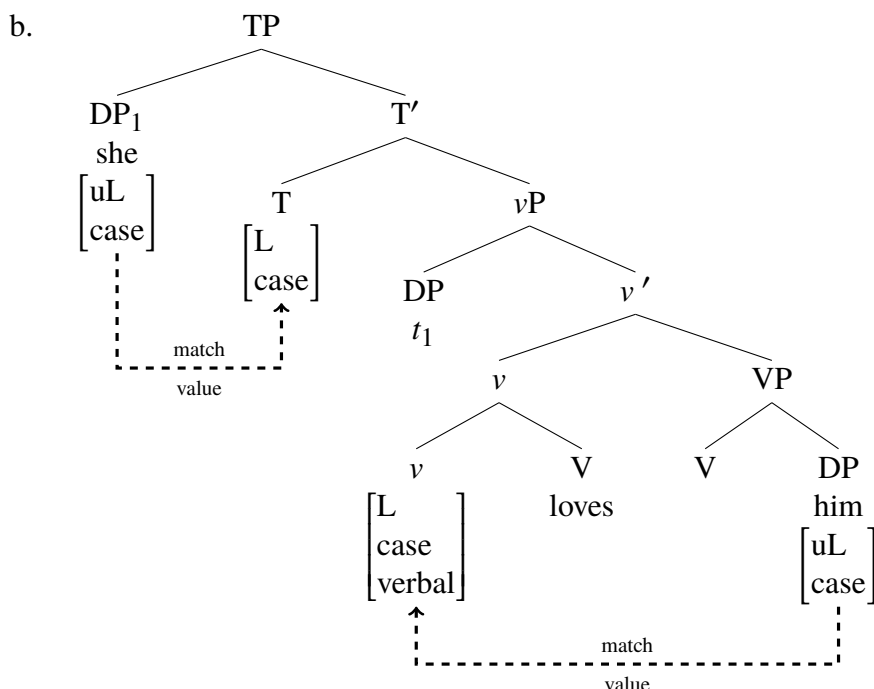
Probe	Goal	MATCH	VALUE	Outcome
[<i>u</i> L [case]]	[L [case [verbal [oblique [dative]]]]]	yes	yes	DAT
[<i>u</i> L [case]]	[L [case [verbal [oblique]]]]	yes	yes	GEN
[<i>u</i> L [case]]	[L [case [verbal]]]	yes	yes	ACC
[<i>u</i> L [case]]	[L [case]]	yes	yes	NOM
[<i>u</i> L [case]]	[L]	yes	no	default form
[<i>u</i> L [case]]	∅	no	no	ungrammatical

This section is intended to walk through how case valuation proceeds for a number of different circumstances, assuming the proposal I've set up so far. I first start with some canonical case valuation examples, just to illustrate that I've not made any drastic modifications to the traditional story of how case valuation works. I then show how each of the default case environments produces default case on the relevant DPs.

4.2.3.1 Canonical Case Valuation

I'll begin this walkthrough with a simple example that illustrates both canonical nominative case valuation and also accusative case valuation. Take the simple transitive clause in (27a) and its derivation in (27b):

(27) a. She loves him.



First, we'll start with canonical accusative valuation. Since the [uL[case]] feature bundle on the object DP *him* serves as its own probe, it will begin to search its c-command domain. Upon not finding anything with which to AGREE, it is allowed to continue the search upwards (Carstens, 2016) until the phase it occupies is spelled out, at which point it *could* cause ungrammaticality if left unvalued. The most local matching feature is the [L] feature on the *v* and the probe finds both a successful MATCH and a successful VALUE. Since VALUE is successful, the features on *v* are then copied over to the DP. The DP will then be spelled out as the accusative pronoun *him*.

The subject DP *she* also has a [uL[case]] feature bundle which will serve as its own individual probe, separate from the one on the object DP *him*. After moving to SPEC_{TP}, this feature will probe, finding both a successful MATCH and a successful VALUE on with the finite T that is in its c-command domain. The spell out of this DP will be the nominative pronoun *she*.

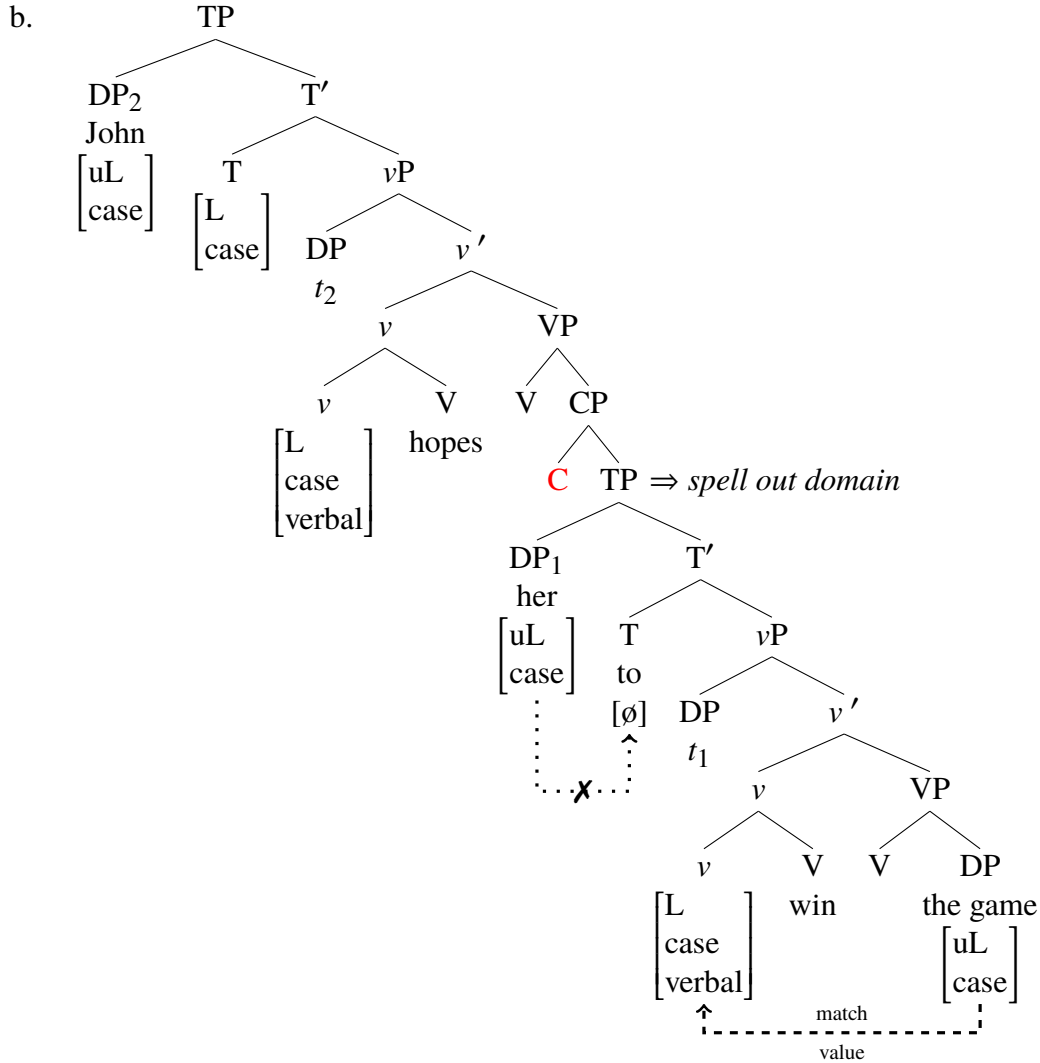
A reasonable question to raise at this point is why the external argument's probe isn't able to agree with the features on the *v* in (27b), since *v* is more local to the DP than finite T. I'm going to argue that since the case features on the *v* functional head have already established a

dependency with the internal argument, the functional head is unavailable to establish future case related dependencies with other DPs – it is not obvious however why this should be true. It is an important concern because with the modification that case and licensing features on functional heads are now valued, there needs to be some mechanism that prevents them from valuing multiple nominals in a clause. Under the standard Chomskyan approach, this issue is avoided with respect to case because once φ -agreement values case on the nominal, it deletes the uninterpretable case features from both the nominal and the functional head. The deletion of the case feature on the functional head prevents case from being reassigned to another nominal by the same head. For now I will be forced to say that once a functional head licenses a nominal it can no longer license another nominal, largely by stipulation.⁶

⁶There is a similar problem in the φ -feature domain under the standard approach. The φ -features on nominals are interpretable and are not deleted when the nominal agrees with a functional head, so in theory they should be able to continue to value other φ -probes on functional heads. What stops them is a constraint on AGREE that says that nominals can only participate in agreement if they have another uninterpretable feature, namely an uninterpretable case feature. There might be a way to extend this sort of proposal in reverse to the realm of case assignment by proposing that there needs to be an unvalued feature on the functional head for it to be able to participate in AGREE with respect to case. An obvious contender would be φ -features which would help account for the DP fulfilling functional heads, but wouldn't work for functional heads like TOP, for example. Another approach would be to somehow model that functional heads that receive something in return from the nominals they license are frozen in a way that prevents them from participating in future AGREE dependencies. This could circumvent the issue of proposing some unvalued feature that the disparate set of licensing heads share. The intuition would be that two-way dependencies are somehow stronger than one-way dependencies and the strength of that dependency freezes it from future participation in other dependency establishing operations.

Next, a canonical ‘ungrammatical due to a Case failure’ type of example:

- (28) a. *John hopes her to win the game.

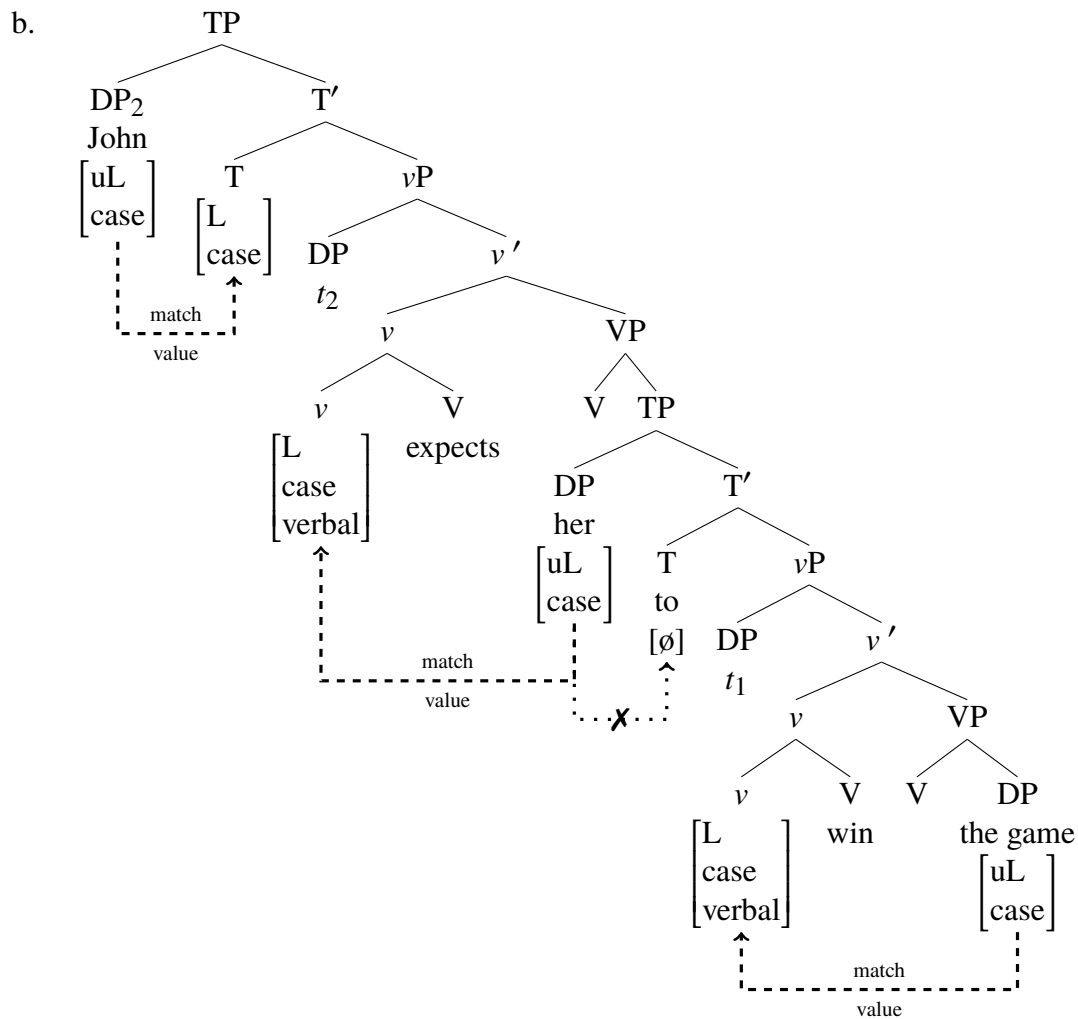


Accusative case valuation for the DP *the game* works exactly as it did for the object DP *him* in example (27b). Next, the embedded subject DP *her* probes down its c-command domain and does not find a MATCH at all in the non-finite T functional head because this functional head is exactly the type that is not specified with any of the case features proposed in section 4.2.2.2. The DP fails to MATCH and is actually allowed to continue probing since it has not yet been sent to spell out via the merger of a phase head. It reaches the *v*, but although this functional head was originally generated with relevant licensing/morphological case features, it has already AGREE-d with the object DP

the game and therefore these features are no longer available to participate in further AGREE. The embedded subject DP, having exhausted its c-command domain, is therefore allowed to continue to probe upwards, still searching for something with which to MATCH. It is allowed to do so until a phase head causes the spell out domain to be transferred. The probe attempts to search upwards, but upon the merger of the C phase head, the TP spell out domain is spelled out and having failed to value its case feature, the subject DP causes the derivation to crash.

To show how embedded DPs in ECM clauses receive a case value, in contrast to ‘regular’ non-finite clauses like the one we just saw, let’s look at (29b)

(29) a. John expects her to win the game.



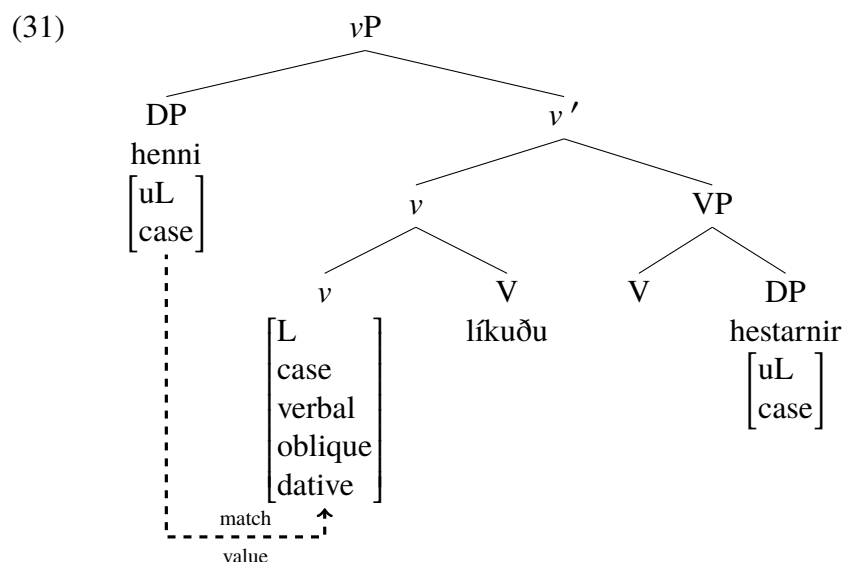
Here, the derivation proceeds exactly as it did in (28b), but because no C has merged into the structure – triggering spell out of TP – the embedded subject may continue its search past the TP into the matrix clause. It then finds a *MATCH*, and subsequent successful *VALUE* in the *v* of the matrix clause. The embedded subject is therefore spelled out with the [ACC] features it receives from the matrix *v*.

4.2.3.2 Quirky Case

Next, we need to sketch how Icelandic quirky case might operate under this analysis. Icelandic famously has non-nominative subjects, often called quirky subjects (Andrews, 1982; Sigurðsson, 1989; Thráinsson, 1979; Zaenen, Maling, & Thráinsson, 1985). When these non-nominative subjects surface, the object surfaces with nominative case, instead of the usual accusative case. This is shown in (30):

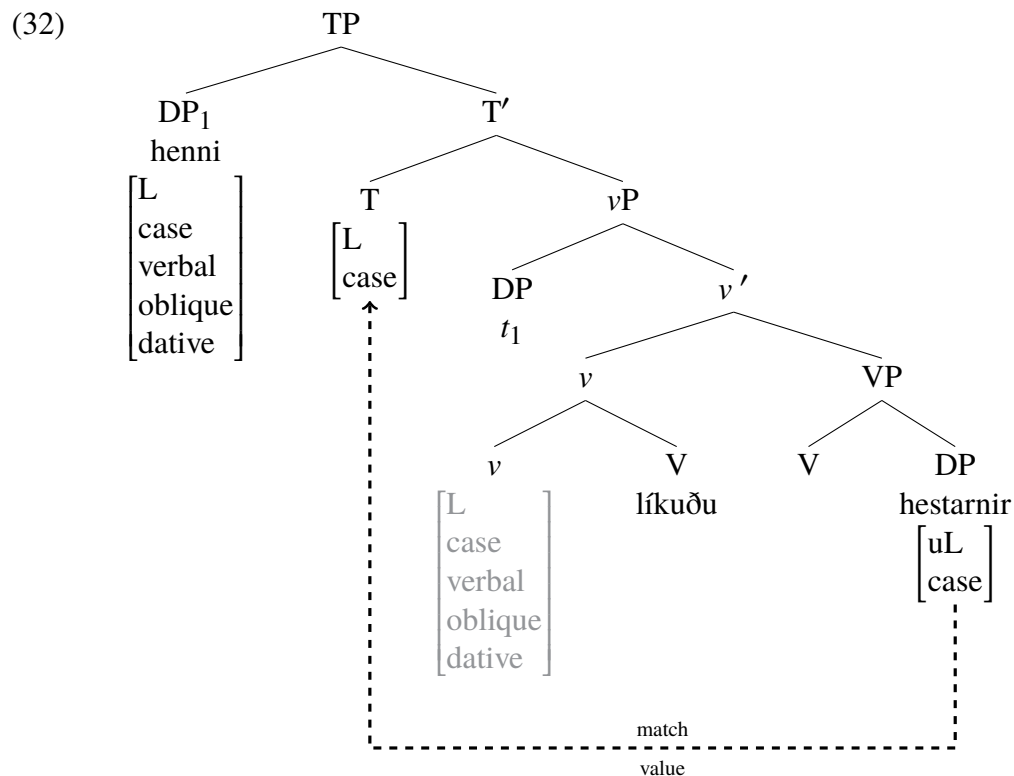
- (30) Henni líkuðu hestarnir
her.DAT liked horses.the.NOM
‘she liked the horses’ (Harley, 1995)

The structure of the *vP* for the example in (30) is shown below in (31):



I assume the ν that merges with quirky verbs is lexically specified with whatever case features correspond to the quirky case required by that verb (Schütze, 1993); in (31), these are the dative features required by the verb *líkuðu*. What we have to assume is that by virtue of being a quirky assigning verb, it is unavailable to the internal argument, otherwise we might expect quirky case on the internal argument as it merges into the structure first. Immediately upon merge, external arguments of quirky verbs automatically probe looking for a MATCH and VALUE for their licensing/morphological case feature bundle inside the ν' . The bundle finds both a successful MATCH and VALUE with the features on the ν that merged with the quirky verb. This successful MATCH and VALUE also has the effect of eliminating the quirky verb from being able to provide licensing/morphological case features to any other DP, leaving the object DP's licensing/morphological case bundle still unvalued.

The derivation then continues as usual, with the merging of T along with the typical movement of the external argument to the SPEC TP position:



The object DP's probe has still yet to successfully *MATCH* and *VALUE* its unvalued licensing/morphological case bundle. It is therefore allowed to continue to probe upward until it finds something with which to *AGREE*, since the quirky case assigning verb was unavailable. Recalling example (28b), one might be concerned about agreement between T and a VP-internal object, given that a phase boundary *vP* appears between the two that should make its constituents unavailable for syntactic operations. If we adopt the weaker version of the PIC from Chomsky (2001), however, this is not a problem. Chomsky (2001) allows agreement between T and a VP-internal object provided that the next phase head, C, has not yet been introduced. In other words the spell-out of *vP* is not triggered until the introduction of C. What this would mean here is that the object DP is able to probe into the TP, provided this happens before C is merged into the structure. Following this assumption, the object DP is then able to continue to probe past the *vP* and finds a successful *MATCH* and *VALUE* with the finite T that usually assigns nominative case to the subject. The object in quirky verb constructions therefore surfaces with nominative case.

While some of the details of the mechanics of the canonical and quirky examples look a bit different than the traditional story, I'd like to point out that the underlying idea is exactly the same. Because these examples involve either a successful *MATCH* and *VALUE* or they involve an unsuccessful *MATCH*, they aren't really different than the traditional account where something either successfully agrees or doesn't. Where this proposal really differs from the traditional Case story is in the cases where *MATCH* is successful, but *VALUE* is not: the places where I predict default case to surface.

4.2.3.3 Hanging Topic/Left-Dislocation

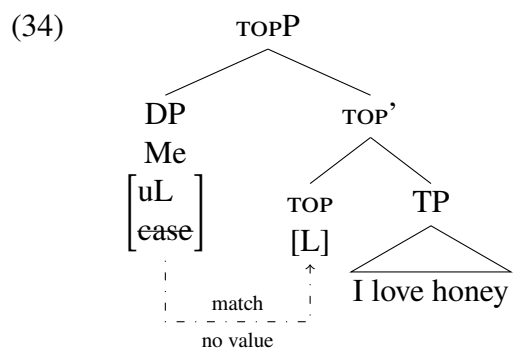
Now that we understand how the approach proposed in the last section would account for canonical case valuation and failure, I'd like to begin our discussion of how the *MATCH/VALUE* based proposal would account for the distribution of default case forms in the default case environments by looking at what is possibly the least complicated environment: the left periphery, including left-dislocation and hanging topics.

(33) *Hanging Topic/Left-Dislocation*

- a. **Me**, I love honey.
- b. What?! **Him** wear a tuxedo?!

Both examples in (33) are considered left-dislocation, where some element is merged into a topic position at the left periphery.⁷ That there is syntactic material in the left periphery is obvious in (33a), but less so in (33b). Lambrecht (1990) proposes that sentences like (33b) actually involve the left-dislocation of both the subject and the predicate into two independent topic positions. For ease of explication, I'm going to provide a walkthrough of (33a), but follow Lambrecht (1990) in assuming that (33b) would also involve left-dislocation and therefore we should expect it to work similar to (33a).

I assume that the hanging topic elements occupy a 'topic' position – the specifier of a TOPIC head.⁸ In order to be grammatical, they must have a reasonable associate. For the sentence in (33a), I assume the structure in (34):



With respect to the task at hand – accounting for how the left-dislocated DP ends up in the accusative form – I propose that like all DPs, it should be specified with the licensing feature I

⁷I don't think that assuming a base generated story for left-dislocation or a movement based account of left-dislocation makes a difference for my proposal, so I leave these details for another time.

⁸Likewise, I would assume that left-dislocated elements occupy the specifier of a FOCUS head. This FOCUS head would have the identical featural specification with respect to the licensing/case bundle as the one I'm proposing for the TOPIC head.

proposed in section 4.2.2.2. I also propose that the TOPIC head should come with an [L] feature, but should not be specified with any more of the licensing/case bundle than that. The motivation for this is that left-dislocation can clearly involve a number of different category types:

- (35) a. [vp Love honey], it's what I like to do.
 b. [pp In the car], it's where I like to eat honey.

Because the TOPIC head can clearly license a number of category types, it should be specified with a feature to do so, but we would not expect it to be specified with any morphological case features, as TOPIC and FOCUS heads do not function primarily to fulfill the needs of DPs, specifically.

With these specifications in mind, we can now work through how the left-dislocated DP satisfies its requirements. First, as an unvalued feature, the unvalued [uL[case]] on the DP probes into its c-command domain. Here, it finds a MATCH with the TOPIC head, which is specified with an [L] feature. This MATCH is successful because MATCH is only evaluated at the root. This successful operation identifies the TOPIC head as a potential valuer of probe. Since MATCH was successful, VALUE then proceeds where the unvalued licensing/morphological case bundle on the DP evaluates whether or not its potential valuing goal is able to transfer its features to value the probe. Since the features on the probe are more specified than those on the potential goal, VALUE evaluates this as a failure. The failure to VALUE triggers the stripping of the features on the probe to its root feature, [L]. Because MATCH was successful, the newly stripped licensing/morphological case bundle on the DP is able to continue again to find a VALUE. This time, because the newly stripped probe is no longer more specified than the goal, VALUE is able to proceed successfully and the [L] feature on the TOPIC head goal is transferred to value the unvalued corresponding feature on the DP. Once this DP is spelled out, it is directed to insert the default case form – for English, accusative. The structure in (34) therefore produces the sentence in (33a).

4.2.3.4 Coordination

Coordination is another place where we find accusative DPs in English, despite no obvious source for the [ACC] features in the derivation. This fact, coupled with the observation that coordinated DPs in other languages are not accusative, but rather often⁹ align with whatever the default case for that particular language is, lead us to treat the examples in (36) as default environments:

(36) *Coordination*

- a. **Me** and **her** will go to the store.

I first want to make clear that I agree with Schütze (2001), among others I'm sure, that we should not propose that coordinators assign morphological case. This is for two reasons: (i) arguing for an analysis like that would in fact be quite ad hoc and (ii) doing so would require us to argue that coordinators both cross-linguistically and within a particular language aren't consistent in which morphological case they assign their conjuncts.

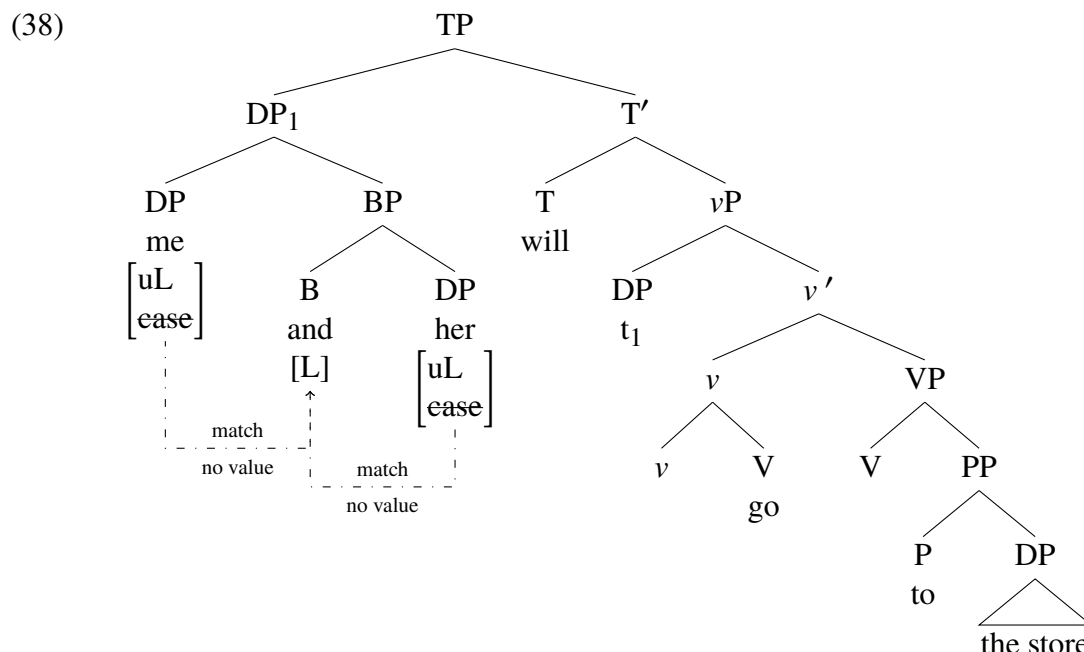
Instead, what I will argue here is that coordinators are not morphological case assigners at all, as discussed above – they do not come specified with any specific case features. What they are able to do though is license DPs. Importantly for the type of feature grounding I've suggested in this thesis, this ability is not restricted to just DPs, but is rather quite cross-categorial:

- (37) a. [DP Jim] and [DP John] will go to the store.
b. Jim will [_{VP} go to the store] and [_{VP} rent a movie].
c. The store is [_{PP} around the corner] and [_{PP} down the street].

Since coordinators are able to license a number of categories rather than just DPs, I argue that these are the kind of functional heads that will come specified with an [L] feature, but no other DP-specific featural structure beyond that. With this featural specification put in place, we now look at how the coordinated DPs in the examples in (36) come to receive accusative case by default.

⁹Not unsurprisingly, the coordination data is a bit more involved. Here I provide an account for the coordinated structures that do involve default case, but admit that the claims here should be tempered in some ways.

Below is an example of the sort of structure I will assume for coordinated DPs, following Munn (1993)'s BP adjunction analysis:



As DPs, both of the coordinated DPs come with the full licensing/morphological case bundle I've proposed: [uL[case]]. The larger coordinated DP, as a DP itself, comes with this specification as well. When each of the coordinated DPs probe, the first functional head each encounters is the coordinator *and*. As a reminder, the lower coordinated DP first probes its c-command domain, but following Carstens (2016) is able to continue to probe upward if nothing is found there, provided the derivation has not yet been spelled out. Since the functional head that each of these DPs finds first has an [L] feature that is capable of successfully MATCH-ing the unvalued licensing/morphological case bundle on the DP, MATCH is successful and VALUE is then attempted. As we saw in the examples with hanging topic/left-dislocation, VALUE between the more specified licensing/morphological case probes and the less specified [L] feature goal is unsuccessful, which triggers the deletion of any additional feature structure on the DPs, minus the root. These newly stripped [uL] features on the probes are then able to VALUE when they encounter the [L] feature on the coordinating head *and*. The resulting DPs are now valued with just their [L] feature which will

trigger the insertion of default case forms at spell out.¹⁰

One specific thing to note with these coordination examples is that I'm of course assuming that both DPs have access to the licensing feature on the coordinating head. I suggest that this isn't really problematic because once again, this licensing feature is independent from morphological case. It seems that the job of coordinating heads in the structure is to license and connect two syntactic objects. It's not unreasonable therefore to assume that both conjuncts have access to this licensing. I'm uncommitted at this point to whether this joint access is due to there actually being two independent licensing features on the coordinating head or whether the two conjuncts are simply just both able to access it due to it being a valued, rather than unvalued type of feature.

4.2.3.5 Gapping

Gapping is considered another syntactic environment where default case surfaces (Schütze, 2001).

(39) *Gapping*

- a. She will eat beans, **him** rice.
- b. For Mary to be the winner and **us** the losers is unfair!

For this discussion, I'm going to adopt the analysis of gapping proposed in Johnson (2009), shown in (40) below. There are three parts to his approach: (i) low coordination of the vPs (ii) heavy NP shift of their objects, and (iii) across the board movement of the verb phrases. Johnson assumes that when two vPs are coordinated, that coordination can trigger two separate processes: the rightward shift of the objects outside of their respective VPs and the subsequent across the board movement

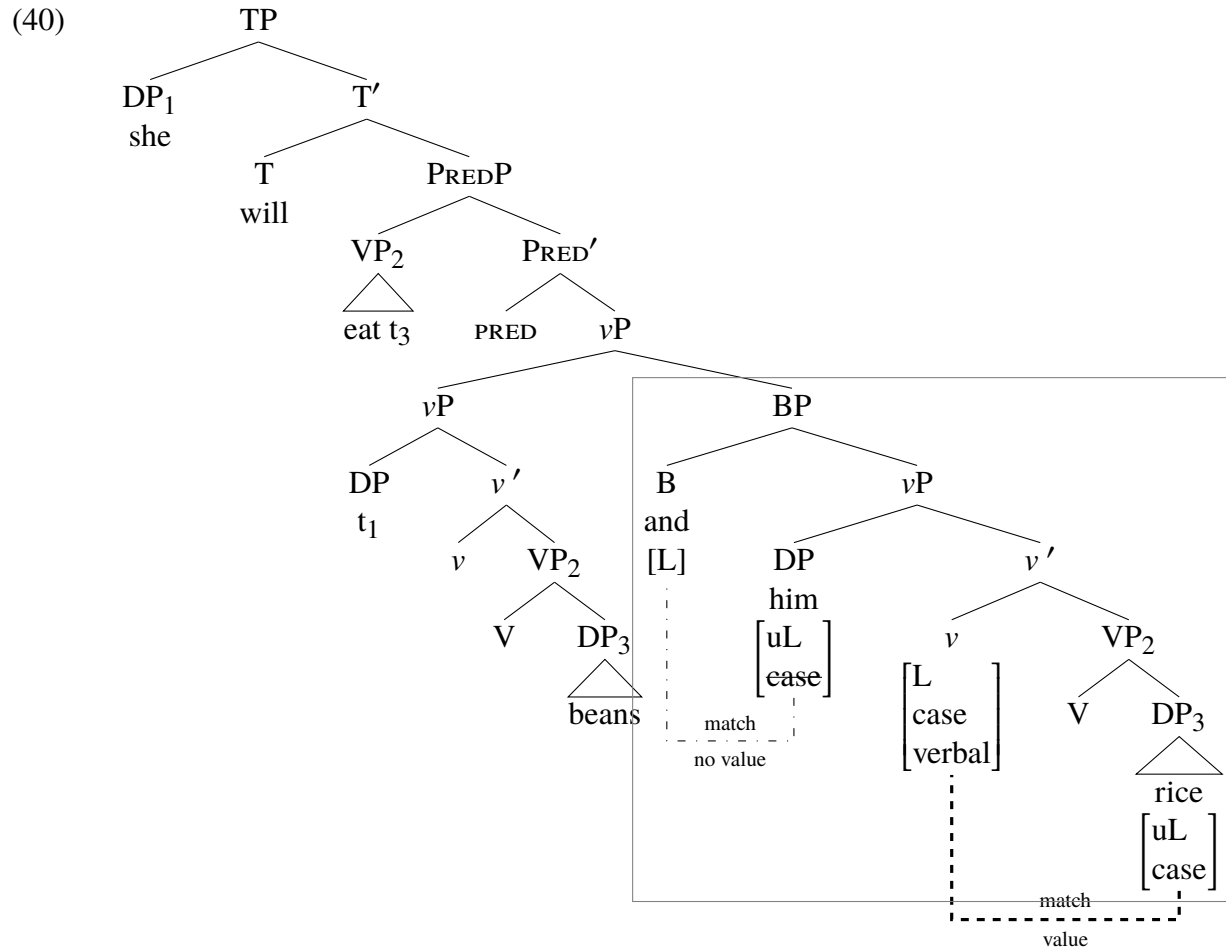
¹⁰Of course you may be wondering about the fact that in English, nominative forms are acceptable as well (ia). Schütze (2001) suggests that these might be instances of hypercorrection given that they're not consistently used across the paradigm (ib). If this is true, then we might be able to account for these situations where a prescriptive rule overrides the grammatical default. (see Sobin, 1997, for a proposal)

- (i) a. between you and I
- b. *between we and they

of those VPs to the specifier of a predicate phrase. The subject of the first ν P, as the highest DP in the structure, will be the one targeted by the EPP feature of T and will raise to subject position. The subject of the second ν P will remain in its original position. It is this position where it receives default case as there isn't a canonical accusative case assigner available to assign its case features.

To see how we can account for this using the case assignment system proposed in this chapter, let's focus on the boxed part of the tree in (40), the BP *and him rice*, to show how the DP *him* is spelled out as the accusative pronoun. Here, the object DP *rice* probes and finds a successful MATCH and VALUE with the accusative ν , as it has done in all of our canonical examples. This successful AGREE makes the features on the ν unavailable to other DP probes. The external argument of the ν P probes its c-command search domain, but because the features on the ν are unavailable, it continues its search upwards. It quickly finds a MATCH with the B head *and* and attempts to VALUE. Since VALUE here is unsuccessful because the features on the goal are less specified than those on the probe, VALUE fails and the features on the DP probe are stripped down to the root feature, [L]. VALUE then once again is attempted and is this time successful. The DP is therefore transferred to spell out with a [L] specification and the default case form for English, the accusative, is inserted.¹¹

¹¹One concern is how the features of the B head are available, despite a general understanding that once a head agrees, its features are no longer available to other probes. One thing we could say here is that the B head, due to its nature of licensing multiple conjuncts, is special in that its features *do* remain available, even after successful AGREE. Another worry that we might have is that by agreeing with the B head, in a way we are saying that the B head is licensing not only its conjuncts, but also DPs inside them. One way to avoid some of this discomfort is to say that coordinators cannot license something if that something is incomplete or ungrammatical in anyway. Since the verb in the coordinated ν P is a transitive verb, it must have an external argument. We could maybe say that the licensing of the external argument comes as part of the licensing of the ν P. Finally, if the idea floated in an earlier footnote that two-way dependencies are the only ones that can't participate in further dependencies winds up working, we could say that since the nominal isn't giving the coordinator anything in return, it's the type of dependency that doesn't freeze the licensing head.



4.2.3.6 ACC-ing gerunds

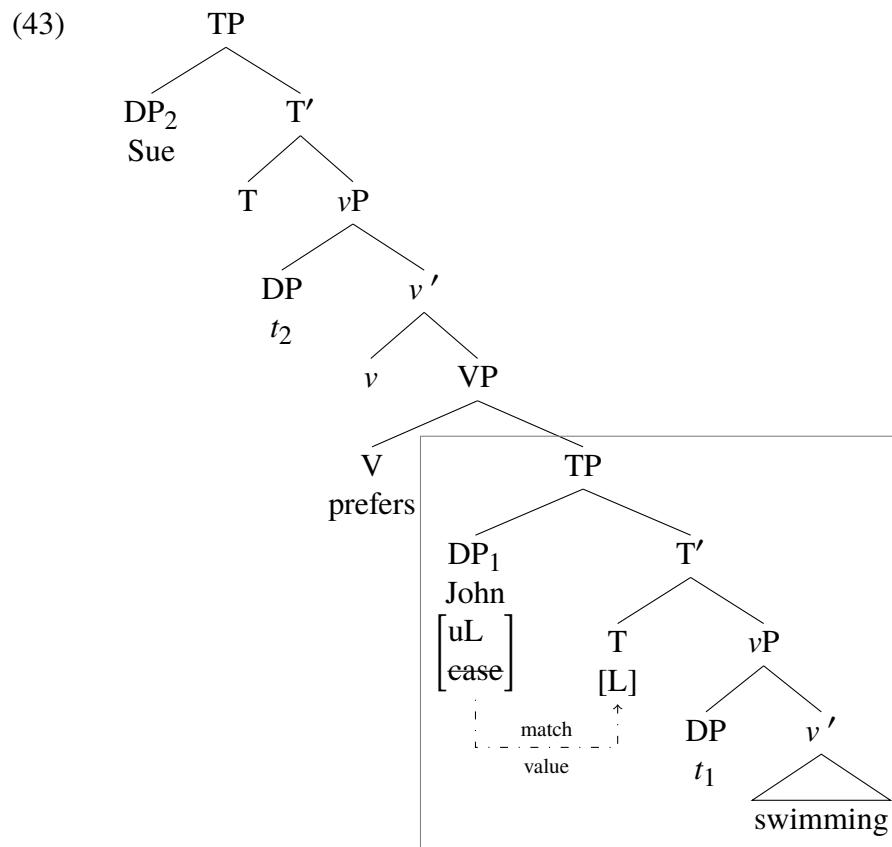
A potential extension that Schütze (2001) didn't detail (although he hinted at its possibility in (Schütze, 1997)) is to treat ACC-ing gerunds as yet another place where nominals receive default case (Abney, 1987; Horn, 1975; Milsark, 1988; Reuland, 1983).

- (41) a. **Her** revising the book is really helpful.
 b. Sue prefers **him** swimming. (Pires, 2007)

I adopt the analysis offered in (Pires, 2007), (See Pires, 2007, for further details) While both types of gerunds are clausal, there are many places where ACC-ing gerunds and poss-ing gerunds exhibit a difference in behavior: ACC-ing can appear with certain adverbs that are ungrammatical with poss-ing gerunds (42a)-(42b), ACC-ing are capable of licensing long-distance wh-extraction

(42c)-(42f), and ACC-ing gerunds can include an expletive while that option is ungrammatical for poss-ing gerunds (42g)-(42h). Given that ACC-ing gerunds do not seem to behave like poss-ing gerunds, Pires suggests that ACC-ing gerunds must have a different structure; proposing that ACC-ing gerunds are TPs and poss-ing acquire some sort of DP layer through the derivation. Of course, the T that heads the gerund must be somehow distinct from finite T all well as we get accusative case assigned inside the gerund rather than nominative. A structure for ACC-ing gerunds is shown in (43).

- (42)
- a. Mary probably being responsible for the accident, the attorney did not want to defend her.
 - b. *Mary's probably being responsible for the accident, the attorney did not want to defend her.
 - c. What did everyone imagine Fred singing?
 - d. *What did everyone imagine Fred's singing?
 - e. Who did you defend Bill inviting?
 - f. *Who did you defend Bill's inviting?
 - g. You may count on there being a lot of trouble tonight.
 - h. *You may count on there's being a lot of trouble tonight.



What I propose is that the gerundive T is specified as [L], but the motivations for its specification are a bit different than what we've seen so far. I argue that there are three 'flavors' of T: the normal finite T that comes with [L[case]], the non-finite T that comes with [∅], and the gerundive T that comes with [L]. We clearly can't ground this specification in the same way we have for the other functional heads with [L] because it's quite impossible to argue that this gerundive T licenses many types of categories. However, I think what might be interesting is if we tied the gerundive T's specification to finite T's specification. Viewing them together, we can characterize gerundive T as a sort of underspecified T and by extension, clausal gerunds a type of underspecified clause. They behave like clauses in many ways and so gerundive T has some of the normal capabilities, but not all of them – namely, it can't assign nominative case to its specifier. While clausal, ACC-ing gerunds do exhibit some nominal-like behavior. There are three positions which are associated with nominals where ACC-ing gerunds can appear: complement to V, complement to P, and subject position; the examples in (44) show this distribution. ACC-ing gerunds also pattern with nominals

with respect to case positions, being ungrammatical in passives and in raising structures (45). Pires proposes that what accounts for the nominal-like distribution of ACC-ing gerunds is that unlike finite T, the gerundive T has its own unvalued case feature, which needs to be valued. With respect to grounding the difference between gerundive T and finite T, it could be that the underspecification of gerundive T is somehow tied to this property.

- (44) a. Mary favored [him taking care of her land]..
 b. Sylvia wants to find a new house without [her helping her].
 c. [Her showing up at the game] was a surprise to everybody.
- (45) a. *Him was preferred [reading a book].
 b. [Him reading a book] was preferred.
 c. *It appears [him liking Mary].

With respect to the proposal itself, if gerundive T is specified as [L], then we can show that the subject of the gerund receives default case via having MATCH-ed, but not VALUE-d its case/licensing bundle. A nice benefit to this approach is that we do not have to propose that T assigns accusative case. All that's needed here is to say that the morphological case assigning function of T is lost, but its licensing function is not. In this circumstance, the default accusative case arises naturally from the system.

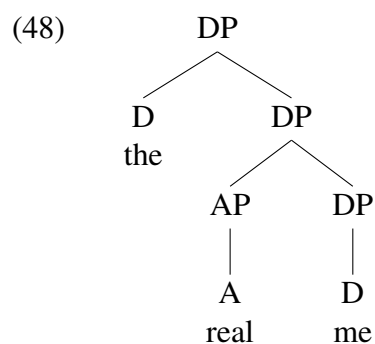
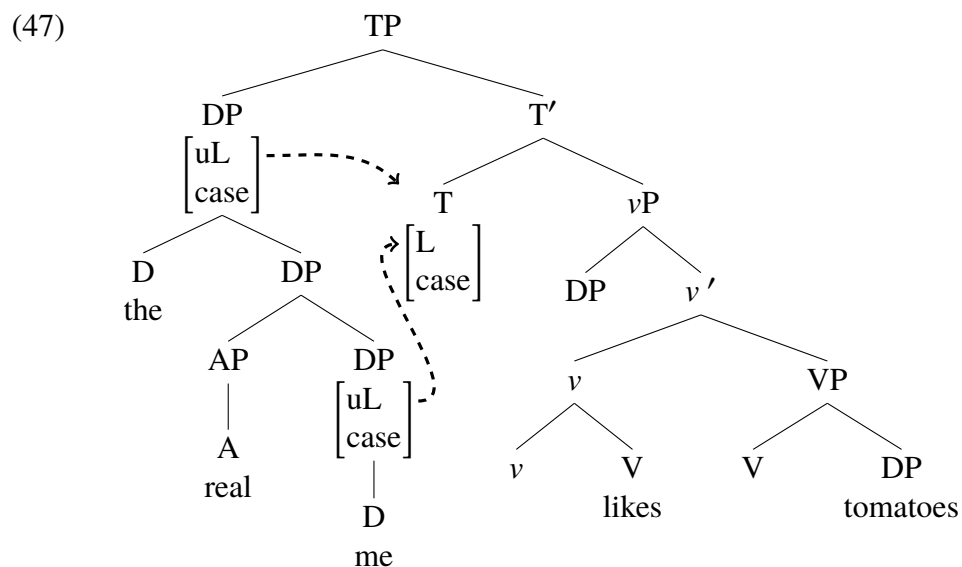
4.2.3.7 Modified Pronouns

Finally, we come to modified pronouns, the final default case environment from Schütze (2001) that I will discuss here. The examples in (46) show that accusative pronouns can often appear as smaller constituents of a larger DP, once again, despite there not being any obvious source for these features.

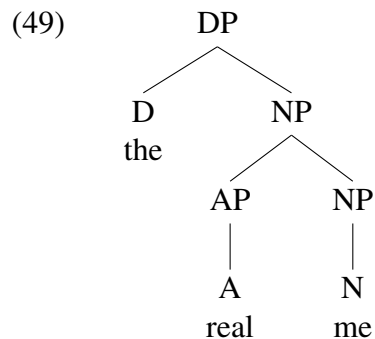
(46) *Modified Pronouns*

- a. Lucky **me**/***I** has to clean the toilets all day.
- b. The real **me**/***I** is finally emerging.

At face value, these examples seem to constitute the most difficult case for the account I'm advancing in this thesis. If the pronoun can't find a licenser within the DP it's a constituent of, it is not clear how we would prevent that pronoun from searching outside the DP and finding the same [NOM] that the larger DP would presumably receive from finite T, shown in (47). What this seems to mean is that my account requires there to be a source for licensing internal to the larger DP to avoid nominative case on the smaller constituent pronoun, *me*. Given a structure like (48), it's clear that there are not many (or any) real options.



One way to avoid this problem is to argue that the nominal *me* does not in fact have a licensing/morphological case feature bundle at all and its distribution is only governed by selection. How could this be possible? If the *me* in (46) were a noun, instead of a pronoun then we could make the argument that *me* doesn't come into the derivation with the same set of requirements that DPs do. *Me* is a regular noun, akin to a proper name and is simply syncretic with the accusative pronouns. Instead of (48), we would model these modified pronouns as (49):



One reason to suggest that the nominals in (46) are not the same as the canonical pronouns is that they trigger third person agreement, rather than the first person agreement that bare pronouns would trigger.

(50) *3rd person agreement*

- a. The real me is/*am emerging.
- b. I *is/am emerging.

Schütze (2001)

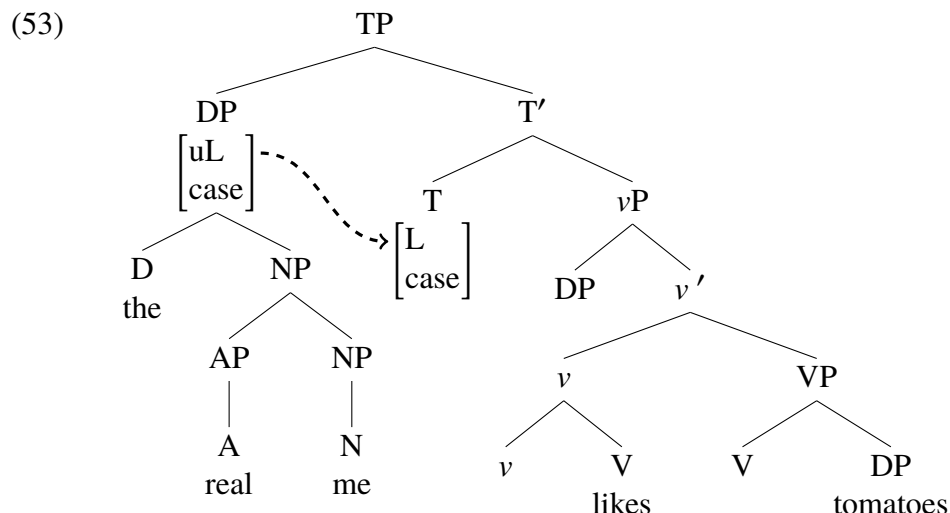
This distinction indicates that these two types of pronouns are different in some way, and I argue that the difference is that they are not of the same category type. Schütze (2001) provides evidence from Italian that supports a *me*-as-noun type of analysis as well:

- (51)
- a. il vero me stesso
the real my self
 - b. ??il vero me
the real me

- c. **me (stesso) vero
me (self) real
- (52) a. il vero Paolo
the real Paul
- b. **Paolo vero
Paul real

The data in (51) and (52) show two things: (i) a parallel between modified pronouns and proper names, lending support to a *me*-as-noun type of analysis, and (ii) that in the proper name subset of nominals, the typical N-to-D movement common in Italian is actually unavailable. If modified pronouns are in fact nouns, then we have to have an explanation for how they do not covertly move to D and receive nominative anyways. The fact that modified pronouns do not move in a language that has overt N-to-D movement supports that these modified pronouns do in fact stay in their merged N position. The agreement data in (50) paired with analogous proper name data in (51) and (52) provide reason to draw a distinction between typical bare pronouns and these modified pronouns.

What's convenient about assuming that the modified pronouns are nouns is that we have a built in explanation for how they avoid receiving nominative case – they are not DPs, so they do not need case at all. Each of these nominals' forms is not an actual example of accusative case, but rather they are regular nouns that have been reanalyzed to some extent and are simply syncretic with the accusative forms in English. As nouns, they do not come specified with the licensing/morphological case bundle that I've argued for in this paper and in this way, the examples in (46) would actually not be considered default case environments under my analysis at all.



What (53) shows is that the nominal *me*, as a noun, does not come with any of the licensing/morphological case bundle. It therefore has no requirements beyond selection and doesn't probe at all. The larger DP that it's a constituent of, *the real me*; however does come specified with a licensing/morphological case bundle, like all DPs. It does have a set of requirements it needs to meet and does so by canonical probing into its c-command domain. As usual, it will find a **MATCH** and a **VALUE** from the featural specification of the finite T and it therefore receives nominative case. In English, nominative case isn't spelled out on this larger DP.

4.3 Evaluating Our Options

We've now argued that default case can be accounted for through the adoption of a **MATCH/VALUE** type of agreement system proposed by Béjar (2003), but we've not yet considered whether or not we *should* account for default case this way. This section outlines a few empirical reasons for why this type of approach has enough promise to be seriously considered. The goal here is to show that default case does not *necessitate* the rejection of the role of case in nominal licensing, nor the abandonment of an **AGREE**-based approach to case valuation. While this may appear far too modest a claim, the existence of defaults and the problems they've introduced has been framed as so serious a problem that its solution requires drastic reconfigurations of basic, long-held assumptions. Showing that smaller modifications to the system are available removes the severity of the call to

adopt the departures. Whether or not one wants to adopt the system proposed here is of course a separate issue, and one that I'll try to address here. However, given the breadth of morphological facts that a theory of case must account for, I simply will not be able to convince the reader that dependent case theory is dead, or should be dead. What I do hope to do in this section is to bolster the attractiveness of this other available option that is more in line with the decades long program of classical case theory.

4.3.1 Some Problems for Dependent Case Models

This section outlines a few types of data that raise serious issues for dependent case models. We can group these examples into two groups by the issue they pose for the dependent case model. In section 4.3.1.1 we'll see data that make it difficult to argue that the assignment of accusative case is dependent on the existence of other nominals and in section 4.3.1.2, we'll see examples from the default case set of environments that make the wrong predictions for case assignment.

4.3.1.1 Sole Accusative Arguments

Because dependent case theory frames the assignment of accusative case as dependent on the presence of another nominal in a given domain, it makes the prediction that sole arguments of predicates should not be able to receive it, at least without proposing an 'invisible' sort of nominal in the structure. Kučerová (2012) makes exactly this point and provides data from Polish and Ukrainian of exactly these instances. There is a construction in these languages called the *-no/-to* construction, shown below in (54)-(55).

(54) *Polish*

- | | | | | | |
|----|------------------------------|-----------------|-------------|----------------|-----|
| a. | Pies | był/został | zabity | przez samochód | |
| | dog.M.SG.NOM | was/stayed.M.SG | killed.M.SG | by | car |
| | 'A dog was killed by a car.' | | | | |
| | CANONICAL PASSIVE | | | | |
| b. | Psa | zabito | | | |
| | dog.M.SG.ACC | killed.N.SG | | | |
| | 'A dog was killed.' | | | | |
| | NT | | | | |

(55) *Ukrainian*

- a. *Žinky* *buly* *vbyty*
 woman.NOM.F.PL WAS.F.PL killed.F.PL
 ‘(The) women were killed. CANONICAL PASSIVE
- b. *Žinok* *bulo* *vbyto*
 woman.ACC.F.PL were.N.SG killed.N.SG
 ‘(The) women were killed. NT

This construction involves the internal argument being marked as accusative, rather than the more expected nominative case, (56).

- (56) a. *Psa* *zabito*
 dog.m.sg.acc killed.n.sg
 ‘A/The dog was killed.
- b. **Pies* *zabito*
 dog.nom.m.sg killed.n.sg

What makes this sort of data difficult for the dependent case theory to account for is that there is no obvious way for the dependent case algorithm to assign accusative case to the nominal in question because there is simply no other nominal that c-commands it. She argues that since the *-no/-to* construction can be formed with unaccusatives, raising verbs, and modal verbs, shown in (57), that there really is an absence of an external argument, even one that is not overt. Since accusative case is dependent on the presence of that additional nominal by definition, this data constitutes a problematic example for this theory to capture.

- (57) a. *Balon* *rozerwano*
 balloon.ACC pierced.N.SG.PPP
 ‘The balloon was pierced.’
- b. *Zdawano się* *nas nie zauważać*
 seem.IMP REFL US not notice.INF
 ‘They seemed not to be noticing us.’
- c. *Musiano to* *vwykonać, bo* *zbliżał się* *termin*
 must.NT this do.INF because approached REFL deadline
 ‘(They) had to do this, because the deadline was approaching.

All is not easy in the Agree-based system either, but the Agree-based system does provide a bit

more leeway for it to work. Traditionally, the issue is that as a v that does not have an external argument, it is unable to assign accusative case. This is the familiar Burzio's Generalization. What Kučerová argues is that we've misunderstood what is responsible for 'bestowing' the ability of v to assign case. It's not the presence or absence of an external argument that allows v to assign case, but rather whether or not the v structure is extended that is the relevant property. Of course, one way this extension can happen is if an external argument is merged in the structure, and in this way Kučerová captures the initial Burzio intuitions. What this proposal does however is it provides an additional set of instances where v case assign accusative case. For her, the *-no/to* construction involves the *have*-perfect, which she argues extends the projection in a way that allows for the assignment of v .

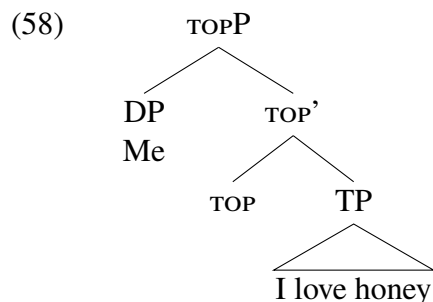
The proposal for the *-no/to* construction is of course not without issue, but the set of possible explanations for how the accusative case ends up on the singular argument conflict less with the version of the system she adopts than it would for dependent case theory. I suggest that while one may have reservations about adopting this particular story, all one needs to do if adopting an Agree-based approach is to propose an understanding for how v , which normally has case assignment abilities anyway, assigns accusative case, despite the expectation that there's some property blocking this ability. To accomplish something similar while adopting dependent case theory requires one to either suspend the definition of accusative case altogether and claim that its assignment is not always dependent on there being another nominal present or it must propose that, despite evidence to the contrary, there is a null nominal in the structure that the dependent case mechanism can be sensitive to.

4.3.1.2 Dependent Case Theory and Default Case

One thing that that makes the default environments especially interesting with respect to evaluating dependent case theory is that they all test the distinction between unmarked and default environments, a distinction that we saw in chapter 2 is not easily maintained in modern versions of the theory. Unmarked cases in a dependent case system are like default cases in that they are cases that

can be assigned without being dependent on the existence of another nominal in a given domain. What distinguishes them informally from defaults is that unmarked cases are context-sensitive while defaults are not. As a reminder, the two main unmarked cases are nominative, which is assigned by the grammar when a nominal does not otherwise receive case *and is typically in the domain of TP spell out* and genitive, which is assigned by the grammar when a nominal does not otherwise receive case *and is typically in the domain of DP/NP spell out*. By comparison, default case is assumed to be the case that is assigned when a nominal is unable to receive either a lexical case, a dependent case via the dependent case algorithm, or the unmarked case. Because the conditions under which unmarked case is assigned are already quite default-like in nature, this essentially means that default case is predicted to only show up where the unmarked case cannot apply – outside the spell out domains of TP and DP/NP. Where modeling defaults in dependent case theory sees difficulty is that it's certainly much easier to argue that there are a set of nominals that are not governed and are thus outside any sort of governing domain, as was true in earlier versions (Marantz, 1991). It's much harder however to make an argument that the nominals in question are not contained or otherwise present in a TP or DP/NP context.

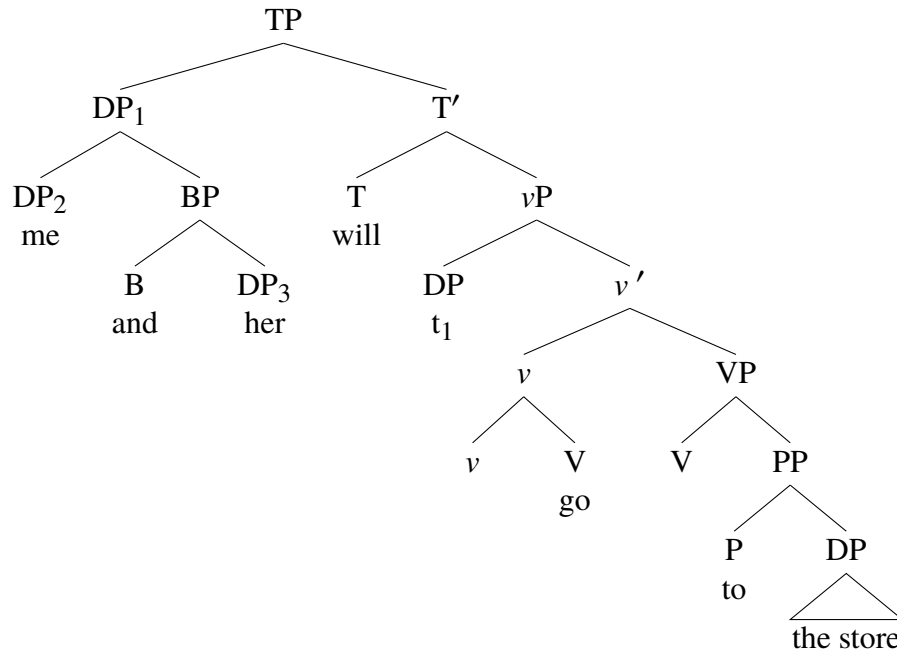
To see the difficulty, let's first look an example of a default case that does not challenge this idea: the default environment of hanging topics and left-dislocation, the structure repeated below in (58). Here, the nominal in question does not receive accusative case because it is not c-commanded by another nominal that also does not have case. It also doesn't appear to qualify for the unmarked case as it is in a position that is not within the relevant TP domain. It therefore remains completely caseless at the end of the derivation. We can therefore assume that at spell out, the grammar assigns default case to that nominal; accusative for languages like English and nominative for others.



What allowed the nominal in (58) to avoid getting the unmarked case was that it existed in a position outside the domain that defines the assignment of unmarked case. Where dependent case would run into trouble is if we found nominals that received default case, rather than unmarked case, in positions that were not outside these unmarked assigning domains. I'll show here that two of the default case environments are exactly these types of positions: coordinated DPs and gapping environments. As a quick reminder, the motivation for classifying these environments as default case environments rather than examples of unmarked case is the cross-linguistic variation we see in them. While some languages mark these nominals with nominative case – and would thus be indistinguishable from unmarked case. Other languages mark the nominals in those same positions with accusative. This indicates a last resort style default mechanism.

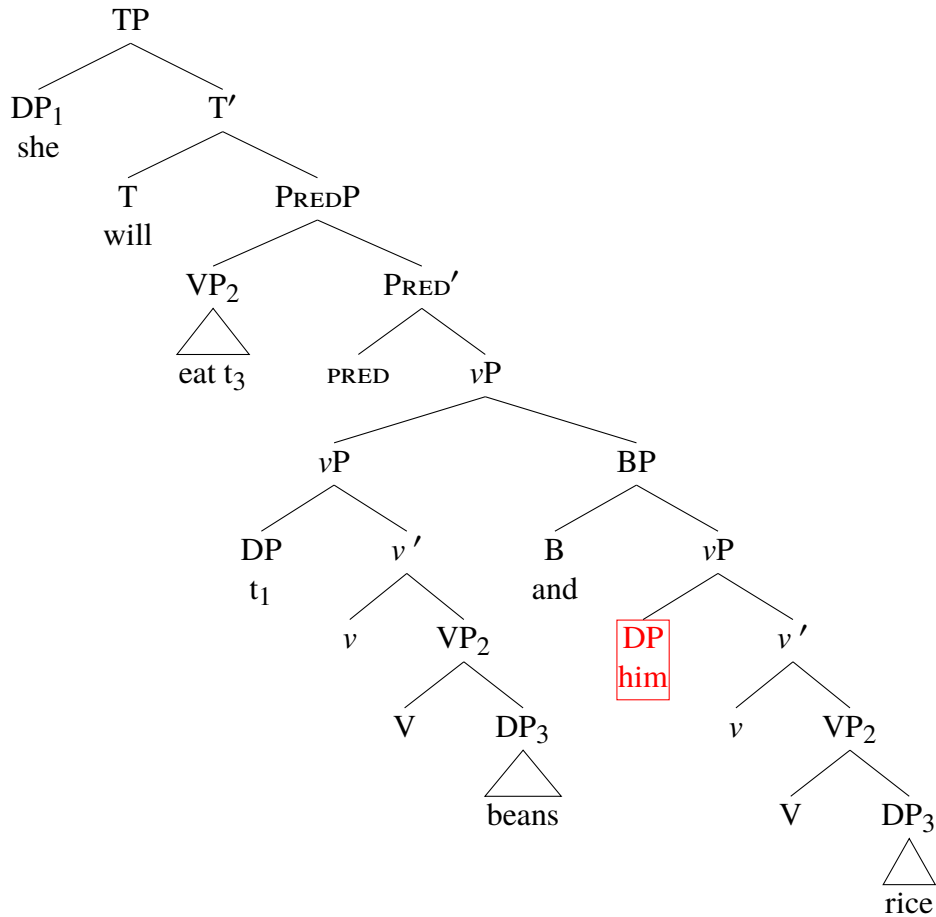
Our discussion regarding coordinated nominals is interesting because there are two nominals to discuss, rather than one. We'll discuss them in turn by examining the structure in (59). DP₃ is in a position where it is c-commanded by another nominal. It could avoid being assigned accusative case by the dependent case mechanism if we assume that the DP₁ domain that contains it serves as a barrier of sorts from the larger TP domain where the dependent case mechanism applies. If it avoids the assignment of dependent case, it is then evaluated for assignment of unmarked case, nominative if in a TP domain, genitive if in a DP/NP domain. It seems that it would be difficult to make an argument that the DP₃ nominal is in neither of these domains. Default case then would not be able to be assigned because the unmarked case would have taken precedence. DP₂ avoids getting dependent case because it, unlike DP₃ is *not* c-commanded by another DP in the TP domain. Like DP₃, it however does have trouble avoiding being assigned unmarked case because it also is in a position that is difficult to argue isn't in the unmarked domain.

(59)

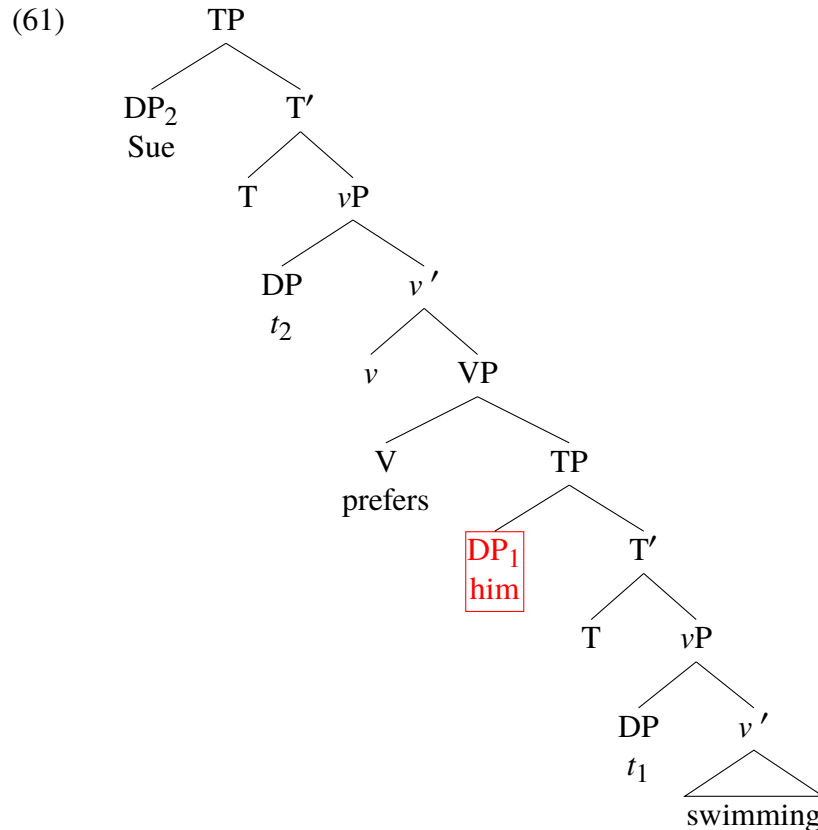


Gapping environments provide a similar argument: that there are default case receiving nominals in positions that would be difficult to claim aren't in either of the unmarked case assigning domains. The assumed structure for gapping environments is shown below in (60), with the relevant nominal highlighted in red. As with the coordinated nominals, it is difficult to make the claim that the highlighted nominal does not exist in the TP spell out domain. Now, one could argue here in the English examples that maybe gapping environments aren't actually examples of default case, but rather are more canonical accusative case examples. Notice that *him* is c-commanded by another nominal in the spell out domain. We'd therefore expect that nominal to receive the dependent accusative case. For English, there isn't an obvious reason to reject that proposal since the default case is accusative and a distinction cannot be made. However to the extent that one considers this a default environment where other languages would instead mark the nominal with nominative case, this becomes an issue.

(60)



Finally, to the extent that one adopts the proposal of ACC-ing gerunds argued for in Pires (2007), this environment offers difficulty for dependent case as well. The highlighted DP appears in the accusative case but there is not other DP that c-commands it within its own spell out domain. Once the matrix *v* merges into the structure, it should trigger the spell out of the embedded TP spell out domain. Because within this domain, there is no other nominal besides DP₁, the dependent case model would predict the highlighted DP to surface in the unmarked nominative form.



To summarize, what causes problems for modern dependent case theory is that there exist a number of default nominals that exist in positions where we'd predict the unmarked case to apply rather than the observed default. AGREE-based models more easily account for default case because the burden for escaping the case assignment process is easier to meet. Defaults in a system like the one proposed in section 4.2 simply have to exist in a position where they cannot form a relationship with a case assigning functional head. Defaults in a configurational based case system must instead exist in positions that are outside clause building domains like TP (and DP), something that is much harder to argue.

4.3.2 Final Remarks

There are a host of other questions that come out of the discussion had in this chapter. Some of those will be speculatively explored in the next chapter. What I hope this chapter has accomplished is the following: we've provided an alternative account to modeling default case in the grammar,

largely maintaining the traditional assumptions about the role of Case in regulating DP distribution. We followed the traditional understanding that licensing and morphological case are independent; but by modeling the features responsible for each as having a hierarchical relationship, we've been able to formally encode the notion that while licensing and case are separate, they are related to one another. This is an interesting finding because data like default case data appeared to require the abandonment of these basic theoretical assumptions. The alternative proposed in this chapter shows that we can in fact reconcile the problematic default case data in a framework where the failure to receive case can still rule derivations ungrammatical.

This chapter has also contributed a novel system of case features that relied heavily on intuitions others have made about case syncretism patterns and the hierarchical structure they suggest. Arguing for this hierarchical feature system allowed us to extend the MATCH/VALUE approach to agreement that captured so many varied patterns in the φ -agreement domain, most important – the ability for the operation to fail, solving a similar issue in a different domain. Being able to account for similar types of syntactic failure while employing the same set of operations is likely a benefit of the proposal offered here.

One part of the licensing/morphological case specifications on functional heads that I've left aside is the question of to what extent do we think that other syntactic categories (like PPs, vPs, etc) also come with a licensing feature. What motivated DPs having this sort of feature bundle was that they seemed to have requirements beyond a simple selection feature. Selection alone isn't enough to license DPs because they appear to need some sort of confirmation beyond this selection to appear in the positions where they are allowed. It does not seem to be the case however that other category types have a requirement like this. As far as we know, selection alone *is* enough to account for the distribution of PPs, vPs, etc. Since selection alone appears to be enough, then proposing that these categories come with an additional head that does the work of licensing these categories seems unnecessary.

Note that selection alone can't be enough to handle DP distribution. If we could reduce the [L] feature proposed here to something like a D feature or a DP feature, then we'd be unable to account

for the non-finite subject position examples that are ungrammatical, as there isn't a convincing way to differentiate finite T from non-finite T based on some version of an EPP/DP feature. I'm certainly not attempting to push these issues aside, as they are quite central to the problems surrounding the needs of DPs. This should be a familiar discussion as it's essentially the Case versus EPP debate that so many researchers have devoted time to (see McFadden (2004) for a review of the relevant literature).

What I'd like the big take-away to be from this chapter is that the existence of defaults in the case domain does not require us to abandon ship, as has been previously argued. By nailing down what DPs are looking for and by being specific in how those needs are encoded into a feature structure, we've been able to make some progress on accounting for how defaults could surface in the realm of case, while maintaining its role in determining grammaticality. I've paired this argument with some comments on instances where dependent case theory might have some trouble. I leave speculation about what these alternatives mean for the bigger picture at large in the next chapter.

CHAPTER 5

CONCLUSIONS

5.1 Introduction

With the arguments and proposal laid out, I offer a quick summary of what we've done before some final comments about what this all might mean. We saw how the existence of defaults raises some interesting and deep problems for our understanding of how we want to model the syntactic framework. The crux of the issue is two fold: how is it that defaults can surface in a system that should categorically rule them out and how can we constrain the mechanism responsible for their production? Three main questions guided our discussion:

- (i) How does the grammar *produce* defaults?
- (ii) How does the grammar *constrain* the production of defaults?
- (iii) What can an understanding of syntactic defaults tell us about the how the syntax can encode underspecification?

To address these issues, researchers have proposed the abandonment of one of the central tenets of the syntactic system, proposing in various ways that the failure to value features is not fatal to the derivation. This quite intuitively solves the problem raised by question (i). If features are allowed to survive to the interfaces without having been valued, then the production of defaults is easily accounted for. We saw a solution of this type in the case domain, where the dependent case proposal built defaults directly into the system by claiming that case features can remain unvalued. This subsequently requires the adoption of a separation between case and licensing given case's central syntactic role in regulating nominal distribution. We also discussed a solution in this vein proposed in the φ -agreement domain with Preminger's (2014) obligatory operations model whereby

operations are triggered by their structural descriptions and inherent need to apply, rather than a need to fulfill the requirements of various syntactic objects.

5.2 The Lifeboats are Headed in the Wrong Direction

In chapters 2 and 3 I presented a number of arguments for why we should be hesitant to jump ship, so to speak, and pursue this general approach. With respect to dependent case theory, I showed that its modern instantiation violates the Inclusiveness Condition in a way that makes the origins of case features – and by extension, their assignment – unclear. This raises questions about whether or not dependent case can truly constitute a real system as opposed to a clever way to describe morphological patterns. Related to that point is the conceptual inconsistency that it indicates. We saw that even under strict dependent case assumptions, case was modeled as the reflection of a number of different syntactic dependencies, undermining the status of case as a *system*. This likely has negative repercussions for acquisition as well. Empirically, we saw that the difficulty in distinguishing between unmarked and default cases raises issues for languages whose default is accusative, rather than nominative. Likewise, intransitive clauses whose sole argument is marked with accusative case appear to constitute serious problems for the model without any obvious solution. When we expand our purview to how dependent case fits in to the system of dependency establishment more broadly, we realize that dependent case isn't just an alternative mechanism; it is an additional one. To the extent that we wish to reduce the number of operations UG has access to, an AGREE based model that can capture both φ -agreement dependencies and case dependencies through the same mechanism is more parsimonious. And finally, the adoption of dependent case theory requires that we abandon the long held assumption that case plays a role in regulating the distribution of nominals. This separation of case from licensing requires that we propose an alternative understanding of the data case has been understood to account for for the past 40 years. The discussion in chapter 2 showed that the alternatives proposed have not yet met the high standard expected for such a central syntactic component.

In the φ -agreement domain, we discovered that when the obligatory operations model operates

over more complicated data, serious issues with respect to the overapplication of defaults arise. Languages for which there are second cycle effects, for example, show evidence for a third outcome of agreement – in addition to success and failure. This third outcome is distinct from defaults and the obligatory operations model is unable to model that distinction because it only differentiates between the success of an operation and the failure of one. This has the result of the grammar being satisfied with failure too early and subsequently inserting defaults where we should be getting second cycle morphology instead. Languages that have more specified probes will also cause issues for dative intervention as the obligatory operations model doesn't have a way to ensure visibility of underspecified nominals. On the conceptual side, it requires simultaneous immediacy and delay, which while not impossible to reconcile are particularly difficult in this domain. Furthermore, I made the argument that obligatory operations only makes sense if we can adopt it framework wide. Otherwise, it is unclear what benefit we gain in adopting it for such a narrow range of phenomena, especially when a framework-wide alternative is available.

Because the logic of using both dependent case and obligatory operations to provide answers to the questions at the start of this thesis is the same, it is perhaps not surprising that their implications are quite similar as well. Both proposals produce defaults by relaxing the rules that encode ungrammaticality. If having an unvalued feature at spell out is problematic, then removing the offense is a way to get defaults to surface. However, while each of the approaches presented was able to get the system to produce defaults, neither was able to constrain that production in ways that prevented the overapplication of defaults – the more theoretically interesting puzzle. Because they are unable to solve the second, and more far reaching, half of the default problem – while also completely upending the system we've built over the past 25 years of work in the Minimalist program – I argue that these lifeboats, intended to save us from the default 'fire on board', veer us off course.

5.3 Putting out the Fire Allows us to Maintain the Course

Of course, jumping into the lifeboats is unavoidable if there's no way to put out the fire in the first place. I've argued in this thesis for modest modifications to the standard set of assumptions that allow us to stay in our boat and maintain the course. In chapter 4 I argued that we can account for the production of defaults while maintaining that the failure to value features induces ungrammaticality by extending the theory of agreement proposed in (Béjar, 2003). This model is largely standard with one major modification: it argues that the two operations behind agree are sensitive to the inherent hierarchical relationships between the individual features that participate. The separation of the feature-sensitive operations introduces a third outcome of AGREE, one that we can exploit to account for the existence of defaults. The third outcome itself is how defaults are produced, but the presence of that third outcome *alongside* the failure-induces-ungrammaticality outcome allows us to constrain that production in ways that the previous approaches couldn't.

I also proposed a novel system of case features based on a number of previously noted intuitions that interacts in interesting ways with the two agreement operations. The result is a model of case valuation that maintains case's standard role in regulating nominal licensing while also allowing for the appearance of default case forms in restricted environments. What is really attractive about this proposal is that it addresses the default issues with surprisingly few modifications to the framework. Given the far reaching problematic implications that the drastic theoretical departures discussed in chapter 2 and chapter 3 invited, this is a welcome result.

5.4 What Have We Learned

Perhaps surprisingly, we learned that syntactic defaults actually *can* be captured in a framework that encodes grammatical requirements through the failure to value features *if* we allow the AGREE operations to be sensitive to the inherent hierarchical relationships that hold between features. Importantly, adopting the separation of AGREE into two operations reveals a third outcome of agreement, allowing us to both produce and constrain the appearance of defaults. Less surprising is the conclusion that by simply allowing features to remain unvalued, we've opened a Pandora's

box with respect to the framework at large.

With respect to the three questions posed at the start of the thesis, we can now provide some answers:

- (i) Defaults are produced when `MATCH` succeeds, but `VALUE` fails.
- (ii) The production of defaults is constrained by the existence of this third outcome *alongside* assumptions that failure to `AGREE` completely still induces ungrammaticality.
- (iii) Inherent hierarchical structure in feature systems allows for underspecification in the syntax with respect to which features are specified on which objects. Underspecification can also can be represented through the probe modification that results from the failure to `VALUE`, much like the Impoverishment operation introduced in chapter 1.

We learned that failure in the syntax does not lead to a singular set of outcomes. Rather, the grammar can use failure to trigger second applications of operations, defaults, strategies for reconciling conjunct features, and of course ungrammaticality. This wide range of outcomes suggests that unvalued features – which can serve as a sort of derivation pacer – do real theoretical work and cannot be removed as easily as one might hope.

We learned that case features also have hierarchical structure and we were able to understand the relationship between case and licensing in a new way. We've shown that the two concepts are independent, but related via entailment. This allows us to encode both their correlations and their mismatches.

The extension of `MATCH/VALUE` into the domain of case showed us that the separation of `AGREE` into two operations only makes different predictions when the probes are not flat. Not only is this a welcome result, but it makes some predictions about which feature categories have access to defaults. Since defaults are the reflex of a successful `MATCH` and a failed `VALUE`– which can only happen when a probe is highly specified – we predict only feature categories with hierarchical organization to be capable of producing default forms.

Finally, the proposal contributes to the discussion on the relationship between case and φ -agreement. Through this exercise, I've proposed that the two are separate dependencies, established via separate probes. However, they are both established via the same set of operations and in a large number of instances, involve the same participants. I suggest that this is why we often see such a strong correlation between the two. Their independence, however, is what can explain why they don't always match up. Like the relationship between abstract Case and morphological case, the relationship between case and φ -agreement shows they are separate, but related.

As I'm sure is often true, this thesis has raised far more questions than it has answered. I hope to have shone a little light on a small piece of an important issue and feel grateful to have gotten the chance to be part of the conversation.

REFERENCES

REFERENCES

- Abney, S. (1987). *The English noun phrase in its sentential aspect* (Unpublished doctoral dissertation). MIT, Cambridge, MA.
- Abondolo, D. (1982). *Verb paradigm in Erza Mordvinian*. Folia Slavica.
- Ackema, P., & Neeleman, A. (2017). Default person versus default number in agreement. In M. Reeve, L. Franco, & M. Moreno (Eds.), *Non-local dependencies in the nominal and verbal domain*. Oxford University Press.
- Adger, D. (2003). *Core syntax*. Oxford University Press.
- Adger, D., & Harbour, D. (2007). Syntax and syncretisms of the person case constraint. *Syntax*, 10(1), 2-37.
- Andrews, A. (1982). The representation of Case in modern Icelandic. In J. Bresnan (Ed.), *The mental representation of grammatical relations* (p. 427-503). MIT Press.
- Aronson, H. (1989). *Georgian: A reading grammar*. Columbus, OH: Slavica.
- Austin, J. (2012). The case-agreement hierarchy in acquisition: Evidence from children learning Basque. *Lingua*, 122(3), 289-302.
- Baerman, M., Brown, D., & Corbett, G. G. (2005). *The syntax-morphology interface* (Vol. 109). Cambridge University Press.
- Baker, M. C. (2008a). The macroparameter in a microparametric world. In T. Biberauer (Ed.), *The limits of syntactic variation* (p. 351-374). John Benjamins Publishing.
- Baker, M. C. (2008b). *The syntax of agreement and concord*. Cambridge University Press.
- Baker, M. C. (2012a). "obliqueness" as a component of argument structure in Amharic. In M. C. Cuervo & Y. Roberge (Eds.), *The end of argument structure?* (p. 43-74). Emerald.
- Baker, M. C. (2012b). On the relationship of object agreement and accusative case: Evidence from Amharic. *Linguistic Inquiry*, 43(2), 255-274.
- Baker, M. C. (2015). *Case: its principles and its parameters*. Cambridge University Press.
- Baker, M. C., Johnson, K., & Roberts, I. (1989). Passive arguments raised. *Linguistic Inquiry*, 20(2), 219-251.

- Baker, M. C., & Vinokurova, N. (2010). Two modalities of case assignment: Case in Sakha. *Natural Language and Linguistic Theory*, 28, 593-642.
- Beatty, J. (1974). *Mohawk morphology*. University of Northern Colorado, Museum of Anthropology.
- Béjar, S. (2003). *Phi-syntax: A theory of agreement* (Unpublished doctoral dissertation). University of Toronto.
- Béjar, S., & Rezac, M. (2003). Person licensing and the derivation of PCC effects. In A. T. Pérez-Leroux & Y. Roberge (Eds.), *Romance linguistics: Theory and acquisition* (p. 49-62). John Benjamins Publishing.
- Béjar, S., & Rezac, M. (2009). Cyclic agree. *Linguistic Inquiry*, 40(1), 35-73.
- Bhatt, R. (2005). Long distance agreement in Hindi-Urdu. *Natural Language and Linguistic Theory*, 23, 757-807.
- Bhatt, R., & Walkow, M. (2013). Locating agreement in grammar: an argument from agreement in conjunctions. *Natural Language and Linguistic Theory*, 31(4), 951-1013.
- Bittner, M., & Hale, K. (1996). The structural determination of case and agreement. *Linguistic Inquiry*, 27(1), 1-68.
- Blake, B. (2001). *Case*. Cambridge University Press.
- Bobaljik, J. D. (2008). Where's phi? agreement as a postsyntactic operation. In S. Béjar, D. Adger, & D. Harbour (Eds.), *Phi theory: Phi-features across modules and interfaces* (Vol. 16). Oxford University Press.
- Borer, H. (1984). *Parametric syntax*. Foris.
- Bright, W. (1957). *The Karok language*. University of California Press.
- Burzio, L. (1986). *Italian syntax*. Dordrecht: Reidel.
- Caha, P. (2009). *The nanosyntax of case* (Unpublished doctoral dissertation). University of Tromsø.
- Calabrese, A. (1998). Some remarks on the Latin case system and its development in Romance. In J. Lema & E. Trevino (Eds.), *Theoretical advances on Romance languages* (p. 71-126). John Benjamins Publishing.
- Carstens, V. (2016). Delayed valuation: a reanalysis of goal features, “upward” complementizer agreement, and the mechanics of case. *Syntax*, 19(1), 1-42.

- Chomsky, N. (1973). Conditions on transformations. In S. Anderson & P. Kiparsky (Eds.), *Festschrift for Morris Halle*. Holt, Rinehart and Winston.
- Chomsky, N. (1980). On binding. *Linguistic Inquiry*, 11, 47-103.
- Chomsky, N. (1981). *Lectures on government and binding: The Pisa lectures* (No. 9). Walter de Gruyter.
- Chomsky, N. (1982). *Some concepts and consequences of the theory of government and binding*. Cambridge, MA: MIT Press.
- Chomsky, N. (1995). *The minimalist program* (Vol. 28; S. J. Keyser, Ed.). MIT Press.
- Chomsky, N. (2000). Minimalist inquiries: the framework. In R. Martin, D. Michaels, & J. Uriagereka (Eds.), *Step by step: Essays on minimalist syntax in honor of Howard Lasnik* (p. 89-155). MIT Press.
- Chomsky, N. (2001). Derivation by phase. In M. Kenstowicz (Ed.), *Ken Hale: a life in language* (p. 1-52). MIT Press.
- Chomsky, N. (2005). Three factors in language design. *Linguistic Inquiry*, 36(1), 1-22.
- Culicover, P. W. (1997). *Principles and parameters: An introduction to syntactic theory*. Oxford University Press.
- Deal, A. R. (2016). Person-based split ergativity in Nez Perce is syntactic. *Journal of Linguistics*, 52(3), 533-564.
- Diercks, M. (2011). The morphosyntax of Lubukusu locative inversion and the parameterization of agree. *Lingua*, 121(5), 702-720.
- Diercks, M. (2012). Parameterizing case: Evidence from Bantu. *Syntax*, 15(3), 253-286.
- Diercks, M. (2013). Lubukusu complementizer agreement as a logophoric relation. *Natural Language and Linguistic Theory*, 31, 257-407.
- Diesing, M., & Jelinek, E. (1993). The syntax and semantics of object shift. *Working Papers in Scandinavian Syntax*, 51, 1-54.
- Donohue, M., & Brown, L. (1999). Ergativity: Some additions from Indonesia. *Australian Journal of Linguistics*, 19, 57-76.
- Dunn, M. (1995). *Sm'algyax: A reference dictionary and grammar for Coast Tsimshian*. Seattle: University of Washington Press.

- Fox, D., & Pesetsky, D. (2005). Cyclic linearization of syntactic structure. *Theoretical Linguistics*, 31, 1-45.
- Halle, M., & Marantz, A. P. (1993). Distributed morphology and the pieces of inflection. In K. Hale & S. J. Keyser (Eds.), *The view from building 20: Essays in linguistics in honor of Sylvain Bromberger* (p. 111-176). Cambridge, MA: MIT Press.
- Harley, H. (1995). *Subjects, events and licensing* (Unpublished doctoral dissertation). Massachusetts Institute of Technology.
- Harley, H., & Ritter, E. (2002). Person and number in pronouns: A feature-geometric analysis. *Language*, 78(3), 482-526.
- Harris, A. C. (1981). *Georgian syntax* (No. 33). Cambridge University Press.
- Hewitt, B. (1995). *Georgian: A structural reference grammar*. John Benjamins Publishing.
- Holmberg, A. (1986). *Word order and syntactic features* (Unpublished doctoral dissertation). University of Stockholm.
- Holmberg, A. (1999). Remarks on Holmberg's generalization. *Studia Linguistica*, 53, 1-39.
- Holmberg, A., & Hróarsdóttir, T. (2003). Agreement and movement in Icelandic raising constructions. *Lingua*, 113(10), 997-1019.
- Horn, L. (1975). On the nonsentential nature of the POSS-ING construction. *Linguistic Analysis*, 1(4), 333-387.
- Hornstein, N. (2018). The minimalist program after 25 years. *Annual Review of Linguistics*, 4, 49-65.
- Johnson, K. (2009). Gapping is not VP-ellipsis. *Linguistic Inquiry*, 40(2), 289-328.
- Kuroda, S.-Y. (1988). Whether we agree or not. In *Papers from the 2nd international workshop on Japanese syntax*.
- Kučerová, I. (2012). Toward a phase account of dependent case. In *Proceedings of the 35th annual Penn Linguistics Colloquium* (Vol. 18).
- Lambrecht, K. (1990). "what, me worry?" - 'mad magazine sentences' revisited. In *Proceedings of the sixteenth annual meeting of the Berkeley Linguistics Society* (p. 215-228).
- Legate, J. A. (2008). Morphological and abstract case. *Linguistic Inquiry*, 39(1), 55-101.
- Leslau, W. (1995). *Reference grammar of Amharic*. Harrossowitz.

- Levin, T. (2015). *Licensing without case* (Unpublished doctoral dissertation). Massachusetts Institute of Technology.
- Levin, T., & Preminger, O. (2015). Case in Sakha: are two modalities really necessary? *Natural Language and Linguistic Theory*, 33, 231-250.
- Mahajan, A. K. (1989). Agreement and agreement phrases. In I. Laka & A. K. Mahajan (Eds.), *Functional heads and clause structure* (Vol. 10, p. 217-252). MITWPL.
- Marantz, A. P. (1991). Case and licensing. In G. Westphal, B. Ao, & H. Chae (Eds.), *Proceedings of the 8th Eastern States Conference on Linguistics ESCOL 8* (p. 234-253).
- Marušič, F., Nevins, A., & Badecker, W. (2015). The grammars of conjunction agreement in Slovenian. *Syntax*, 18(1), 39-77.
- McFadden, T. (2004). *A study on the syntax-morphology interface* (Unpublished doctoral dissertation). University of Pennsylvania.
- McFadden, T. (2007). Default case and the status of compound categories. In *Proceedings of the 30th annual Penn Linguistics Colloquium*.
- McFadden, T., & Sundaresan, S. (2010). *Nominative case is independent of finiteness and agreement*. Talk Presented at BCGl 5: Case at the Interfaces.
- Milsark, G. L. (1988). Singl-ing. *Linguistic Inquiry*, 19(4), 611-634.
- Müller, G. (2004a). A distributed morphology approach to syncretism in Russian noun inflection. In O. Arnaudova, W. Browne, M. L. Rivero, & D. Stojanovic (Eds.), *Proceedings of FASL 12* (p. 353-373). Michigan Slavic Publications.
- Müller, G. (2004b). On decomposing inflection class features: Syncretism in Russian noun inflection. In G. Müller, L. Gunkel, & G. Zifonun (Eds.), *Explorations in nominal inflection* (p. 189-227). Mouton de Gruyter.
- Müller, G. (2005). Syncretism and iconicity in Icelandic noun declensions: a distributed morphology approach. *Yearbook of Morphology 2004*, 229-271.
- Munn, A. (1993). *Topics in the syntax and semantics of coordinate structures* (Unpublished doctoral dissertation). University of Maryland.
- Nash, L. (1996). The internal ergative subject hypothesis. In K. Kusumoto (Ed.), *Proceedings of NELS 26* (p. 195-209).
- Nevins, A. (2007). The representation of third person and its consequences for person-case effects. *Natural Language and Linguistic Theory*, 25(2), 273-313.

- Owens, J. (1985). *A grammar of Harar Oromo (Northeastern Ethiopia)*. Hamburg: Helmut Buske Verlag.
- Perlmutter, D. (1971). *Deep and surface structure constraints in syntax*. New York: Holt, Rinehart and Winston.
- Pesetsky, D. (2013). *Russian case morphology and the syntactic categories*. MIT Press.
- Pesetsky, D. (2017). Complementizer-trace effects. In M. Everaert & H. van Riemsdijk (Eds.), *The Wiley Blackwell companion to syntax* (2nd ed.). John Wiley and Sons, Inc.
- Pesetsky, D., & Torrego, E. (2001). T-to-c movement: Causes and consequences. In M. Kenstowicz (Ed.), *Ken Hale: a life in language* (p. 355-426). MIT Press.
- Pesetsky, D., & Torrego, E. (2007). The syntax of valuation and the interpretability of features. In *Phrasal and clausal architecture: Syntactic derivation and interpretation in honor of Joseph E. Emonds*. John Benjamins Publishing.
- Pires, A. (2007). The derivation of clausal gerunds. *Syntax*, 10(2), 165-203.
- Postal, P. (1979). *Some syntactic rule in Mohawk*. Garland Publishing.
- Preminger, O. (2014). *Agreement and its failures* (No. 68). MIT Press.
- Reuland, E. (1983). Govern -ing. *Linguistic Inquiry*, 14(1), 101-136.
- Reuland, E. (2011). *Anaphora and language design*. Cambridge, MA: MIT Press.
- Rezac, M. (2011). *Phi-features and the modular architecture of language* (No. 81). Springer.
- Rizzi, L. (1990). *Relativized minimality*. Cambridge, MA: MIT Press.
- Schütze, C. (1993). Towards a minimalist account of quirky case and licensing in Icelandic. *MIT Working Papers in Linguistics*, 19, 321-375.
- Schütze, C. (1997). *INFL in child and adult language: Agreement, case and licensing* (Unpublished doctoral dissertation). Massachusetts Institute of Technology.
- Schütze, C. (2001). On the nature of default case. *Syntax*, 4(3), 205-238.
- Shima, E. (2000). A preference for move over merge. *Linguistic Inquiry*, 31(2), 375-385.
- Sigurðsson, H. Á. (1989). *Verbal syntax and case in Icelandic* (Unpublished doctoral dissertation). University of Lund.

- Sigurðsson, H. Á. (1996). Icelandic finite verb agreement. *Working Papers in Scandinavian Syntax*, 57, 1-46.
- Sobin, N. (1997). Agreement, default rules, and grammatical viruses. *Linguistic Inquiry*, 28(2), 318-343.
- Stowell, T. (1981). Complementizers and the empty category principle. In V. Burke & J. Pustejovsky (Eds.), *Proceedings of NELS 11* (p. 345-363).
- Thráinsson, H. (1979). *On complementation in Icelandic*. Garland Publishing.
- Thráinsson, H. (2007). *The syntax of Icelandic*. Cambridge University Press.
- Valentine, R. (2001). *Nishnaabemwin reference grammar*. University of Toronto Press.
- Vergnaud, J.-R. (2008). Letter to Noam Chomsky and Howard Lasnik on “filters and control” April 17, 1977. In R. Freidin, C. P. Otero, & M. L. Zubizarreta (Eds.), *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*. MIT Press.
- Wali, K., & Koul, O. N. (1997). *Kashmiri: A cognitive-descriptive grammar*. Routledge.
- Willson, S. (1996). Verb agreement and case marking in Burushaski. In *Working papers of the Summer Institute of Linguistics North Dakota* (Vol. 40, p. 1-71).
- Woolford, E. (1997). Four-way case systems: Ergative, nominative, objective, and accusative. *Natural Language and Linguistic Theory*, 15, 181-227.
- Woolford, E. (2006). Lexical case, and inherent case, and argument structure. *Linguistic Inquiry*, 37, 111-130.
- Yip, M., Maling, J., & Jackendoff, R. (1987). Case in tiers. *Language*, 63(2), 217-250.
- Zaenen, A., Maling, J., & Thráinsson, H. (1985). Case and grammatical functions: the Icelandic passive. *Natural Language and Linguistic Theory*, 3, 441-483.
- Zeijlstra, H. (2004). *Sentential negation and negative concord* (Unpublished doctoral dissertation). University of Amsterdam.
- Zeijlstra, H. (2008). Modal concord. In M. Gibson & T. Friedman (Eds.), *Proceedings of SALT 17* (p. 317-332).