Poole, Ethan. 2024. Dependent-case assignment could be Agree. *Glossa: a journal of general linguistics* 8(1). pp. 1–23. DOI: https://doi.org/10.16995/glossa.9894

Open Library of Humanities

Dependent-case assignment could be Agree

Ethan Poole, University of California, Los Angeles, US, epoole@ucla.edu

Preminger (to appear) claims that an Agree-based theory of case assignment undergenerates because it cannot handle attested dependent-case patterns. This paper argues that dependent-case assignment can in fact be modelled using the operation Agree, building on independently motivated assumptions. Therefore, an Agree-based theory of case assignment does not undergenerate.

Glossa: a journal of general linguistics is a peer-reviewed open access journal published by the Open Library of Humanities. © 2024 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See http://creativecommons.org/licenses/by/4.0/.
OPEN ACCESS

1 Introduction

There are presently three prominent theories of case assignment. The first theory is case assignment via AGREE (also known as "functional-head case theory"), according to which all case reflects an AGREE-dependency between a DP and a particular head (typically, a functional head), as schematized in (1) (e.g. Chomsky 2000; 2001; Legate 2008).

(1) *Case assignment via AGREE*

$[\ X^0 \ldots [\ \ldots \mathrm{DP}_{[\textsc{case: x}]} \ldots Y^0 \ldots [\ \ldots \mathrm{DP}_{[\textsc{case: y}]} \ldots] \ldots] \ldots]$

The second theory is Marantz's (1991) configurational-case model (also known as "dependent-case theory"), which follows the algorithm in (2) (see also Bittner & Hale 1996; McFadden 2004; Preminger 2011; 2014; to appear).

(2) *Configurational-case algorithm*

a. Assign idiosyncratic lexical and inherent cases.

$[_{\mathrm{PP/VP/XP}}\ P^0/V^0/X^0\ \ \mathrm{DP}_{[\textsc{case: lex}]}\ ]$ where LEX = the relevant lexical case

b. Take the remaining DPs. If DP_α c-commands DP_β, assign dependent case either to DP_α ("high") or to DP_β ("low"). This directionality is parameterized.

$[\ \mathrm{DP}_{[\textsc{case: }\square]} \ldots [\ \ldots \mathrm{DP}_{[\textsc{case: }\square]} \ldots]\,]$ →

$[\ \underline{\mathbf{DP}}_{[\textsc{case: dep}]} \ldots [\ \ldots \mathrm{DP}_{[\textsc{case: }\square]} \ldots]\,]$

—or—

$[\ \mathrm{DP}_{[\textsc{case: }\square]} \ldots [\ \ldots \underline{\mathbf{DP}}_{[\textsc{case: dep}]} \ldots]\,]$

c. If a DP was not assigned case in the previous two steps, then assign it unmarked case.
[CASE:□] ↔ UNMARKED CASE

The third theory is the combination of the other two theories: a configurational-case model with the addition of case assignment via AGREE (Baker & Vinokurova 2010; Baker 2015). Following Preminger (to appear), I will adopt the labels in (3) for the three models.

(3) a. **m1 :=** case assignment via AGREE
b. **m2 :=** configurational case assignment
c. **m3 := m1 + m2**

In a comparison of the three case-assignment models, Preminger (to appear) advances three interconnected claims: The first claim is that the addition of case assignment via AGREE in **m3** is empirically vacuous because any **m3**-account can be systematically translated into an **m2**-account (i.e. **m2** ≡ **m3**). (The specific translation recipe is briefly outlined in section 2.) The second claim is that the expressive power of **m1** is a proper subset of the expressive power of **m2** (i.e. **m1** < **m2**). In other words, any **m1**-account can be systematically translated into an **m2**-account, but not vice versa. This asymmetry in translatability is due to the fact that **m1** has no analogue of dependent case, so not all **m2**-accounts can be translated into **m1**-accounts. The third claim is that given arguments in the literature that the notion of dependent case is necessary to account for various case patterns crosslinguistically (see e.g. Marantz 1991; Baker & Vinokurova 2010; Baker 2014; 2015; Levin & Preminger 2015; Poole 2015a; 2023; Baker & Bobaljik 2017; Jenks & Sande 2017; Yuan 2018; 2020; 2022; Agarwal 2022), the additional expressive power of **m2** is both necessary and sufficient; that is, **m1** undergenerates.

In this paper, I push back against the second claim and, by extension, the third claim. I will argue that a dependent-case mechanism can be implemented in **m1**, building on independently motivated assumptions about AGREE.[1] Consequently, **m2** does not in fact have more expressive power than **m1** does; rather, **m1** has at least the same expressive power as **m2** (i.e. **m1** ≥ **m2**). Crucially then, **m1** does not undergenerate. Any adjudication between **m1** and **m2** must therefore be carried out on the basis of theoretical considerations, not empirical reach.

The argumentation proceeds as follows: Section 2 briefly reviews Preminger's (to appear) claims concerning the relationships between **m1, m2**, and **m3**. Section 3 lays out the ingredients needed for the AGREE-based dependent-case mechanism, all of which have been independently proposed in the literature, and most of which are also assumed by Preminger. In section 4, I put together these pieces and show that they can model dependent-case assignment. For illustration, section 5 demonstrates how the proposed system handles "dative" and "accusative" in Sakha, both of which have been claimed to be dependent case (Baker & Vinokurova 2010; Baker 2015). Section 6 concludes by discussing the ramifications of the AGREE-based dependent-case mechanism.

2 Preminger's (to appear) claims

Preminger's (to appear) claims concerning the relationships between the three case-assignment models—**m1**, **m2**, and **m3**—crucially rest on the recipe in (4) for translating an **m3**-account into an equivalent **m2**-account.[2] In **m3**, case assignment via AGREE is used to capture instances

[1] Clem & Deal (2023) independently develop a similar system for modelling dependent case in terms of AGREE, couched in the interaction-satisfaction framework. Their argument though is stronger than the argument here: I argue for **m1** *or* **m2**, while they argue for **m1**.

[2] For Preminger, C being assigned configurationally may involve C being a lexical case assigned under closest c-command, perhaps by H^0 itself (so that C and H^0 travel together). This analysis of lexical case in Preminger (to appear) is more permissive than the standard assumption in **m2** that lexical case is assigned super-locally (i.e. by a head to its sibling). It is reasonable, I think, to question whether this analysis of lexical case just *is* case assignment via AGREE, and hence Preminger's version of **m2** just *is* **m3**. For the sake of argument though, I will set this question aside. Crucially, the validity of (4) as an **m3**-to-**m2** translation recipe does not affect the main argument of this paper, because the main argument applies to *both* **m2** and **m3**.

where a given case and agreement always occur together (e.g. “nominative” and φ-agreement in English); the recipe in (4) handles such instances. Note that this recipe makes use of *case discrimination*, which allows probes to only target goals bearing a certain case (Bobaljik 2008; Preminger 2011; 2014; Poole 2015b; Deal 2017). I will discuss this notion in a little more detail in the following section, as it is one of the ingredients for the AGREE-based dependent-case mechanism as well.

(4) *Recipe for translating an **m3**-account into an **m2**-account*
Let C be a case assigned under AGREE with H^0 in an **m3**-account. The corresponding **m2**-account of C is as follows:
a. H^0 enters the derivation with a φ-probe;
b. the φ-probe on H^0 is case-relativized to target DPs bearing case C; and
c. C is assigned configurationally before H^0 merges into the structure.

Under the **m2**-account produced by (4), agreement shows up on H^0 only when there is a DP bearing C in H’s search domain. If the φ-probe on H^0 finds no suitable goal, because there is no DP bearing C in its domain, then AGREE fails gracefully (following Preminger 2011; 2014). The result is precisely the same result as the **m3**-account of C: agreement on H^0 reflects the φ-features of the highest DP bearing C.

Against this backdrop, let us review Preminger’s three claims about the case-assignment models. The first claim is that given the recipe in (4), the addition of case assignment via AGREE in **m3** is empirically vacuous. Any instance of a case being assigned via AGREE in an **m3**-account can already be handled by the configurational-case half of **m3**. Therefore, the expressive power of **m2** and **m3** are equivalent (i.e. **m2** ≡ **m3**). Since the two models differ only in whether they include case assignment via AGREE, the addition of case assignment via AGREE is unnecessary.

The second claim is twofold. First, the recipe in (4) can be repurposed to translate any **m1**-account into an equivalent **m2**-account. For **m3**-accounts, (4) only applies to those cases that are not already assigned configurationally, whereas for **m1**-accounts, it applies to *all* the cases. Second, there exists no “inverse” recipe for translating any **m2**-account into an equivalent **m1**-account, because there is no corresponding notion of dependent case in **m1**. Taken together then, the expressive power of **m1** is a proper subset of the expressive power of **m2** (i.e. **m1** < **m2**).

Finally, the third claim is that given the many arguments in the literature for the notion of dependent case being necessary to account for various case patterns (see e.g. Marantz 1991; Baker & Vinokurova 2010; Baker 2014; 2015; Poole 2015a; 2023; Levin & Preminger 2015; Baker & Bobaljik 2017; Jenks & Sande 2017; Yuan 2018; 2020; 2022; Agarwal 2022), the extra expressive power afforded by **m2** is both necessary and sufficient. Because **m1** lacks an analogue of dependent case, it is unable to handle these case patterns and thus undergenerates.

In this paper, I am contesting the second claim, in particular that any **m2**-account cannot be translated into an **m1**-account. In what follows, I will show that it is possible to model dependent-case assignment using the operation AGREE and that consequently, **m1** and **m2** are, at the very least, equal in their expressive power.

3 Ingredients

The AGREE-based dependent-case mechanism employs several component pieces, all of which have been independently proposed in the literature. The first three of these components are explicitly assumed by Preminger (to appear: 6, 9) as well, namely all but ordered probes and minimal compliance.

I will use the term PROBE to refer to unvalued features that trigger AGREE, and I will notate them as [⋆F⋆] (following Heck & Müller 2007). In **m1**, case assignment is technically a byproduct of an AGREE-relation ("goal flagging" in the terminology of Deal to appear). I will notate the case byproduct as a subscript on probes: $[\star\text{F}\star]_{\text{CASE}}$. Conventionally, case-assigning probes are considered to be φ-probes (Chomsky 2000; 2001), which I will follow for the sake of simplicity (see section 6.4 for discussion).

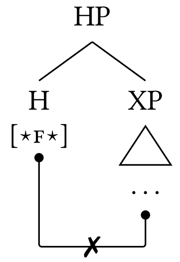
Case-discriminating probes: Probes may be CASE-DISCRIMINATING; that is, they may be restricted to targeting elements that bear certain case values (Bobaljik 2008; Preminger 2011; 2014; Poole 2015b; Deal 2017). I will notate case discrimination as a superscript on probes: $[\star\text{F}\star]^{\text{CASE}}$. (Thus, case-discriminating probes that themselves assign case are notated with both super- and subscripts.) Case discrimination is parameterized in terms of Marantz's (1991) implicational case hierarchy in (5). For example, a probe case-relativized to "dependent" $[\star\text{F}\star]^{\text{DEP}}$ may establish an AGREE-relation with a DP bearing dependent or unmarked case, but not lexical case.

(5) *Case-discrimination hierarchy*
unmarked ≫ dependent ≫ lexical

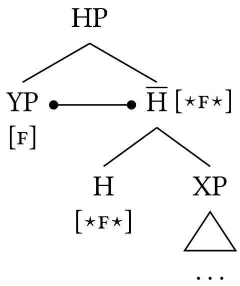
AGREE can fail: If a probe can be valued, then it must be valued, but unvalued probes do not crash the derivation (Preminger 2011; 2014).

Spec-head AGREE: Following Bare Phrase Structure, the nodes labelled XP, $\overline{\text{X}}$, and X^0 share the same features, because what projects is the head itself (Chomsky 1995), thereby allowing specifier–head AGREE under the guise of an AGREE-relation between siblings (Rezac 2003; Béjar & Rezac 2009; Clem 2022; Keine & Dash 2023). Specifically, if a probe π on a head H does not find a viable target in [Comp, HP] (6a), then π reprojects, and when HP merges with its specifier, π subsequently searchs [Spec, HP] (6b).

(6) a. *Step 1: Probe the complement*



b. *Step 2: Probe the specifier*



Ordered probes: Probes may be (extrinsically) ordered on a head $\langle \pi_1 \prec \pi_2 \prec \ldots \prec \pi_n \rangle$ such that π_m is active only if π_{m-1} has first been valued—that is, a stack (Müller 2010; 2011; Georgi 2014; 2017). In trees, I will depict ordered probes in a vertical column, where the highest probe is the first on the stack, the second highest probe is the second on the stack, and so forth.

Minimal compliance: When a head bears two or more probes, these probes interact with each other according to the Principle of Minimal Compliance (PMC), as formulated in (7) (Richards 1998; Rackowski & Richards 2005; van Urk & Richards 2015; Preminger 2019).[3] Under (7), a probe renders its goal invisible to subsequent probes on the same head.[4] (See also Georgi 2012

[3] The formulation of the PMC in (7) differs from the formulation in Rackowski & Richards (2005) in two ways. The first is an inconsequential terminological change: Rackowski & Richards use “probe” to refer to the head bearing the AGREE-triggering feature, whereas I use “probe” to refer to the feature (see van Urk & Richards 2015: 145 for discussion that the PMC applies to heads and not to features). The second change is a strengthening: for Rackowski & Richards, G *may* be ignored, whereas for (7), G *is* ignored. This strengthening is crucial for the present proposal, which requires a case-assigning head to target two distinct goals (see Georgi 2012: 200–201 for relevant discussion about “two-arguments-against-one-head” approaches). As far as I can tell, this strengthening is compatible with the existing PMC literature, and it is de facto how the PMC operates and is applied. However, should this strengthening of the PMC end up being consequential, then (7) should be considered a more substantive part of the proposal, rather than an off-the-shelf ingredient, which would weaken the main argument here about **m1**.

[4] There is a body of work arguing essentially the opposite: when a head bears two probes, those probes have to target a single goal satisfying both probes (e.g. van Urk & Richards 2015; Erlewine 2018; Bossi & Diercks 2019). I suspect that this comes down to the difference between two strictly ordered probes (which will feature in the proposal below; see (8)) and two probes fused together (i.e. composite probes). In the latter, the two probes search the structure simultaneously and thus cannot bleed one another. I leave exploring this issue to future research.

and Paparounas & Salzmann to appear, which make similar proposals about AGREE, but do not identify it with the PMC.)

(7) *Principle of Minimal Compliance (PMC)*
Once a probe on a head H has established an AGREE-relation with a goal G, subsequent probes on H ignore G for the rest of the derivation.
[based on Rackowski & Richards 2005: 582]

4 Proposal

Dependent-case assignment is handled with the probe stack in (8). The first probe, $[\star\varphi\star]^{\text{UNM}}$, is a simple φ-probe case-relativized to "unmarked", so that it may only establish an AGREE-relation with a DP whose [CASE: □] feature is unvalued; it does not assign case itself. The second probe, $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$, targets the closest DP and assigns it dependent case. Like the first probe, it is also case-relativized to "unmarked" and hence can only target DPs that have not yet been assigned case. For both probes, the φ-features copied over may or may not manifest as overt agreement.

(8) *Dependent-case probe stack*
$\langle [\star\varphi\star]^{\text{UNM}} \prec [\star\varphi\star]^{\text{UNM}}_{\text{DEP}} \rangle$

The order of the probes in the dependent-case probe stack (8) is crucial. The probe $[\star\varphi\star]^{\text{UNM}}$ serves to encode the dependency requirement—namely that another DP be present in order to license dependent case. The case-assigning probe $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ only becomes active *after* $[\star\varphi\star]^{\text{UNM}}$ has itself been valued. In other words, valuing $[\star\varphi\star]^{\text{UNM}}$ "unlocks" the ability to assign dependent case with $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$. Because AGREE may fail gracefully (Preminger 2011; 2014), not valuing $[\star\varphi\star]^{\text{UNM}}$ or $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ does not crash the derivation.

Given the PMC in (7), $[\star\varphi\star]^{\text{UNM}}$ and $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ on the same head cannot target the same DP, because the goal of $[\star\varphi\star]^{\text{UNM}}$ will be ignored by $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$.[5] Therefore, a DP cannot value $[\star\varphi\star]^{\text{UNM}}$ and then subsequently be assigned dependent case by now-active $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$.

The two types of dependent case—low and high—follow from the syntactic position of the dependent-case probe stack (8) relative to the relevant DPs. For ease of illustration, let us assume the simple functional sequence in (9) and begin by limiting our attention to simple transitive

[5] Under the PMC, ignoring a goal may involve probing *into* that goal; this is the logic behind "phase unlocking" in Rackowski & Richards (2005). Thus, with (8), the goal of $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ could in principle be inside the goal of $[\star\varphi\star]^{\text{UNM}}$. This possibility is certainly unusual with respect to case, and I do not know of any case patterns for which such an analysis would be obviously appropriate. Whether this is an overgeneration problem, however, depends on one's assumptions about DP-internal syntax, in particular the distribution of DP-internal phase boundaries and how case is assigned in DP, which together determine whether probing into a DP ever offers up any eligible DPs. Exploring these issues would take us too far afield, so I leave this issue for future research (see also fn. 9).

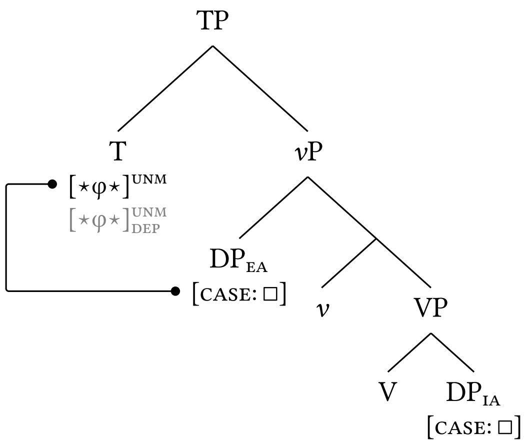
clauses, where the external argument (DP_{EA}) is base-generated in [Spec, *v*P] and the internal argument (DP_{IA}) is base-generated in [Comp, VP].

(9) *fseq* = ⟨C ≻ T ≻ *v* ≻ V⟩

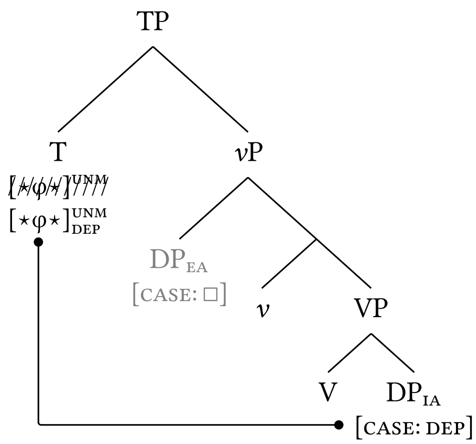
For low dependent case, the dependent-case probe stack is borne by some head above both DP_{EA} and DP_{IA}. Given the *fseq* in (9), this head is either C or T; for simplicity, let us assume it is T, as schematized in (10). Dependent case is assigned when T merges into the structure: First, DP_{EA} values $[\star\varphi\star]^{\text{UNM}}$, thereby unlocking $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ (10a). Second, $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ assigns dependent case to DP_{IA} (10b). Because DP_{EA} establishes an AGREE-relation with $[\star\varphi\star]^{\text{UNM}}$, it is invisible to $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$.

(10) *Low dependent case*

a. *Step 1: Unlock with caseless DP_{EA}*



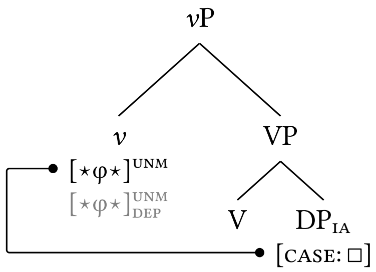
b. *Step 2: Assign dependent case to DP_{IA}*



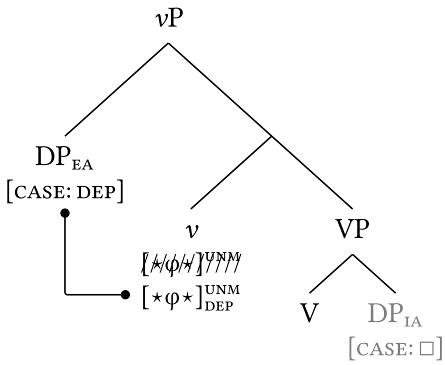
For high dependent case, the dependent-case probe stack is borne by whatever head introduces DP_{EA}, which here is *v*.[6] First, when *v* merges into the structure, DP_{IA} values $[\star\varphi\star]^{UNM}$, thereby unlocking $[\star\varphi\star]^{UNM}_{DEP}$ (11a). Second, when DP_{EA} merges into [Spec, *v*P], now-active $[\star\varphi\star]^{UNM}_{DEP}$ assigns dependent case to DP_{EA} (11b).

(11) *High dependent case*

a. *Step 1: Unlock with caseless* DP_{IA}



b. *Step 2: Dependent case assigned to* DP_{EA}



[6] This analysis makes two predictions with respect to high dependent case (which Clem & Deal's (2023) analysis makes as well). Thanks to Michael Yoshitaka Erlewine and anonymous reviewer, respectively, for bringing these to my attention. The first prediction is that high dependent case is more local than low dependent case. For low dependent case, the probe stack may be arbitrarily far away from the two DPs, while for high dependent case, the probe stack must be on the head introducing the higher of the two DPs. To my knowledge, this is not a problematic prediction, but it deserves more attention in future research.

The second prediction involves configurations with three DPs: a dependent-case probe stack on H that assigns high dependent case in simple two-DP configurations—one in [Comp, HP] and one in [Spec, HP]—should effectively "switch" to low dependent case when there is more than one DP in [Comp, HP], as schematized in (i).

(i) a. $[_{HP}$ $\underline{DP}_{DEP}$ [H [… DP …]]]

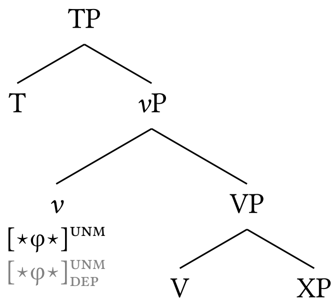
b. $[_{HP}$ DP [H [… DP … $\underline{DP}_{DEP}$ …]]]

In many instances, (i.b) will not arise because one of the DPs in [Comp, HP] will have already been assigned case (e.g. "dative"). Presumably though, (i.b) needs to be blocked more generally. I leave this task for future research, but note that this problem is similar in nature to how **m2** must block overapplication of high dependent case in ditransitive constructions (i.e. *ERG-ERG-ABS). Baker (2015: 230–240) handles this problem by invoking cyclic domains. Here, we might make a similar appeal to cyclic domains (though in a different way than Baker): the lowest DP in (i.b) is inaccessible to H because it is in a lower domain.

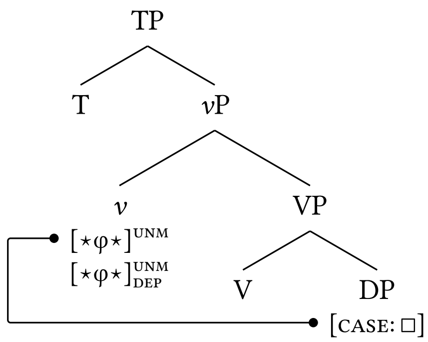
In both low-dependent (10) and high-dependent (11) configurations, there ends up being a DP whose [CASE:□] feature remains unvalued after the dependent-case probe stack has fully discharged—the one that is conventionally "nominative" or "absolutive". This [CASE:□] feature remains active for subsequent valuation. Thus, this DP may be assigned case by some higher head, such as T or C. We may also adopt the notion of unmarked case, whereby unvalued [CASE:□] features are assigned (or realized as) unmarked case at PF. Either way, the case may then be called "nominative" or "absolutive". Since the focus here is on dependent case, I will remain agnostic on this choice; see section 6.2 for further discussion.

If there is no DP in the search domain of (the head bearing) the dependent-case probe stack, then $[\star\varphi\star]^{\text{UNM}}$ is never valued, and $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ is never unlocked. In a similar fashion, if there is only one DP in the search domain, $[\star\varphi\star]^{\text{UNM}}$ gets valued, but now-active $[\star\varphi\star]^{\text{UNM}}_{\text{DEP}}$ is inert because there is no second DP to be assigned dependent case. For illustration, these two situations are schematized for high dependent case in (12) and (13) respectively, where XP is some non-DP element (e.g. a clause) and inclusion of T represents *v*P having been completed.

(12) *No DP in search domain*



(13) *Only one DP in search domain*



As with any **m1**-approach, there are many details in this proposed system that are a matter of analysis and must be established (in part) on a language-by-language basis. In particular, like in classical **m1**, languages may differ in terms of which heads assign which cases. The specific use of T and *v* above is for illustration. The addition of the proposed system to **m1** is only that a head may assign case using the probe stack in (8). If a language has multiple dependent cases, the generic case value DEP can be replaced with more specific values (e.g. ACC, ERG, DAT). These factors are illustrated using Sakha in the next section.

Another variable factor in the proposed system is case discrimination. Both probes in (8) being case-relativized to "unmarked" derives the canonical dependent-case pattern: only two unmarked DPs standing in a c-command relationship results in dependent case. Changing this setting though to lower positions on the hierarchy in (5) expands the range of licensors and eligible case-assignees, both of which are arguably needed empirically. Manipulating the case discrimination of the first probe allows DPs with valued case to unlock the case-assigning probe—that is, to license dependent case. Baker (2015: 194–197) argues that there are indeed languages where dependent case is licensed by DPs with case. Tweaking the first probe's case discrimination can account for such languages. Manipulating the case discrimination of the second probe allows case stacking—that is, DPs that have already been assigned case by a lower head to be assigned case again—, which has been documented in the literature (see Levin 2017 and references therein).

In sum, under this system, dependent-case assignment is achieved via AGREE and a particular ordered set of probes. The two types of dependent case—low and high—are the result of placing that ordered set of probes in different syntactic positions in relation to the relevant DPs. Thus, the core innovation of **m2**—dependent-case assignment—can be implemented using the tools of **m1**.[7]

5 Illustration: Sakha

For illustration, I demonstrate in this section how the proposed system handles "dative" and "accusative" case in Sakha, as described and analyzed in Baker & Vinokurova (2010) (henceforth B&V), which is commonly cited in support of **m2**. I will assume that B&V's analysis is descriptively correct, and so the task is to translate their **m2**-account into an **m1**-account using the AGREE-based dependent-case mechanism developed in section 4.

In Sakha, "dative" is a high dependent case within VP, and "accusative" is a low dependent case elsewhere. (To avoid going too far afield, I do not provide the relevant data

[7] Interestingly, the AGREE-based dependent-case mechanism proposed here is reminiscent of the original version of **m2** from Marantz (1991) and the **m2**-system of Bittner & Hale (1996). All three systems involve dependent-case assignment being mediated by a head. However, the similarities stop there.

Furthermore, nothing rules out other AGREE-configurations giving rise to dependent-case(-like) patterns (see also fn. 1). For instance, Branan (2019) suggests that dependent case might be modelled in terms of two DPs establishing an AGREE-relation with each other (though he does not work out the details). I will note here that such a system cannot straightforwardly handle languages with multiple dependent cases, like Sakha (see section 5), and thus cannot be responsible for all dependent-case patterns.

supporting these claims; see Baker & Vinokurova 2010). Under B&V's **m2**-account, "dative" and "accusative" are assigned by the dependent-case rules in (14a) and (14b) respectively (modified for consistency).

(14) a. *Sakha dative-case rule*
If there are two distinct argumental DPs in the same VP-phase such that DP_1 c-commands DP_2, then value the case feature of DP_1 as *dative* unless DP_2 has already been marked for case. [Baker & Vinokurova 2010: 595]

b. *Sakha accusative-case rule*
If there are two distinct argumental DPs in the same phase such that DP_1 c-commands DP_2, then value the case feature of DP_2 as *accusative* unless DP_1 has already been marked for case. [Baker & Vinokurova 2010: 595]

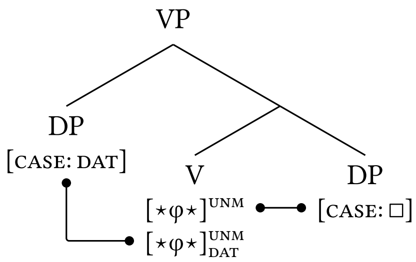
Under the AGREE-based dependent-case mechanism, the two dependent-case rules in (14) each correspond to a dependent-case probe stack. For "dative" case, V bears a dependent-case probe stack assigning DAT; because it is sandwiched between the relevant argument positions, it produces a high-dependent pattern. For "accusative" case, T bears a dependent-case probe stack assigning ACC; because it is positioned above the relevant argument positions, it produces a low-dependent pattern. For the sake of comparison, I will follow B&V in treating VP as a phase. Therefore, a DP must undergo object shift out of VP to be eligible for "accusative" (on this analysis, via the dependent-case stack on T). This object shift is contingent on the usual factors (e.g. definiteness, specificity).

Section 4 already walked through low-dependent and high-dependent configurations in isolation (see (10) and (11)). What is notable here is the combination and interplay between the two configurations. Consider the ditransitive in (15), where the direct object is "accusative", meaning it has undergone object shift out of VP.

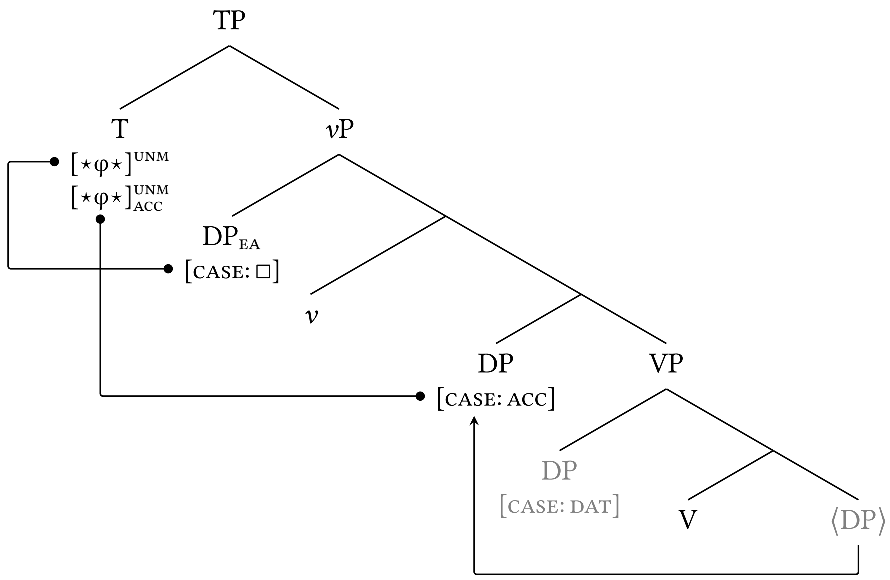
(15) Min kinige*(-ni) Masha-qa bier-di-m.
I book -ACC Masha-DAT give-PAST-1SG
'I gave the book to Masha.' [Baker & Vinokurova 2010: 602]

In VP, the direct-object DP in [Comp, VP] values $[\star\varphi\star]^{\text{UNM}}$ on V, and now-active $[\star\varphi\star]^{\text{UNM}}_{\text{DAT}}$ assigns DAT to the indirect-object DP in [Spec, VP], as schematized in (16).[8]

[8] For ease of presentation, I am assuming a very simple structure for ditransitives, where the indirect object is base-generated in [Spec, VP]—as B&V assume as well. The analysis of Sakha in this section is compatible with other approaches to ditransitives (e.g. with an Appl head); the dependent-case probe stack on V would just need to be moved up to whatever head introduces the indirect object.

(16)


After constructing the VP in (16), the [CASE: □] feature on the direct-object DP remains unvalued. As such, this feature is active for probes on higher heads. When T merges into the structure, DP_{EA} in [Spec, *v*P] values $[\star\varphi\star]^{\text{UNM}}$ on T, and then now-active $[\star\varphi\star]^{\text{UNM}}_{\text{ACC}}$ assigns ACC to the direct-object DP, which has object-shifted out of VP, as schematized in (17). The indirect-object DP, having remained in the VP-phase, is inaccessible. Even if it were to move out of VP, it would be invisible to both probes in a dependent-case probe stack because it already bears case.

(17)


If the direct-object DP does not object-shift out of VP, as in (18), then its [CASE: □] feature remains unvalued: DP_{EA} values $[\star\varphi\star]^{\text{UNM}}$ on T, but now-active $[\star\varphi\star]^{\text{UNM}}_{\text{ACC}}$ cannot access DPs in the VP-phase and so no case is assigned.

(18) Min Masha-qa kinige(#-ni) bier-di-m.
I Masha-DAT book -ACC give-PAST-1SG
'I gave Masha books/a book.' [Baker & Vinokurova 2010: 602]

When there are fewer than three argumental DPs, then the case-assigning probe on V or T is either never active, because $[\star\varphi\star]^{\text{UNM}}$ does not get valued, or it is inert, because there is no DP to assign case. All of the possible configurations with two argumental DPs and their outcomes are listed below in (19). The resulting case patterns are all what one finds in Sakha. For concreteness, the rightmost column in (19) gives an applicable example number from Baker & Vinokurova (2010) as support. (Note that there are multiple examples in their paper for each pattern).

(19)	*configuration*	*V's stack*	*T's stack*	*case pattern*	*B&V*
a.	[$_{vP}$ DP [$_{VP}$ DP DP]]	unlock→assign	unlock	UNM–DAT–UNM	(13a)
b.	[$_{vP}$ DP DP$_1$ [$_{VP}$ DP __$_1$]]	unlock→assign	unlock→assign	UNM–DAT–ACC	(13b)
c.	[$_{vP}$ DP [$_{VP}$ DP]]	unlock	unlock	UNM–UNM	(12b)
d.	[$_{vP}$ DP DP$_1$ [$_{VP}$ __$_1$]]	unlock	unlock→assign	UNM–ACC	(12a)
e.	[$_{vP}$ [$_{VP}$ DP DP]]	unlock→assign	n/a	DAT–UNM	(19)
f.	[$_{vP}$ DP$_1$ [$_{VP}$ DP __$_1$]]	unlock→assign	unlock	DAT–UNM	(21b)

(unlock = case-assigning probe is active; assign = case-assigning probe assigns case)

This analysis extends straightforwardly to embedded "accusative" subjects as well. In Sakha, an embedded subject may bear "accusative" case provided that the matrix clause contains another DP, as shown in (20).

(20) a. Keskil [**Aisen-y** [kel-bet dien]] xomoj-do.
Keskil Aisen-ACC come-NEG.AOR.3SG that become.sad-PAST.3SG
'Keskil became sad that Aisen is not coming.' [Vinokurova 2005: 366]

b. [**Aisen(*-y)** [massyyna atyylah-ar-a]] naada buol-la.
Aisen -ACC car buy-AOR-3SG need become-PAST.3SG
'It became necessary for Aisen to buy a car.' [Baker & Vinokurova 2010: 619]

B&V analyze this pattern in terms of raising: the embedded subject may move to embedded [Spec, CP], and the embedded CP may move out of VP. If both of these movement steps happen and there is another eligible DP in the matrix clause, the accusative-case rule in (14b) is triggered, thereby assigning "accusative" to the embedded subject. The same analysis can be applied here: if both of these movement steps happen, the embedded subject is accessible to the dependent-case probe stack on T. Thus, if there is another DP to value $[\star\varphi\star]^{\text{UNM}}$, then $[\star\varphi\star]^{\text{UNM}}_{\text{ACC}}$ becomes active and assigns ACC to the raised embedded subject.

It is worth making explicit that this analysis does not succumb to the particular argument that B&V raise against **m1** involving embedded "accusative" subjects. To derive the basic case alternation between actives and passives, **m1**-analyses standardly assume that transitive *v* assigns "accusative", but unaccusative *v* does not. Crucially, in Sakha, an embedded subject may get "accusative" even when the matrix predicate is unaccusative, as in (20a). An **m1**-analysis is thus faced with a conundrum: if unaccusative *v cannot* assign "accusative", the source of "accusative" in (20a) is unaccounted for, and if unaccusative *v can* assign "accusative", then it should do so in simple passives as well. Moreover, the head that triggers raising of the embedded subject cannot be an extra source of "accusative", because this would predict that all raised embedded subjects can be "accusative", contra (20b). The analysis of Sakha proposed here does not face this problem, precisely because it treats "accusative" as a dependent case. Thus, it does not assume that the head assigning "accusative" is absent in unaccusatives to account for the basic active–passive case alternation. In the same way that the rules in (14) are always active in B&V's analysis, the dependent-case probe stacks on T and V are present irrespective of verbal valency; they only assign case in the presence of another eligible DP. The same reasoning extends to other arguments against **m1** of this sort in the literature (e.g. Baker 2014; 2015; Poole 2015a).

6 Discussion

Recall from section 2 Preminger's (to appear) claims about the relationships between the three case-assignment models: (i) **m3** is empirically vacuous (i.e. **m2** ≡ **m3**), (ii) the expressive power of **m1** is a proper subset of the expressive power of **m2** (i.e. **m1** < **m2**), and (iii) the additional expressive power of **m2** is both necessary and sufficient. The core argument of this paper—a theoretical housekeeping of sorts—is that the second claim is incorrect. The AGREE-based mechanism developed in section 4 is a proof of concept demonstrating that dependent-case assignment can be implemented using the operation AGREE. Therefore, **m1** has at least the same expressive power as **m2** (i.e. **m1** ≥ **m2**).[9]

By extension, the third claim is incorrect as well, since, if the arguments advanced here are on the right track, **m2** does not have more expressive power than **m1**. In other words, the various case patterns identified in the literature as requiring the notion of dependent case (e.g. Marantz 1991; Baker & Vinokurova 2010; Baker 2014; 2015; Levin & Preminger 2015; Poole 2015a; 2023; Baker & Bobaljik 2017; Jenks & Sande 2017; Yuan 2018; 2020; 2022; Agarwal 2022) are not definitive arguments in favor of **m2**. As demonstrated in sections 4 and 5, these patterns can in fact be handled by **m1**. Thus, **m1** does not undergenerate.

Interestingly, the AGREE-based dependent-case mechanism provides further support for Preminger's first claim, but from the opposite side. Preminger (to appear) argues that **m3** is empirically vacuous because any **m3**-account can be translated into an **m2**-account (though see the caveat in fn. 2). A consequence of the AGREE-based dependent-case mechanism is that any **m3**-account can also be systematically translated into an **m1**-account. Putting these two points together: the combination of **m1** and **m2** (= **m3**) offers no advantage compared to **m1** and **m2** on their own.

[9] It is possible that **m1** has greater expressive power than **m2** (i.e. **m1** > **m2**) in light of the proposals made here—perhaps leading to overgeneration (see also fn. 6). Exploring this issue goes beyond the core argument of this paper (i.e. that **m2** does *not* have greater expressive power than **m1**), so I leave this issue for future research.

The goal of this paper is not to argue in favor of a particular case-assignment model, but rather to show that the dimension on which the models are standardly compared—namely, what data they can account for—is not (necessarily) the most insightful dimension of comparison. In terms of their empirical reach, **m1** and **m2** have comparable expressive power, and so they can in principle account for the same data. Therefore, comparison of the two case-assignment models must be carried out on the basis of theoretical parsimony and explanatory adequacy. Such arguments have of course been made in the literature—see e.g. Bárány & Sheehan (to appear) arguing for **m1** and much of the discussion in Baker (2015) arguing for **m2** (technically, **m3**); these are the kinds of arguments that should be given more weight and attention.

In what remains, I will conclude by discussing several issues that emerge from the proposals here. The first two are essentially empirical issues, and the last two concern theoretical parsimony.

6.1 Negative c-command conditions

The AGREE-based dependent-case mechanism cannot model dependent-case rules that rely on *negative* c-command conditions, where a case is assigned to a DP iff that DP is not c-commanded by any other DP in a given domain (Baker 2015: 89–110). This gap is only a problem insofar as such rules are warranted empirically, and I contend that they are not.

Baker (2015) uses such rules to handle so-called "marked nominative" and "marked absolutive". For simplicity, let us focus on "marked nominative". In marked-nominative languages, transitive and intransitive subjects are morphologically marked, and transitive objects are morphologically unmarked; the morphologically-marked case is referred to as "marked nominative". In fragment answers (and other elsewhere contexts), DPs are morphologically *un*marked—that is, not "marked nominative". Thus, "marked nominative" cannot be straightforwardly modelled as unmarked case in the **m2** sense. To account for this pattern, Baker proposes the dependent-case rule in (21) (modified for consistency). In simple clauses, it targets only the subject. In fragment answers, the rule does not apply because, by assumption, there is no TP.

(21) Assign DP_1 *marked nominative* if there is no other DP_2 in the same TP as DP_1 such that DP_2 c-commands DP_1. [Baker 2015: 93]

There is no barrier to **m1** generating marked-nominative and marked-absolutive patterns. For instance, "marked nominative" could just be assigned by T under closest c-command; T would always target the subject and would not occur in fragment answers (assuming that fragment answers lack TP, as Baker does). Baker's only argument against **m1**-analyses of "marked nominative" and "marked absolutive" is that such cases appear in contexts without overt agreement. I discuss this kind of objection to **m1** below, but in short, it is not a definitive argument.

As it stands then, while **m1** cannot model dependent-case rules with negative c-command conditions, it can handle the data that have motivated such rules. Therefore, the fact that the proposed system cannot model such rules is not inherently problematic, though this issue deserves more attention in future research.

6.2 Unmarked case

This paper has focused on dependent-case assignment, which I see as the core, defining innovation of **m2**. However, **m2** is also generally associated with the notion of unmarked case: the [CASE: □] feature on a DP may remain unvalued in the narrow syntax. As discussed in section 4, the AGREE-based dependent-case mechanism is compatible with, but does not require, the notion of unmarked case. After a dependent-case probe stack has fully discharged, there is a DP whose [CASE: □] feature remains unvalued. This [CASE: □] feature can be valued by a higher case-assigning head in the structure, possibly with another dependent-case probe stack, as in (17). If we adopt the notion of unmarked case, it could also remain unvalued.

The notion of unmarked case sometimes features in arguments for **m2** over **m1**, under the assumption that unmarked case is proprietary to **m2** (see e.g. Kornfilt & Preminger 2015; Levin 2017). However, unmarked case is not incompatible with **m1**. It is only incompatible with the (anachronistic) idea that case serves a role licensing nominal (i.e. "abstract Case") and the Case Filter, which enforces said licensing requirement. While these ideas are classically associated with **m1**, they are in no way integral to it. That is, nothing about case being assigned via AGREE requires that case be what licenses nominals (if they even need to be licensed) or that [CASE: □] features be valued in the narrow syntax.

6.3 Baroqueness

The AGREE-based dependent-case mechanism is perhaps baroque. It brings about dependent case essentially as an accident: a combination of probes specified in a very particular way and ordered extrinsically. In so doing, it arguably misses the underlying generalization that dependent case is about having two accessible DPs in a domain. On the AGREE-based system, this fundamental relation is indirect, mediated by a head. This baroqueness stands in stark contrast to the simplicity of dependent-case assignment in **m2** (see (2b)), which might be reason enough to favour **m2** over **m1**.

On the other hand, the individual components of the AGREE-based dependent-case mechanism are all *independently* motivated (see section 3). All else being equal, if these components are independently necessary and AGREE is able to assign case, then probe stacks like (8) are possible, irrespective of considerations of baroqueness.[10] That **m1** does not posit any proprietary operations for case assignment, while **m2** posits at least one (i.e. dependent-case assignment), might then be taken as reason to favor **m1** over **m2**.

6.4 Link between case and agreement

Under **m1**, case-assignment probes are conventionally assumed to be φ-probes (Chomsky 2000; 2001). This convention is arguably at odds with the rather well-documented fact that case and φ-agreement often diverge (e.g. Baker 2008; 2012; 2015; Bobaljik 2008; Preminger 2011; 2014; Bárány 2017). As such, a case appearing in the absence of corresponding φ-agreement is sometimes taken as an argument against **m1** and in favor of **m2** (see e.g. Baker 2012; Baker 2015: 34–47, 98–104; Preminger to appear). Under **m2**, case and φ-agreement are formally separate (though perhaps via fiat; see fn. 2), and so a dissociation between case and agreement is expected.

First, there is no (formal) reason that case assignment and φ-agreement cannot be the result of distinct, separate φ-probes. Whether the φ-features copied over to a φ-probe are given exponence is a matter for PF, not the narrow syntax. Although it is reasonable to be uncomfortable with pervasive null agreement, it has always been the price-of-admission for **m1** (and its Spec-Head predecessor), because the vast majority of case simply does not cooccur with φ-agreement.

Second, case-assigning probes (and probe stacks) could just as easily be modelled as probing for [D], the category feature borne by DPs, as illustrated in (22). It is not clear to me that there is an independent reason to make such a move, but it would alleviate the issue of null agreement.

(22) a. $\left\langle [\star\text{D}\star]^{\text{UNM}} \prec [\star\text{D}\star]^{\text{UNM}}_{\text{DEP}} \right\rangle$

b. $[\star\text{D}\star]^{\text{UNM}}_{\text{ERG}}$

[10] A reviewer brings to my attention an independent problem with Cyclic Agree (one of the components necessary for the AGREE-based dependent case mechanism), which, to my knowledge, has not been mentioned in the literature, and so I document it here. The problem has two components. First, under Cyclic Agree, a φ-probe on *v* produces an ergative agreement alignment: in transitives and unaccusatives, the probe targets the DP in VP, and in unergatives, the probe (via Cyclic Agree) targets the DP in [Spec, vP]. Second, case assignment after *v*P—more generally, after all the arguments have been introduced—can generate an accusative case alignment by singling out the highest DP. For example, in classical **m1**, T assigns "nominative" under closest c-command (and "accusative" is, say, assigned by transitive *v*). Alternatively, in **m2**, there is a low dependent case rule keyed to TP, assigning "accusative" to the lower of two DPs (and "nominative" is, say, unmarked case). Put together, it is possible then to generate a language with an accusative case alignment and an ergative agreement alignment, which is typologically unattested (Bobaljik 2008). I have nothing constructive to say about this problem, but I will note that it holds on any theory that (i) adopts Cyclic Agree, (ii) locates φ-agreement in the narrow syntax, and (iii) locates (some) case assignment in the narrow syntax as well. Given (ii) and (iii), φ-agreement is not necessarily ordered after case assignment, which Bobaljik (2008) argues is necessary to block this typologically unattested pattern. Virtually all contemporary theories of case assume that case assignment happens in the narrow syntax. Thus, the culprit seems like Cyclic Agree, which is also widely adopted.

These considerations of baroqueness and the link between case and agreement highlight a deeper tension between **m1** and **m2**, namely what one values more: simple operations, but more of them, or fewer operations, but with more bells and whistles. More specifically, it relates to the question of whether AGREE is only for φ-agreement or whether AGREE underlies other (or all) long-distance dependencies in syntax. What I have contributed in this paper is a proof-of-concept that AGREE can in principle handle dependent-case assignment—a long-distance dependency that has previously proven problematic for AGREE-based theories of case (i.e. **m1**). This is not in and of itself an argument in favor of **m1**, but it demonstrates that these broader questions about AGREE at least remain open with regards to case.

Abbreviations

1 = first person; 3 = third person; ACC = accusative; AOR = aorist tense; DAT = dative; DEP = dependent case; ERG = ergative; NEG = negation; PAST = past tense; SG = singular; UNM = unmarked case

Acknowledgements

For helpful discussion of these and related issues, many thanks to Hashmita Agarwal, Joseph Class, Emily Clem, Amy Rose Deal, Michael Yoshitaka Erlewine, Stefan Keine, Arthur Mateos, and three anonymous reviewers.

Competing interests

The author has no competing interests to declare.

References

Agarwal, Hashmita. 2022. *Phases are read-only*. Los Angeles, CA: UCLA MA thesis.

Baker, Mark. 2008. *The syntax of agreement and concord*. Cambridge, UK: Cambridge University Press. DOI: https://doi.org/10.1017/CBO9780511619830

Baker, Mark. 2012. On the relationship of object agreement and accusative case: Evidence from Amharic. *Linguistic Inquiry* 43(2). 255–274. DOI: https://doi.org/10.1162/LING_a_00085

Baker, Mark. 2014. On dependent ergative case (in Shipibo) and its derivation by phase. *Linguistic Inquiry* 45(3). 341–379. DOI: https://doi.org/10.1162/LING_a_00159

Baker, Mark. 2015. *Case*. Cambridge, UK: Cambridge University Press. DOI: https://doi.org/10.1017/CBO9781107295186

Baker, Mark & Bobaljik, Jonathan. 2017. On inherent and dependent theories of ergative case. In Coon, Jessica & Massam, Diane & Travis, Lisa (eds.), *The Oxford handbook of ergativity*, 111–134. New York: Oxford University Press. DOI: https://doi.org/10.1093/oxfordhb/9780198739371.013.5

Baker, Mark & Vinokurova, Nadya. 2010. Two modalities of case assignment: Case in Sakha. *Natural Language and Linguistic Theory* 28. 593–642. DOI: https://doi.org/10.1007/s11049-010-9105-1

Bárány, András. 2017. *Person, case, and agreement: The morphosyntax of inverse agreement and global case splits*. Oxford: Oxford University Press. DOI: https://doi.org/10.1093/oso/9780198804185.001.0001

Bárány, András & Sheehan, Michelle. to appear. Challenges for dependent case theory. In Sevdali, Christina & Mertyris, Dionysios & Anagnostopoulou, Elena (eds.), *The place of case in grammar*, Oxford: Oxford University Press.

Béjar, Susana & Rezac, Milan. 2009. Cyclic agree. *Linguistic Inquiry* 40(1). 35–73. DOI: https://doi.org/10.1162/ling.2009.40.1.35

Bittner, Maria & Hale, Kenneth. 1996. The structural determination of case and agreement. *Linguistic Inquiry* 27. 1–68.

Bobaljik, Jonathan. 2008. Where's phi? Agreement as a post-syntactic operation. In Harbour, Daniel & Adger, David & Béjar, Susana (eds.), *Phi-theory: Phi features across interfaces and modules,* 295–328. Oxford: Oxford University Press. DOI: https://doi.org/10.1093/oso/9780199213764.003.0010

Bossi, Madeline & Diercks, Michael. 2019. V1 in Kipsigis: Head movement and discourse-based scrambling. *Glossa* 4. 65. DOI: https://doi.org/10.5334/gjgl.246

Branan, Kenyon. 2019. Nominals Agree with each other in Coahuilteco. Ms., National University of Singapore.

Chomsky, Noam. 1995. Bare phrase structure. In Webelhuth, Gert (ed.), *Government and Binding Theory and the Minimalist Program,* 383–439. Cambridge, MA: Blackwell.

Chomsky, Noam. 2000. Minimalist inquiries: The framework. In Martin, Roger & Michaels, David & Uriagereka, Juan (eds.), *Step by step: Essays on minimalist syntax in honor of Howard Lasnik,* 89–155. Cambridge, MA: MIT Press.

Chomsky, Noam. 2001. Derivation by phase. In Kenstowicz, Michael (ed.), *Ken Hale: A life in language,* 1–52. Cambridge, MA: MIT Press. DOI: https://doi.org/10.7551/mitpress/4056.003.0004

Clem, Emily. 2022. Cyclic expansion in Agree: Maximal projections as probes. *Linguistic Inquiry* 54(1). 39–78. DOI: https://doi.org/10.1162/ling_a_00432

Clem, Emily & Deal, Amy Rose. 2023. Dependent case by Agree: Ergative in Shawi. Ms., UC San Diego and UC Berkeley.

Deal, Amy Rose. 2017. Syntactic ergativity as case discrimination. In Kaplan, Aaron & Kaplan, Abby & McCarvel, Miranda K. & Rubin, Edward J. (eds.), *Proceedings of the 34th West Coast Conference on Formal Linguistics (WCCFL 34),* 141–150. Somerville, MA: Cascadilla Proceedings Project.

Deal, Amy Rose. to appear. Current models of Agree. In Crippen, James & Dechaine, Rose-Marie & Keupdjio, Hermann (eds.), *Move and Agree: Towards a formal typology,* John Benjamins.

Erlewine, Michael Yoshitaka. 2018. Extraction and licensing in Toba Batak. *Language* 94(3). 662–697. DOI: https://doi.org/10.1353/lan.2018.0039

Georgi, Doreen. 2012. A relativized probing approach to person encoding in local scenarios. *Linguistic Variation* 12(2). 153–210. DOI: https://doi.org/10.1075/lv.12.2.02geo

Georgi, Doreen. 2014. *Opaque interactions of Merge and Agree.* Leipzig, Germany: Universität Leipzig dissertation.

Georgi, Doreen. 2017. Patterns of movement reflexes as the result of the order of Merge and Agree. *Linguistic Inquiry* 48(4). 585–626. DOI: https://doi.org/10.1162/LING_a_00255

Heck, Fabian & Müller, Gereon. 2007. Extremely local optimization. In Brainbridge, Erin & Agbayani, Brian (eds.), *Proceedings of the 26th Western Conference on Linguistics (WECOL 26),* 170–183. Fresno, CA: California State University.

Jenks, Peter & Sande, Hannah. 2017. Dependent accusative case and caselessness in Moro. In Lamont, Andrew & Tetzloff, Katie (eds.), *Proceedings of the 47th Meeting of the North East Linguistic Society (NELS 47),* vol. 2, 109–119. Amherst, MA: GLSA.

Keine, Stefan & Dash, Bhamati. 2023. Movement and cyclic Agree. *Natural Language and Linguistic Theory* 41. 679–732. DOI: https://doi.org/10.1007/s11049-022-09538-1

Kornfilt, Jaklin & Preminger, Omer. 2015. Nominative as no case at all: An argument from raising-to-ACC in Sakha. In Joseph, Andrew & Predolac, Esra (eds.), *Proceedings of the 9th Workshop on Altaic Formal Linguistics (WAFL 9)*, 109–120. Cambridge, MA: MITWPL.

Legate, Julie Anne. 2008. Morphological and abstract case. *Linguistic Inquiry* 39(1). 55–101. DOI: https://doi.org/10.1162/ling.2008.39.1.55

Levin, Theodore. 2017. Successive-cyclic case assignment: Korean nominative-nominative case-stacking. *Natural Language and Linguistic Theory* 35. 447–498. DOI: https://doi.org/10.1007/s11049-016-9342-z

Levin, Theodore & Preminger, Omer. 2015. Case in Sakha: Are two modalities really necessary? *Natural Language and Linguistic Theory* 33(1). 231–250. DOI: https://doi.org/10.1007/s11049-014-9250-z

Marantz, Alec. 1991. Case and licensing. In Westphal, German & Ao, Benjamin & Chae, Hee-Rahk (eds.), *Proceedings of the 8th Eastern States Conference on Linguistics (ESCOL 8)*, 234–253. Ithaca, NY: CLC Publications.

McFadden, Thomas. 2004. *The position of morphological case in the derivation: A study on the syntax-morphology interface.* Philadelphia, PA: University of Pennsylvania dissertation.

Müller, Gereon. 2010. On deriving CED effects from the PIC. *Linguistic Inquiry* 41. 35–82. DOI: https://doi.org/10.1162/ling.2010.41.1.35

Müller, Gereon. 2011. *Constraints on displacement: A phase-based approach.* Amsterdam: John Benjamins. DOI: https://doi.org/10.1075/lfab.7

Paparounas, Lefteris & Salzmann, Martin. to appear. First conjunct clitic doubling in Modern Greek: Evidence for Agree-based approaches to clitic doubling. *Natural Language and Linguistic Theory*.

Poole, Ethan. 2015a. A configurational account of Finnish case. In Wilson, David (ed.), *Proceedings of the Penn Linguistics Conference 38 (PLC 38),* Philadelphia, PA: University of Pennsylvania.

Poole, Ethan. 2015b. Deconstructing quirky subjects. In Bui, Thuy & Özyıldız, Deniz (eds.), *Proceedings of the 45th Meeting of the North East Linguistic Society (NELS 45),* vol. 2, 247–256. Amherst, MA: GLSA.

Poole, Ethan. 2023. Improper case. *Natural Language and Linguistic Theory* 41. 347–397. DOI: https://doi.org/10.1007/s11049-022-09541-6

Preminger, Omer. 2011. *Agreement as a fallible operation.* Cambridge, MA: MIT dissertation.

Preminger, Omer. 2014. *Agreement and its failures.* Cambridge, MA: MIT Press. DOI: https://doi.org/10.7551/mitpress/9780262027403.001.0001

Preminger, Omer. 2019. What the PCC tells us about "abstract" agreement, head movement, and locality. *Glossa* 4. 13. DOI: https://doi.org/10.5334/gjgl.315

Preminger, Omer. to appear. Taxonomies of case and ontologies of case. In Sevdali, Christina & Mertyris, Dionysios & Anagnostopoulou, Elena (eds.), *The place of case in grammar,* Oxford: Oxford University Press.

Rackowski, Andrea & Richards, Norvin. 2005. Phase edge and extraction. *Linguistic Inquiry* 36. 565–599. DOI: https://doi.org/10.1162/002438905774464368

Rezac, Milan. 2003. The fine structure of Cyclic Agree. *Syntax* 6(2). 156–182. DOI: https://doi.org/10.1111/1467-9612.00059

Richards, Norvin. 1998. The Principle of Minimal Compliance. *Linguistic Inquiry* 29(4). 599–629. DOI: https://doi.org/10.1162/002438998553897

van Urk, Coppe & Richards, Norvin. 2015. Two components of long-distance extraction: Successive cyclicity in Dinka. *Linguistic Inquiry* 46. 113–155. DOI: https://doi.org/10.1162/LING_a_00177

Vinokurova, Nadya. 2005. *Lexical categories and argument structure: A study with reference to Sakha.* Utrecht: LOT.

Yuan, Michelle. 2018. *Dimensions of ergativity in Inuit: Theory and microvariation*. Cambridge, MA: MIT dissertation.

Yuan, Michelle. 2020. Dependent case and clitic dissimilation in Yimas. *Natural Language and Linguistic Theory* 38. 937–985. DOI: https://doi.org/10.1007/s11049-019-09458-7

Yuan, Michelle. 2022. Ergativity and object shift across Inuit. *Language* 98(3). 510–551. DOI: https://doi.org/10.1353/lan.0.0270