Derivation by Merge and Its Input Restriction*

Hideharu Tanaka
Mie University

Rough draft, July 23, 2022

Abstract

This paper explores the applicability of the structure-building operation *Merge*, addressing how its input search is independently restricted. The issue is discussed in light of an *Anti-stacking Effect* (ASE), which is a generalization subsuming a *Case Adjacency Effect* and states that no adverb phrase occurs above a head that shares matching features with an element at its edge. To derive the ASE in a principled fashion, it is proposed that derivation by Merge is constrained by a *Matching Search Condition* (MSC), under which if the head of one input X to Merge holds feature matching inside X, then the search for the other input Y is limited and cannot select any non-matching objects in the workspace. It is shown that the MSC deduces all instances of the ASE found in the domains of VP and CP, and even covers a crosslinguistic variation in the distribution of adverb phrases, once the varying height of verb raising across languages is considered. A notable implication of the entire theory is that Merge should only be constrained in terms of input restrictions and interface conditions, suggesting that no output restrictions internal to the computational system are required.

Keywords

Merge, input search, workspaces, feature matching, case, adverbs

* This paper is a radically revised version of a chapter of my unpublished dissertation, submitted to Osaka University in December, 2015. I am very grateful to Shintaro Hayashi, Masaharu Kato, Teruyuki Mizuno, Kenta Mizutani, Takanobu Nakamura, Junya Nomura, Eri Tanaka, Sadayuki Okada, Yukio Oba, Hiroaki Saito, Yuta Sakamoto, and Yuta Tatsumi for constructive discussions. This work is supported by JSPS Grant-in-Aid for Early-Career Scientists, Grant Number 21K13024.

1. Introduction

Natural language syntax involves two structure-building processes, *base-generation* and *movement*. In the literature, these apparently different processes have been unified by a single operation *Merge*, a theoretical shift advanced by Bobaljik (1995) and Chomsky (2004). At the core of the Merge operation is set formation, whose inputs and outputs are called *syntactic objects* (SO). Thus, in its minimal definition, Merge takes two SOs as input and gives a set of them as output.[1]

(1) For any syntactic objects X, Y, Merge(X, Y) = {X, Y}.

Crucially, Merge has two modes of application, *External Merge* (EM) and *Internal Merge* (IM). These modes only differ in how the two input SOs are related; Merge(X, Y) counts as EM iff neither X nor Y contains the other, while it counts as IM iff either X or Y contains the other. These descriptions of EM and IM correspond to base-generation and movement, respectively. It therefore holds that these two processes are not distinct operations, but merely two modes of the same operation Merge.

In this paper, we explore the nature of this unifying operation Merge, addressing how its application is restricted. This question may be elaborated in different ways, as Merge may be regulated by different sorts of restriction. Here, we focus on its *input restriction* and formulate the issue as follows.

(2) **Input Search Problem**
Is Merge applicable to any syntactic objects?

This asks whether the input search for Merge itself is restricted, independently of general conditions such as Chomsky's (2000, 2001) *Phase Impenetrability Condition* (PIC), which would restrict the input search for any operation. If there is an input restriction on Merge, some SO may be inaccessible to Merge in extending another SO′, while if there is no such restriction, every SO must be accessible to Merge at any derivational stage, unless the PIC applies.

The predominant view for this issue has been *Partial Search*. This view claims that the input search for Merge is not totally free but partially limited, so there may be some SO that Merge cannot select. One implementation of this view is Chomsky's (1995) *Extension Condition*, which requires Merge to target a *root*, a SO not contained in any others. In other words, the input search is conditioned as follows.

(3) **Chomsky's (1995) Extension Condition**
Any syntactic objects X, Y, may be input to Merge, *only if X or Y is a root*.

Under this condition, once Merge takes a non-root SO as one input, the other input is confined to a root SO. Importantly, Chomsky (2004) suggests that the Extension Condition accords with a *Strong Minimalist Thesis* (SMT), which says that the design of syntax meets principles of efficient computation, satisfying interface conditions. This perspective is made clear in the following quote.

> Elementary considerations of efficient computation require that Merge of α to β involves minimal

[1] This definition of Merge only indicates its bare minimum, and leaves it open whether the output of Merge is *syntactically labeled*, as proposed by Chomsky (1995), or not, as argued by Collins (2002) and Seely (2006). Whichever approach we may adopt, the following discussion is worth considering.

> search of β to determine where α is introduced, as well as least tampering with β: search therefore satisfies some locality conditions (let us say, defined by least embedding, "closest" under c-command), and Merge satisfies an extension condition, with zero search. One possibility is that β is completely unchanged (the strong extension condition) … Assume some version of the extension condition to hold, in accord with the SMT. (Chomsky 2004: 109)

This way, the view of Partial Search leads to a sort of minimal search, serving as an optimal design specification for the computational system. Of course, this does not mean that we must adopt the specific proposal of the Extension Condition; it is an empirical issue what input restriction we should come up with. The point is that it is a conceptually reasonable option to restrict the input search for Merge.

More recently, however, the opposite view, *Total Search*, has been offered by Chomsky (2019) and Chomsky et al. (2019). Under this view, the input search for Merge itself is not restricted at all, and it can select any SO in extending another SO′. The most explicit instantiation of this view is Chomsky's (2019) *Recursion Principle*, which permits Merge to target any SOs as stated below.

(4) **Chomsky's (2019) Recursion Principle**
Any two syntactic objects X, Y, may be input to Merge. (*no input condition*)

Clearly, this principle is incompatible with a minimal search restriction on Merge such as the Extension Condition, so one may wonder how this move is motivated by the SMT. In fact, there is reason to believe that the Recursion Principle is the null hypothesis, complying with the SMT. For instance, consider Chomsky's (2019) following proposal (see also Chomsky et al. 2019: 245).

> [W]hat's the basic idea of recursion? It's that every object that's generated must be available for later computations … we want to try [this principle] given [the SMT], we want to formulate *recursion* in a way that stipulates no specific properties, so we don't put any extra conditions on it: recursion ought to be *free* … One consequence shows that what's been called the Extension Condition is a mistake, because [it] simply stipulated that the only accessible syntactic objects are the whole syntactic objects … [T]he Extension Condition … has to be withdrawn. (Chomsky 2019: 276)

Under this reasoning, the view of Total Search or the Recursion Principle counts as an optimal design specification. In other words, it is also a conceptually reasonable option to impose no restriction on the input search for Merge, although it may be restricted by general conditions such as the PIC.

Our concern is these opposite views; that is, which is tenable, Partial Search or Total Search? As we have seen, the SMT allows us to take either direction, as both are conceptually reasonable. In this paper, therefore, we will develop an empirical discussion on the issue. Specifically, we will argue for Partial Search by establishing a problematic generalization for Total Search, which we call an *Anti-stacking Effect* (ASE). Put simply, the ASE says that no adverbial phrase (AdvP) occurs at the edge of a head H iff H shares matching features F with a phrase XP at its edge, as schematized below.[2]

---

[2] To simplify the discussion, this paper does not consider the operation *Pair Merge*, which is set to introduce "adjuncts" (e.g., Chomsky 2000, 2004, 2013), since the question is the same even for Pair-Merge; why can't it combine AdvP and H″? See also Oseki (2015) for conceptual and empirical problems with Pair Merge.

(5) **Anti-stacking Effect (ASE)**
$[_{\text{HP}}$ *AdvP $[_{\text{H}''}$ **XP**$_{[\text{F}]}$ $[_{\text{H}'}$ H$_{[\text{F}]}$ […]]]]

A potential limitation with Total Search is that it forces us to develop a representational account of the ASE, since it permits Merge to select any SOs. In other words, we need to invoke *filters on the outputs of derivations*, which may rule out the *output* of Merge(AdvP, H″). Partial Search, in contrast, allows us to develop a derivational account of the ASE. This means that we may invoke *filters on the inputs to derivations* in order to rule out the *inputs* in Merge(AdvP, H″). Of course, there is no *a priori* preference between representational and derivational accounts, but we argue that the most successful way to derive the ASE is to exclude an AdvP as an input to Merge when it is applied to the structure H″.

This paper is organized as follows. Section 2 aims to establish the ASE, exploring the syntax of VP in terms of Stowell's (1981) *Case Adjacency Effect*. We show that the ASE is a generalization subsuming that effect, and argue that the best representational account of the ASE, which is due to Koizumi's (1995) *Agreement-based Case Theory*, has some problematic aspects. Then, to give a derivational account of the ASE, Section 3 offers a theory of *derivation* and *workspace* and proposes that the input search for Merge is restricted by a *Matching Search Condition* (MSC), under which if the head of one input holds feature matching in the sense of Chomsky (2001), the search for the other input cannot select any non-matching SOs in the workspace. With this proposal, Section 4 considers the ASE found in the domain of CP and shows that the MSC even derives a crosslinguistic difference in the distribution of AdvPs, once the variation in the height of verb raising is considered. Section 5 concludes with some theoretical implications.

2. Description

2.1. Preliminary: The Case Adjacency Effect

In order to establish the ASE, we begin by reconsidering a *Case Adjacency Effect* (CAE). The CAE is a traditional generalization that Stowell (1981) establishes based on Chomsky's (1981) insight, and it concerns the linear adjacency between a main verb (V) and its object DP. Among Stowell's various observations, we focus on the distribution of AdvPs and formulate the CAE as follows.

(6) **Case Adjacency Effect (CAE)** (*to be revised*)
No AdvP occurs between V and its object DP.

For instance, no AdvP can occur between the verb *read* and its righthand XP if the XP is DP, but an AdvP may occur between them if the XP is PP. This difference is shown in (7b) and (8b).[3]

(7) a. Mary *slowly* **read the book** yesterday.
b. * Mary **read** *slowly* **the book** yesterday.
c. Mary **read the book** *slowly* yesterday.

(8) a. Mary *slowly* **read to her son** yesterday.
b. Mary **read** *slowly* **to her son** yesterday.
c. Mary **read to her son** *slowly* yesterday.

[3] Each original example given without a citation was judged by 10 to 20 native speakers of U.S. English, who were recruited via Amazon Mechanical Turk, and examples of low acceptability are presented with asterisk *.

The question then is whether the CAE also concerns the structural relation between V and its object DP. This is an important point, because Stowell (1981) proposed that the CAE only concerns their linear relation, and should be reduced to an *Adjacency Condition on Case Assignment*; a Case-assigner V and its Case-assignee DP do not allow any XP to intervene. However, many researchers have accumulated evidence that the proper account of the CAE must consider more than the linear relation (e.g., Pesetsky 1989, Pollock 1989, Johnson 1991, Bowers 1993, Koizumi 1995, Runner 1998, Janke and Neeleman 2012, Belk and Neeleman 2017). In the following, built on these previous achievements, we will argue that the CAE must be characterized by *two structural preconditions*.

The first precondition, as alluded to by Pesetsky (1989), is the lack of V-to-I raising, where the category I is a set of inflections such as tense and mood. This means that, for the CAE to hold, V must not raise to I. That is why the verb in (7) and (8) holds the CAE, since English regular Vs do not undergo V-to-I raising (e.g., Emonds 1978, Pollock 1989). Still, we assume with Kratzer (1996) that a functional head *Voice* occurs above VP, introducing an Agent DP to its edge, and that V undergoes raising to Voice. The first precondition therefore amounts to saying that V must stay overtly within VoiceP.

(9) **Precondition 1: Overt placement of V in the position of Voice**
  a. CAE holds only if V is in VoiceP: $[_{\text{IP}}$ I $[_{\text{VoiceP}}$ … **V** … **DP** …]]
  b. It does not hold if V occurs in I: $[_{\text{IP}}$ **I-V** $[_{\text{VoiceP}}$ … t_{V} … **DP** …]]

The validity of this precondition can be upheld if we look at the distribution of the copula verb *be*. Becker (2002) shows that the English copula occurs in the same position as regular Vs when it is in the bare form (i.e., *be*), whereas it occupies the position of I when it is in an inflected form (e.g., *is*). Thus, the inflected *is* must precede the negation particle *not*, but the bare *be* must follow it, as we see below.

(10) a. * John **not** *is* tired.
  b. John *is* **not** tired. (Becker 2002: 3)

(11) a. John will **not** *be* late.
  b. * John will *be* **not** late. (Becker 2002: 3)

Given this difference, it is expected that *be* holds the CAE when it comes with an argument DP, as in the expletive *there* construction. This is a correct expectation, and Larson (1989) and Lasnik (1992) illustrate that *be* blocks an AdvP from intervening in the same way as regular Vs do, while *is* does not.[4]

(12) a. There will *usually* **be a question of legality**.
  b. * There will **be** *usually* **a question of legality**. (Larson 1989: 20)

(13) a. I believe there **is** *usually* **a solution**.
  b. * I believe there to **be** *usually* **a solution**. (Lasnik 1992: 387–388)

The first precondition is also motivated crosslinguistically, as noted by Belk and Neeleman (2017).

---

[4] We assume that the associate DP in the *there* construction obtains Case within VoiceP in the same way as accusative DPs do; that is, the associate DP gets its Case-feature valued as accusative, not as nominative, contra Chomsky (2001). See Lasnik (1992, 1995) for a proposal along these lines.

For instance, Emonds (1978) and Pollock (1989) argue that regular Vs in French, unlike those in English, undergo V-to-I raising, by which they precede the negation particle *pas*, as in (14b). Thus, given the first precondition, it is no wonder why French regular Vs do not hold the CAE, as in (15b).

(14) a. * John *likes* **not** Mary.
b. Jean n' *aime* **pas** Marie. (Pollock 1989: 367)

(15) a. * John **kisses** *often* **Mary**.
b. Jean **embrasse** *souvent* **Marie**. (Pollock 1989: 367)

Moreover, Vikner (2001) reveals that other languages than French may also put an AdvP between V and its object DP, including Icelandic and Yiddish as shown in (16). Note that these languages have been analyzed by Vikner (1995) as having V-to-I raising. We therefore conclude that V and its object DP need not be adjacent when V raises to I, a fact that Stowell's (1981) Adjacency Condition cannot capture.

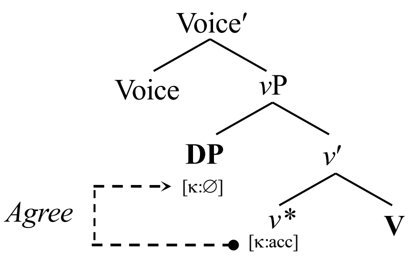
(16)

English:	* That	John	**eats**	*often*	**tomatoes**	surprises most people.	
Danish:	* At	Johan	**spiser**	*ofte*	**tomater**	overrasker de fleste.	
Faroese:	* At	Jón	**etur**	*ofta*	**tomatir**	kemur óvart á tey flestu.	
Icelandic:	Að	Jón	**borðar**	*oft*	**tómata**	kemur flestum á óvart.	
Yiddish:	Az	Jonas	**est**	*oft*	**pomidorn**	iz a khidesh far alemen.	
French:	Que	Jean	**mange**	*souvent*	**des tomates**	surprend tout le monde.	(Vikner 2001: 4)

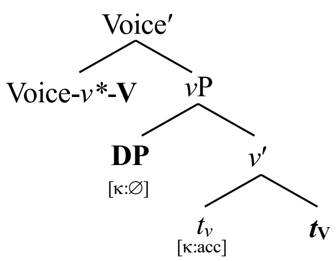
Still, we believe that Stowell's approach to the CAE is basically right in attributing the difference between DP and PP to their Case-theoretic status. Recall that we must distinguish DP and PP somehow, since only DP must be adjacent to V. This contrast has been shown in (7) and (8), where the same verb *read* is used, so it cannot be reduced to the height of V. Given this, we assume that the DP *the book* and PP *to her son* in (7) and (8) occupy distinct positions. In Chomsky's (2001) terms, DP and PP differ in that DP has an unvalued Case or κ-feature, but PP does not. That is, DP must occur in a κ-position where its unvalued κ-feature [κ:∅] is valued by *Agree*, a feature valuation operation on *lexical items* (LIs).

What is then needed is a precise definition of κ-positions. We will define a κ-position as the closest position that *c-commands* a valued κ-feature (cf. Chomsky 2001). To enable this kind of VP syntax, we adopt Marantz's (1997) theory of *Distributed Morphology*, under which a word is syntactically split into at least two LIs, a *root* and a *categorizer*, and assume that a verb consists of its root V, *verbalizer v*, and Voice (e.g., Pylkkänen 2002, Harley 2013, Legate 2014). Suppose that the head *v* falls into two sorts: *athematic v* and *thematic v**. We suggest that *v** is like Voice in its semantic type and introduces a Theme DP to its edge, as shown below (cf. Borer 2005, Basilico 2008, Bowers 2010, Lohndal 2014).

(17) a. Base structure of [*read the book*]



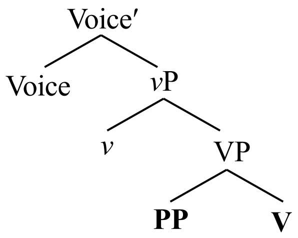
b. V-raising through *v* to Voice



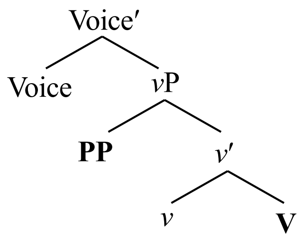
Importantly, we assume that a verbalizer, either *v* or *v**, may have an accusative κ-feature [κ: acc]. This idea allows us to define the edge position of *v*P in (17) as a κ-position, once we adopt Bošković's (2007) view of κ-valuation as an independent case of Agree, not a reflex of φ-agreement. Specifically, Bošković proposes that every DP with an unvalued κ-feature acts as a *probe*, searching its c-command domain for a *goal* with a valued κ-feature. In (17), for example, the DP with [κ:∅] c-commands the head *v* with [κ: acc], so they can undergo Agree in κ-features, resulting in the valuation of [κ:∅] as accusative.

On the other hand, we assume that PP generally acts as a kind of modifier. This view makes sense under *event semantics* (e.g., Davidson 1967, Parsons 1990), where PP and AdvP are treated as predicates of events and conjoined to the main predicate as modifiers. Accordingly, any PP may occur at the edge of VP or *v*P as indicated below, so long as its semantic composition with its sister succeeds.[5]

(18) a. Base structure 1 of [*read to her son*]



b. Base structure 2 of [*read to her son*]



Still, since PP does not have [κ:∅], the head *v* need not have [κ: acc]. In other words, the head *v* must have [κ: acc] if the *v*P contains a DP as in (17), but it need not, if the *v*P contains no DP as in (18). Thus, while we do not deny the possibility that a PP may occupy the edge position of *v*P in (18), we define only the edge position of *v*P in (17) as a κ-position, in the sense that it c-commands a valued κ-feature.

At this point, we argue that placing the object DP at the edge of *v*P is empirically motivated. For instance, Runner (1998) observes that the object DP and an AdvP or PP modifier can form a conjunct excluding the V as in (19a), and that modifiers alone can do so excluding the object DP as in (19b).

(19) a. Chris ate **(** **the meat** slowly **)** but **(** **the vegetables** quickly **)**. (Runner 1998: 51, 58)
b. Ginger saw Mary Ann **(** **in the park** after dinner **)** and **(** **at the dock** around sunset **)**.

For these examples, suppose that the modifiers (MO) occur at the edge of VP. It is then possible to identify the conjuncts in (19a) as *v*Ps, and those in (19b) as VPs, assuming that the root V has undergone across-the-board raising to Voice. The rough structures of these conjuncts are given below.

(20) a. [Voice-*v*-V [**(**$_{vP}$ **DP** t_v [$_{VP}$ MO t_V]**)** & **(**$_{vP}$ **DP** t_v [$_{VP}$ MO t_V]**)**]]
b. [DP t_v [**(**$_{VP}$ **MO** [$_{V'}$ MO t_V]**)** & **(**$_{VP}$ **MO** [$_{V'}$ MO t_V]**)**]]

Remarkably, this analysis entails that the object DP can c-command everything to its right. Thus, given that *negative polarity items* (NPI) must be c-commanded by negative elements, we predict that a negative object DP can license an NPI in a following modifier. This prediction is correct, as shown below.

[5] Merging a PP modifier with a root V is a case of root modification. Given that root modification is cross-linguistically attested by Pylkkänen (2002) and Levinson (2010), we assume that appropriate types of PPs and AdvPs can modify verbal roots to the extent that their combinations are semantically allowed.

(21) a. I visited **none of the patients** *at any of the hospitals*.
b. * I visited *any of the patients* **at none of the hospitals**. (Runner 1998: 41)

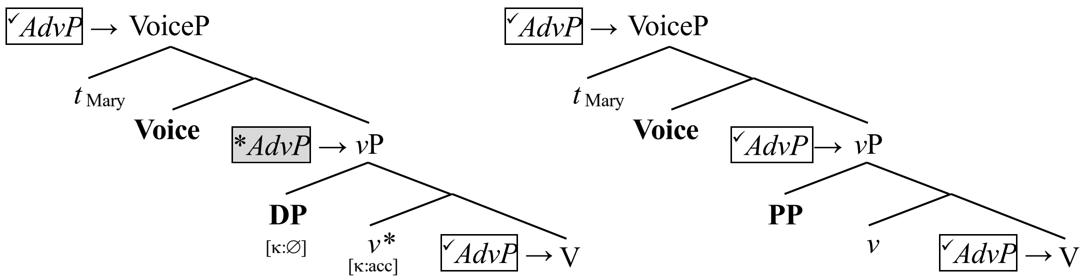
It is therefore assumable to base-generate the object DP at the edge of *v*P, enabling it to c-command everything to its right. Note that this base position is also a κ-position under our analysis, so the object DP is allowed to undergo κ-valuation in situ without moving at all.

This theory will distinguish the Case-theoretic status of DP and PP in a different way than previous studies. For example, Johnson (1991), Koizumi (1995), and Runner (1998) claim that DP and PP must occupy *structurally distinct positions*, assuming that the object DP moves to a position where no PP can occur, like the edge of *Agreement Phrase* in Chomsky (1993). Such a premise is dispensable under our theory. This point is made clear by examining the distribution of the AdvP *slowly*, repeated here.

(22) a. Mary (✓*slowly*) **read** (**slowly*) **the book** (✓*slowly*) yesterday. (= (7))
b. Mary (✓*slowly*) **read** (✓*slowly*) **to her son** (✓*slowly*) yesterday. (= (8))

First of all, we assume that the subject DP *Mary* occurs at the edge of VoiceP and moves to the edge of IP for κ-valuation, and that the PP *to her son* may occupy the edge of *v*P as the DP *the book* does. Then, our structure of VoiceP leads us to analyze the possible sites for *slowly* as including the left edge of VoiceP and VP. This analysis is visualized as follows, where we omit V-raising to Voice.[6]

(23) a. Distribution of *slowly* in (22a) b. Distribution of *slowly* in (22b)



On the other hand, the impossible site for *slowly* is identified with the edge of *v*P in (23a). Our point is that this site has no same status as the edge of *v*P in (23b). Specifically, the edge positions of *v*P in (23a) and (23b) are *featurally distinct positions*, and only the former counts as a κ-position in the sense that it c-commands a valued κ-feature. This is how we distinguish the status of DP and PP.

With this feature-based view of κ-positions, we now formulate the second precondition for the CAE. That is, no AdvP can intervene between V and its object DP only if the DP overtly occurs at the edge of *v*P whose head has a valued κ-feature. This formulation is schematized below.

(24) **Precondition 2: Overt placement of DP in the κ-position of *v*P**
a. CAE holds only if DP is in a κ-position: [$_{\text{VoiceP}}$ Voice [$_{v\text{P}}$ **DP**$_{[\kappa:\varnothing]}$ [$_{v'}$ $v^*_{[\kappa:\text{acc}]}$ **V**]]]]
b. It does not hold if DP occurs elsewhere: [$_{\text{VoiceP}}$ Voice [$_{v\text{P}}$ $\boldsymbol{t}_{\mathbf{DP}}$ [$_{v'}$ $v^*_{[\kappa:\text{acc}]}$ **V**]]]]

---

6 We do not deny the possibility that modifiers like AdvPs and PPs can also merge to the right edge of VoiceP, *v*P, or VP, as the AdvP *yesterday* seems to do it. For relevant discussion on this possibility, see Bowers (1993), Ernst (2002), Junke and Neeleman (2012), and Belk and Neeleman (2017).

The first support for this precondition comes from the *heavy DP shift* construction, which is widely assumed to involve rightward DP movement (e.g., Ross 1967). For instance, Postal (1974) shows that an AdvP may occur between V and its object DP, if the DP is phonologically heavy, as we see below.

(25) a. I **love** *very much* **the woman who came to see me**.
b. * I **love** *very much* **Joan**. (Postal 1974: 136)

This contrast readily follows from the second precondition, once we assume that heavy DP shift involves an operation that places the DP in a different position than the edge of *v*P. On this assumption, the contrast above shows that V and its object DP need not be adjacent if the position of the DP structurally shifts, another fact that Stwoell's (1981) Adjacency Condition fails to predict.

There is a further argument that the CAE is caused by the overt placement of DP in the κ-position of *v*P. The evidence we provide here is based on the *exceptional case-marking* (ECM) construction, whose embedded subject we refer to as the ECM subject. First of all, it should be noted that the ECM subject must appear in accusative, and not in nominative, as the following contrast illustrates.

(26) a. Mary proved **him** to be innocent.
b. * Mary proved **he** to be innocent.

Given this, we analyze the ECM subject as moving overtly into the matrix clause (e.g., Postal 1974, Lasnik and Saito 1991, Johnson 1991, Bowers 1993, Koizumi 1995, Runner 1998, Richards 1997, Lasnik 2003). Specifically, we assume that it reaches the edge of the matrix *v*P to get its κ-feature valued. Hence, while the direct object of a transitive verb does not move, the ECM subject does, as shown below.[7]

(27) a. Base: $[_{\text{VoiceP}}$ Voice $[_{vP}$ $v_{[\kappa:\text{acc}]}$ $[_{\text{VP}}$ $\mathbf{V}$ $[_{to\text{P}} \ldots \mathbf{DP}_{[\kappa:\varnothing]} \ldots]]]]$
b. Movement: $[_{\text{VoiceP}}$ Voice-v-$\mathbf{V}$ $[_{vP}$ $\mathbf{DP}_{[\kappa:\varnothing]}$ $[_{v'}$ $t_{v[\kappa:\text{acc}]}$ $[_{\text{VP}}$ $\boldsymbol{t}_{\mathbf{V}}$ $[_{to\text{P}} \ldots \boldsymbol{t}_{\mathbf{DP}} \ldots]]]]]$

Note that this DP raising is ensured by Bošković's (2007) κ-valuation system, since the ECM subject in situ does not c-command the matrix *v* and cannot undergo Agree in κ-features. This forces the ECM subject to raise to c-command the matrix *v*, keeping it from staying within the *to*-infinitive. Thus, the examples below are ruled out, as long as unvalued features are illegible at the interfaces.

(28) a. * Mary proved to be **him** innocent.
b. * Mary proved to **him** be innocent.

Let us now offer the clearest evidence that the ECM subject overtly moves into the matrix clause. It comes from the *particle verb* construction, as Johnson (1991) and Koizumi (1995) point out. For example, consider the particle verb *make out*, which may select both finite and infinitival clausal complements. The point is that the finite subject cannot precede the particle *out*, but the ECM subject can.

---

[7] In (27), we assume that the *to*-infinitive is not an argument introduced by the head *v*, but a kind of modifier for the root V, so it occurs within VP. This is a plausible analysis under Elliot's (2018) semantics of clausal complementation, based on Kratzer (2006) and Moulton (2015), under which an embedding V and its sister clause are of the same semantic type and combined via Heim and Kratzer's (1998) *Predicate Modification.*

(29) a. They're trying to *make out* that **John** is a liar.
b. ? They're trying to *make out* **John** to be a liar. (Koizumi 1995: 35)

(30) a. * They're trying to *make* that **John** *out* is a liar.
b. They're trying to *make* **John** *out* to be a liar. (Koizumi 1995: 35)

Since the particle *out* is in the matrix clause, this contrast suggests that the ECM subject moves to the edge of the matrix *v*P, given that it has accusative.[8] We then predict that the matrix V and the ECM subject hold the CAE, allowing no AdvP to intervene. This is borne out. For instance, (31) and (32) show that the ECM subject must move over a matrix PP, and that no AdvP occurs above the moved DP.

(31) a. We **proved Smith** *to the authorities* to be the thief.
b. * We **proved** *to the authorities* **Smith** to be the thief. (Bowers 1993: 632)

(32) a. We **proved Smith** *conclusively* to be the thief.
b. * We **proved** *conclusively* **Smith** to be the thief. (Bowers 1993: 632)

In summary, we have shown that the CAE is characterized by two structural preconditions; the CAE holds between V and its object DP iff (i) V is in the position of Voice, and (ii) the DP is in the κ-position of *v*P. This formulation leads us to revise the CAE as a *category-free* generalization like below.

(33) **Case Adjacency Effect (CAE)** (*revised*)
No AdvP occurs between V and XP iff V is in Voice, and XP is in a κ-position within VoiceP.

That is, we need not identify the category of XP, which corresponds to the object DP or the ECM subject, because it is only important to know whether XP needs to occur in a κ-position. This revision amounts to saying that no AdvP occurs above any XP in the κ-position of *v*P, as schematized below.

(34) **The distribution of AdvPs in the single object VoiceP**
$[_{\text{VoiceP}}$ Voice-*v*-**V** $[_{vP}$ *AdvP $[_{v'}$ **DP**$_{[\kappa:\varnothing]}$ $[_{v'}$ $t_{v[\kappa:\text{acc}]}$ $\boldsymbol{t}_{\mathbf{V}}$]]]]

2.2. Corroboration: The Double Object Construction

We then corroborate the validity of our revised CAE by shifting our focus to ditransitive verbs, such as *give*, *send*, and *throw*. One property of these verbs is that most of them have two frames for introducing their internal arguments: namely, the DP-DP frame and the DP-PP frame. As observed by Stowell (1981), both frames also hold the CAE in its original sense, which is shown below.

(35) a. John *secretly* **gave Mary the ring** yesterday. (DP-DP frame)
b. * John **gave** *secretly* **Mary the ring** yesterday.
c. * John **gave Mary** *secretly* **the ring** yesterday.
d. John **gave Mary the ring** *secretly* yesterday.

---

[8] Richards (1997: 146, fn.8) reports that examples such as (29b) are "quite bad," where the ECM subject follows the particle. If this is true, then the ECM subject obligatorily moves into the matrix clause, contra Lasnik's (2003) conclusion that raising to object is optional.

(36) a. John *secretly* **gave the ring to Mary** yesterday. (DP-PP frame)
b. * John **gave** *secretly* **the ring to Mary** yesterday.
c. John **gave the ring** *secretly* **to Mary** yesterday.
d. John **gave the ring to Mary** *secretly* yesterday.

Here, the verb *give* must be adjacent to its first argument DP in both frames, as shown in (35b) and (36b). Interestingly, the DP-DP frame does not even allow an AdvP to occur between the verb and its second argument DP, as the contrast between (35c) and (36c) indicates.

It should then be asked if our revised CAE can capture the distribution of AdvPs relative to ditransitive Vs. In particular, is it possible to keep our generalization for the DP-DP frame, saying that no AdvP occurs above a κ-position inside VoiceP? Below, we argue that it is possible, if we analyze the syntax of the DP-DP frame in line with Marantz (1993), whose idea has been advanced by Pylkkänen (2002), Basilico (2008), Georgala (2012), Legate (2014), Jerro (2016), and Bruening (2001, 2010, 2015, 2021).

To begin with, we assume that the VoiceP in the DP-DP frame has the head *Applicative* π, a category invented by Marantz (1993). In our view, the English π head takes a root V as its sister, introducing a direct object (DO) and indirect object (IO) to its own edge (cf. Pylkkänen 2002: 21ff.), and the entire πP is embedded by a verbalizer *v*, which we suppose encodes a sort of causative semantics (cf. Legate 2014: 124ff.).[9] With these properties, the DP-DP frame is structured as follows, where V-raising is omitted.

(37) a.	Base:	$[_{\text{VoiceP}}$	Voice	$[_{v\text{P}}$			$v_{[\kappa:\text{acc}]}$	$[_{\pi\text{P}}$	**IO**	[	**DO**	[	$\pi_{[\kappa:\text{acc}]}$	V]]]]]
b.	Movement:	$[_{\text{VoiceP}}$	Voice	$[_{v\text{P}}$	**IO**	[	$v_{[\kappa:\text{acc}]}$	$[_{\pi\text{P}}$	$\boldsymbol{t}_{\textbf{IO}}$	[	**DO**	[	$\pi_{[\kappa:\text{acc}]}$	V]]]]]

Note that the heads *v* and π are both assumed to have an accusative κ-feature [κ: acc], at least in English. Under Bošković's (2007) κ-valuation system, this means that DO can be κ-valued in situ through Agree with the head π. In contrast, IO cannot undergo Agree with the head π, because DO is a probe closer to π and counts as an intervener that prevents Agree between IO and π (cf. Chomsky 2000, 2001). Thus, IO must move to the edge of *v*P so that it can c-command, and hold Agree with, the head *v*.

This analysis of the DP-DP frame is supported by Runner's (1998) findings again. For instance, he shows that DO and a modifier MO can form a conjunct excluding IO as in (38a), and MOs alone can do so excluding IO and DO as in (38b). In our analysis, the conjuncts in (38a) are πP, out of which a single IO moves across-the-board, while those in (38b) are VPs, as roughly represented in (39).

(38) a. I gave John [**the book** in the morning] and [**the magazine** in the evening].
b. I gave Greg a gift [**on purpose** last Christmas] but [**only reluctantly** this year].
(Runner 1998: 132–133)

[9] Pylkkänen's (2002) original analysis classifies the English π head as a case of *low applicatives*, which she takes to occur under VP and encode a *transfer-of-possession* relation between IO and DO. As Georgala (2012) notes, however, placing the π head in the low position is not well-motivated morphologically. In addition, Basilicon (2008) and Jerro (2016) argue that an applicative head occurring above VP, namely a *high applicative* in Pylkkänen's terms, may also express a transfer-of-possession relation. These facts suggest that it is a viable option to place the English π head above VP, as we assume here. Note that Pylkkänen's strongest argument for her low applicative syntax is that it can derive the unavailability of IO for *secondary depictive predication*, but see Bruening (2015), who develops an analysis that is compatible with our high applicative syntax.

(39) a. [$_{vP}$ IO [**[**$_{\pi P}$ t_{IO} **DO** [$_{VP}$ MO t_V]**]** & **[**$_{\pi P}$ t_{IO} **DO** [$_{VP}$ MO t_V]**]**]]
b. [$_{vP}$ IO [$_{\pi P}$ DO [**[**$_{VP}$ **MO** [$_{V'}$ MO t_V]**]** & **[**$_{VP}$ **MO** [$_{V'}$ MO t_V]**]**]]]

Moreover, given that NPIs must be c-commanded by negative elements, it should follow that a negative IO can license an NPI in the DO position, while a negative DO can license an NPI in a modifier position within VP. This is because our analysis entails that IO c-commands DO, which in turn c-commands everything in VP. This prediction is borne out, as shown below.

(40) a. I gave **no one** *anything*.
b. * I gave *anyone* **nothing**. (Runner 1998: 128)

(41) a. Laverne gave Shirely **nothing** *at any of these events*.
b. * Laverne gave Shirely *anything* **during none of these events**. (Runner 1998: 130)

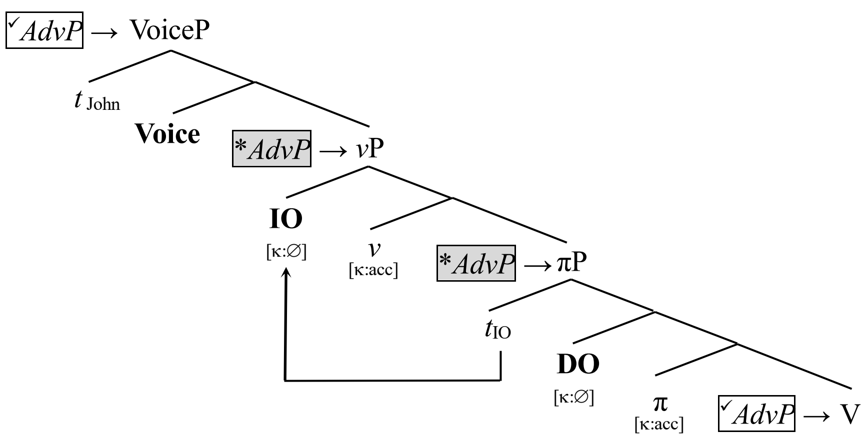
Note that these facts not only lend support to our analysis of the double object VoiceP, but they can also be used to argue against some alternative analyses, such as Harley's (2002) and Pylkkänen's (2002). In Pylkkänen's analysis, for example, the English π head is supposed to occur below VP, taking DO as its sister and IO as its specifier, as shown in (42a).

(42) a. Pylkkänen's (2002) analysis: [$_{VoiceP}$ Voice [$_{\mathbf{VP}}$ V [$_{\pi P}$ **IO** [$_{\pi'}$ π **DO**]]]]
b. Harley's (2002) analysis: [$_{VoiceP}$ Voice [$_{\mathbf{VP}}$ V [$_{PP}$ **IO** [$_{P'}$ P_{HAVE} **DO**]]]]

An immediate problem with Pylkkänen's structure is how it can capture, for example, the contrast in (41), since it does not enable DO to c-command VP or any modifiers attached to V; see also footnote 9 for its potential drawbacks. The same criticism holds for Harley's (2002) analysis, too, under which the head π in (42a) is simply replaced by a null preposition P_{HAVE}, as shown in (42b).

Given this result, suppose that our analysis of the DP-DP frame is basically right. Then, our structure of the DP-DP frame allows us to describe the impossible sites for AdvPs in a desirable way, because they can be identified with the edge of *v*P and πP, as the following diagram indicates.

(43) a. John (✓*secretly*) **gave** (**secretly*) **Mary** (**secretly*) **the ring** (✓*secretly*) yesterday. (= (35))
b.



Thus, since the edge positions of *v*P and πP are both κ-positions in the sense that their heads have a valued κ-feature, we can apply our revised CAE to the DP-DP frame, too. That is, it is possible to maintain that no AdvP occurs above any XP in the κ-positions of *v*P and πP, as roughly represented below.

(44) **The distribution of AdvPs in the double object VoiceP**

a. $[_{\text{VoiceP}}$ **Voice-*v*-π-V** $[_{vP}$ *AdvP $[_{v'}$ **IO**$_{[\kappa:\varnothing]}$ $[_{v'}$ $t_{v[\kappa:\text{acc}]}$ πP]]]]

b. $[_{vP}$ **IO** $[_{v'}$ t_v $[_{\pi P}$ *AdvP [… **DO**$_{[\kappa:\varnothing]}$ $[_{\pi'}$ $t_{\pi[\kappa:\text{acc}]}$ $\boldsymbol{t}_{\mathbf{V}}$]]]]]

Let us now turn to the DP-PP frame. Here, we assume that it is not derivationally related to the DP-DP frame, and that they have distinct base structures, which has been a popular view since Marantz (1993) and Pesetsky (1995) (e.g., Harely 2002, Bruening 2001, 2010, 2021). In particular, the VoiceP in the DP-PP frame lacks πP, and the object DP is introduced by the thematic *v**, while the PP is combined with the root V. With these premises, we assume that the DP-PP frame is derived as follows.

(45) a. Base: $[_{\text{VoiceP}}$ Voice $[_{vP}$ **DP**$_{[\kappa:\varnothing]}$ $[_{v'}$ $v^*_{[\kappa:\text{acc}]}$ $[_{\text{VP}}$ **PP** V]]]]]

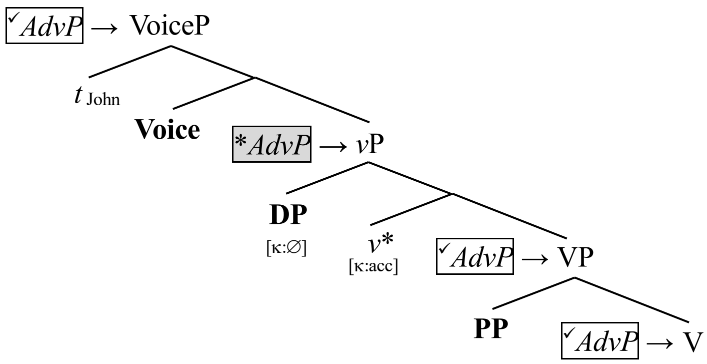
b. V-raising: $[_{\text{VoiceP}}$ Voice-*v**-V $[_{vP}$ **DP**$_{[\kappa:\varnothing]}$ $[_{v'}$ $t_{v^*[\kappa:\text{acc}]}$ $[_{\text{VP}}$ **PP** t_{V}]]]]

Note that the VoiceP in the DP-PP frame is structurally identical to the single object VoiceP with a PP modifier, with the only difference being that the root V in the DP-PP frame is ditransitive and its PP is *to*-PP.[10] Although Hallman (2015) suggests that the DP-PP frame may be associated with more than one base structure, it suffices for our purposes to adopt only that in (45), which is at least motivated by Larson's (1988) observation that the DP c-commands the PP in the DP-PP frame.

With this structure, we can maintain our revised CAE for the DP-PP frame, too. Specifically, it allows us to describe the distribution of AdvPs in the DP-PP frame as we want, because their impossible site can be identified with the edge of *v*P. This is indicated in the following diagram.

(46) a. John (✓*secretly*) **gave** (**secretly*) **the ring** (✓*secretly*) **to Mary** (✓*secretly*) yesterday. (= (36))

b.



Here, it is important to keep in mind that the head *v** has a valued κ-feature and makes its edge position a κ-position. Our revision of the CAE therefore seems to be on the right track, which says that no AdvP occurs above a κ-position within VoiceP, namely at the edge of *v*P or πP.

[10] Although *to*-PP in the DP-PP frame is often treated as an argument (e.g., Larson 1988), it is a kind of modifier in our view, since we analyze all PPs as such. Thus, *to*-PP is essentially optional in the sense that it is not required by the semantics of its verb. This view seems to be on the right track. For instance, *give* may drop its Goal argument as long as it is pragmatically recoverable. See Bruening (2021) for relevant discussion.

2.3. Generalization: The Anti-stacking Effect

In sum, we have described some structural restrictions on AdvP placement, which are beyond the literal scope of Stowell's (1981) adjacency-based account. That is, within VoiceP, no AdvP occurs above any DP at the edge of *v*P or πP. The schematic representations of the restrictions are repeated here.

(47) a. $[_{\text{VoiceP}}\ \text{Voice-}v\text{-}\mathbf{V}\ [_{vP}\ \boxed{\text{*AdvP}}\ [_{v''}\ \mathbf{DP}_{[\kappa:\varnothing]}\ [_{v'}\ t_{v[\kappa:\text{acc}]}\ \boldsymbol{t}_{\mathbf{V}}\ ]]]]$

b. $[_{\text{VoiceP}}\ \text{Voice-}v\text{-}\pi\text{-}\mathbf{V}\ [_{vP}\ \boxed{\text{*AdvP}}\ [_{v''}\ \mathbf{IO}_{[\kappa:\varnothing]}\ [_{v'}\ t_{v[\kappa:\text{acc}]}\ \pi\text{P}\ ]]]]$

c. $[_{vP}\ \mathbf{IO}\ [_{v'}\ t_{v}\ [_{\pi P}\ \boxed{\text{*AdvP}}\ [\ldots\ \mathbf{DO}_{[\kappa:\varnothing]}\ [_{\pi'}\ t_{\pi[\kappa:\text{acc}]}\ \boldsymbol{t}_{\mathbf{V}}\ ]]]]]$

Importantly, these restrictions can be unified as a more general ban, since they share one structural description; all of them exclude AdvPs that occur right above κ-positions. We call this general ban an *Anti-stacking Effect* (ASE), which says that no AdvP occurs at the edge of a κ-assigner head H.

(48) **Anti-stacking Effect (ASE)**

$[_{\text{HP}}\ \boxed{\text{*AdvP}}\ [_{\text{H}''}\ \mathbf{XP}_{[\kappa]}\ [_{\text{H}'}\ \text{H}_{[\kappa]}\ [\ \ldots\ ]]]]$

Given the ASE, we are now able to discuss the issue of how the input search for Merge is restricted. Recall that there are only two possible views, *Total Search* and *Partial Search*. Under Total Search, the input search is free, and Merge can combine any two SOs. Thus, this view must invoke *filters on the outputs of derivations*, which may rule out the output of Merge(AdvP, H″). Under Partial Search, in contrast, the input search may be limited, so there can be some SO that Merge cannot select. As a result, this view may invoke *filters on the inputs to derivations* in order to rule out the inputs in Merge(AdvP, H″). The question is which view allows us to derive the ASE in a principled fashion.

To narrow down the range of possible approaches, we note that the ASE should not be analyzed by positing a *semantic output filter*. For instance, one might try to block merging an AdvP at the edge of *v*P by assuming that their semantic composition results in a type mismatch. However, such a solution is not general enough, because under a compositional semantics for the split-V structure, the maximal projections of V, *v*, π, and Voice can all be analyzed as denoting the same type of predicates, so it is unclear why some of them cannot host predicate modifiers like *slowly* and *secretly*. To make this point clearer, it helps to consider how the single object VoiceP can be compositionally interpreted.

Given Heim and Kratzer's (1998) framework, suppose that the semantic types of SOs include *individual* (type **e**), *event* (type **v**), *truth value* (type **t**), *event predicate* (type ⟨**v, t**⟩), and *thematic relation* (type ⟨**e,** ⟨**v, t**⟩⟩). Then, assuming that proper or definite DPs are expressions of type **e**, we specify Voice and the thematic *v** as expressions of type ⟨⟨**v, t**⟩, ⟨**e,** ⟨**v, t**⟩⟩⟩, and a root V like *read* as an expression of type ⟨**v, t**⟩. The denotations of these elements are given below.[11]

(49) a. $[\![\text{Voice}]\!] = \lambda P.\lambda x.\lambda e.\ [\text{AG}(e) = x \wedge P(e)]$ "the Agent of *e* is *x* and *e* is an event of *P*"

b. $[\![v^*]\!] = \lambda P.\lambda x.\lambda e.\ [\text{TH}(e) = x \wedge P(e)]$ "the Theme of *e* is *x* and *e* is an event of *P*"

c. $[\![\text{V}_{read}]\!] = \lambda e.\ [read(e)]$ "*e* is an event of reading"

[11] As for the athematic *v*, we simply analyze it as an *identify function* in event predicates. That is, it is an expression of type ⟨⟨**v, t**⟩**,** ⟨**v, t**⟩⟩, like $[\![v]\!] = \lambda P.\lambda e.\ [P(e)]$, and it does not introduce any individual argument, unlike Voice and *v**. Still, adapting Folli and Harley's (2007) approach, it is possible to encode an additional "flavor" in the meaning of the athematic *v* and divide it into several subtypes, such as v_{action}, v_{cause}, etc.

Note that our semantics of Voice and *v** allows them to combine with predicates of type **⟨v, t⟩** just via *Function Application*. This means that Kratzer's (1996) additional rule of *Event Identification* is unnecessary, since its supposed job is done by assuming Voice and *v** to be of type **⟨⟨v, t⟩, ⟨e, ⟨v, t⟩⟩⟩**, as Bruening (2015, 2021) suggests. Thus, the meaning of the single object VoiceP, like *Mary read the book*, is simply derived by Function Application, as shown below.

(50)

- ⟦VoiceP⟧ ⟨v, t⟩ : λ*e*. [AG(*e*) = *Mary* ∧ TH(*e*) = *the book* ∧ *read*(*e*)]
  - ⟦DP⟧ e : *Mary*
  - ⟦Voice′⟧ ⟨e, ⟨v, t⟩⟩ : λ*x*.λ*e*. [AG(*e*) = *x* ∧ TH(*e*) = *the book* ∧ *read*(*e*)]
    - ⟦Voice⟧ ⟨⟨v, t⟩, ⟨e, ⟨v, t⟩⟩⟩ : λ*P*.λ*x*.λ*e*. [AG(*e*) = *x* ∧ *P*(*e*)]
    - ⟦*v*P⟧ ⟨v, t⟩ : λ*e*. [TH(*e*) = *the book* ∧ *read*(*e*)]
      - ⟦DP⟧ e : *the book*
      - ⟦*v*′⟧ ⟨e, ⟨v, t⟩⟩ : λ*x*.λ*e*. [TH(*e*) = *x* ∧ *read*(*e*)]
        - ⟦*v**⟧ ⟨⟨v, t⟩, ⟨e, ⟨v, t⟩⟩⟩ : λ*P*.λ*x*.λ*e*. [TH(*e*) = *x* ∧ *P*(*e*)]
        - ⟦V_{read}⟧ ⟨v, t⟩ : λ*e*. [*read*(*e*)]

The final output ⟦VoiceP⟧, if existentially closed, conveys the proposition ∃*e*. [AG(*e*) = *Mary* ∧ TH(*e*) = *the book* ∧ *read*(*e*)], which says that there is an event *e* such that its Agent is *Mary*, its Theme is *the book*, and it is an event of reading. Note that this resulting semantics is not an outlandish proposal, and it can be regarded as a variant of Kratzer's (1996). The point here is that the maximal projections of V, *v*, and Voice all denote predicates of type **⟨v, t⟩**, and they constitute a natural class type-theoretically.

It is now clear why it is semantically odd that the maximal projection of the κ-assigner head *v** cannot host *manner AdvPs*, such as *slowly* and *secretly*. The standard analysis of these AdvPs is to define them as *predicate modifiers* of type **⟨⟨v, t⟩, ⟨v, t⟩⟩** and the meaning of *slowly* looks like λ*P*.λ*e*. [*slow*(*e*) ∧ *P*(*e*)] (e.g., Davidson 1967, Parsons 1990). Thus, given that V, *v*P, and VoiceP are all of type **⟨v, t⟩**, *slowly* should be able to combine with any of them via Function Application, as shown below.

(51) a.
- ⟦VP⟧ ⟨v, t⟩
  - ⟦*slowly*⟧ ⟨⟨v, t⟩, ⟨v, t⟩⟩
  - ⟦V_{read}⟧ ⟨v, t⟩

b.
- ⟦*v*P⟧ ⟨v, t⟩
  - ⟦*slowly*⟧ ⟨⟨v, t⟩, ⟨v, t⟩⟩
  - ⟦*v*P⟧ ⟨v, t⟩

c.
- ⟦VoiceP⟧ ⟨v, t⟩
  - ⟦*slowly*⟧ ⟨⟨v, t⟩, ⟨v, t⟩⟩
  - ⟦VoiceP⟧ ⟨v, t⟩

However, the *v*P modification pattern in (51b) is impossible when the head *v** is a κ-assigner, as we have seen in examples like below, and in this case, *slowly* can only modify VP and VoiceP.

(52) * Mary [$_{\text{VoiceP}}$ **read** [$_{vP}$ *slowly* [$_{v''}$ **the book** [$_{v'}$ …]]] yesterday.

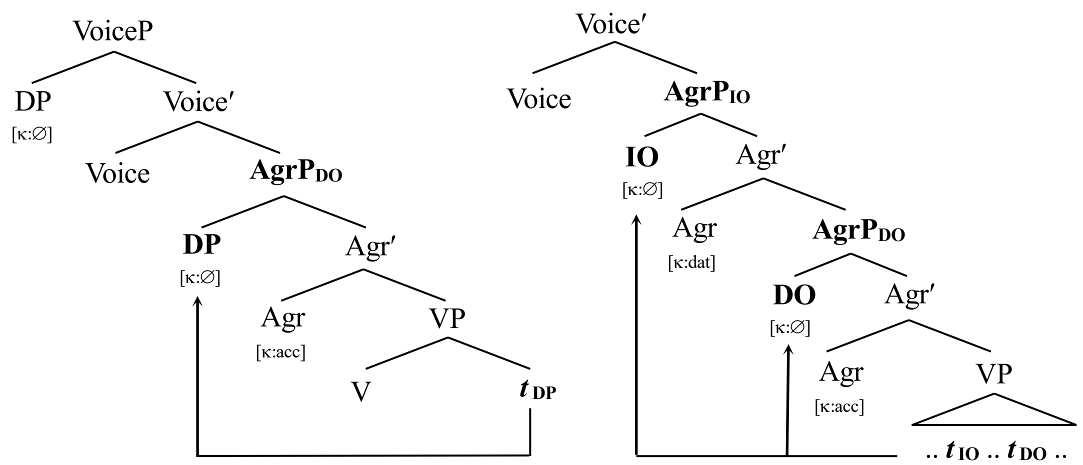
This way, as long as our semantics of the single object VoiceP and manner AdvPs is on the right track, the cause of the ASE should be found in, not the semantics, but the syntax of *v*P. This seems to be a right direction for the analysis, because the syntax appropriately distinguishes *v*P from VP and VoiceP; semantically, all of them are of type **⟨v, t⟩** and potential loci of modification, but syntactically, only *v*P (and πP) can be a locus of κ-valuation. In other words, unless we have a semantic theory that explicitly distinguishes *v*P from VP and VoiceP, we should conclude that the ASE is caused by κ-valuation.

We now turn to previous accounts of the CAE or ASE. In fact, all of them posit a kind of *syntactic output filters* (e.g., Chomsky 1981, Stowell 1981, Pollock 1989, Pesetsky 1989, Johnson 1991, Bowers 1993, Koizumi 1995, Runner 1998, Janke and Neeleman 2012, Belk and Neeleman 2017). As far as the author judges, Koizumi's (1995) account is the most successful, since it can derive all the restrictions on AdvP placement above; see Koizumi (1995: 16ff.) for a critical review of other accounts. However, his approach is not free from a problem. To show this point, let us review his proposal briefly.

2.4. Comparison: Koizumi's (1995) Agr-based Account

Koizumi (1995) begins by revising Chomsky's (1993) *Agreement-based Case Theory*. This theory posits *Agreement Phrases* (AgrP) as loci of κ-valuation, which is assumed to apply in a spec-head relation. In short, Koizumi's proposal is that the English Agr head overtly triggers κ-driven movement, and that the VoiceP may contain at least two instances of AgrP; (i) $AgrP_{DO}$, which occurs above VP and attracts the direct object, and (ii) $AgrP_{IO}$, which occurs above $AgrP_{DO}$ and attracts the indirect object. Thus, the single object, direct object, and indirect object appear in the following positions.

(53) a. Agr-*based* single object VoiceP b. Agr-*based* double object VoiceP



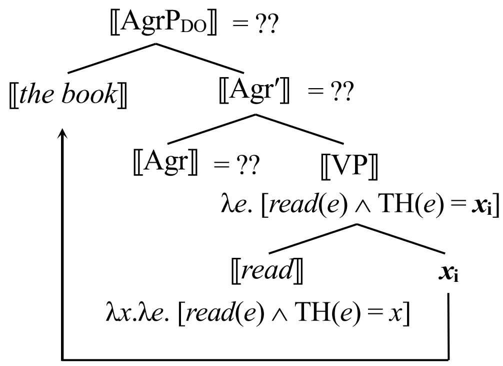
Crucially, Koizumi (1995: 28, ft.12) assumes that, as a "pure functional category," the head Agr has no lexical semantic content and lacks the ability to license the occurrence of an AdvP. He then suggests that the impossible sites for AdvPs described so far are all projected by Agr. This means that he invokes an output filter to account for the ASE; that is, he derives all the restrictions on AdvPs by assuming that no AdvP is licensed within projections of Agr. This uniform account can be schematized as follows.

(54) **Koizumi's (1995) Agr-based account of the ASE**

$[_{AgrP}$ *AdvP $[_{Agr''}$ **XP**$_{[\kappa]}$ $[_{Agr'}$ Agr$_{[\kappa]}$ […]]]]

However, though Koizumi's formulation of the ASE is already explanatory, it has a theoretically problematic aspect; that is, it relies on the uninterpretable category AgrP. The postulation of AgrP itself is certainly a theoretical option, but it is an assumption that requires us to make a tradeoff; while deriving the ASE, the recourse to AgrP makes it extremely difficult to maintain the compositionality of interpretation. To illustrate, consider an Agr-based structure for the transitive VP *read the book*.

(55)



The question is how the moved DP in AgrP is compositionally interpreted, and the simplest answer is that it cannot. This is because Agr lacks lexical semantic content, unlike V, *v*, π, and Voice, which is why AgrP can license no AdvP. Of course, there would be a way to reconcile Agr-based syntax with compositional semantics, but our point is that Koizumi's account is not without a theoretical issue.

In fact, Koizumi's theory also has an empirically problematic aspect that concerns *freezing effects*. Since Wexler and Culicover (1980), it has been widely argued that extraction out of moved SOs is impossible. A good illustration is given by Lasnik and Park (2003), who point out that an in-situ subject in the *there* constriction allows extraction out of it, but its moved counterpart does not, as we see below.

(56) a. *Which candidate*$_i$ were there **[posters of** t_i**]** all over the town?
b. * *Which candidate*$_i$ were **[posters of** t_i**]** all over the town? (Lasnik and Park 2003: 651)

Interestingly, the same restriction is found in the ECM construction, as Narita (2014) observes.

(57) a. *Which candidate*$_i$ did you believe there to be **[posters of** t_i**]** all over the town?
b. * *Which candidate*$_i$ did you believe **[posters of** t_i**]** to be all over the town? (Narita 2014: 108)

This contrast can be captured under our analysis, because we assume that the ECM subject moves to the edge of the matrix *v*P, so the unacceptable ECM case is ruled out by whatever principles derive freezing effects; see Bošković (2018) for a recent proposal. Note that the same account is also available to Koizumi's approach, which analyzes the ECM subject as raising to the edge of the matrix AgrP_{DO}.

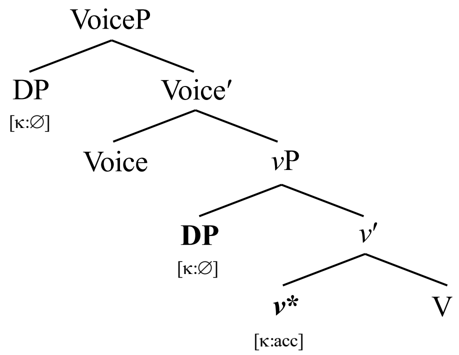
Still, what is at stake here is Koizumi's uniform claim that the single, direct, and indirect object DPs all move to the edge of AgrPs. Accordingly, it should follow that all the DPs exhibit freezing effects, disallowing extraction out of themselves. The facts contrary to this prediction are given below.

(58) a. *Who*$_i$ did John select **[a picture of** t_i**]**? (Single object)
b. * *Who*$_i$ was **[a picture of** t_i**]** selected? (Passive subject)
c. * *Who*$_i$ did **[a picture of** t_i**]** frighten John? (Active subject)

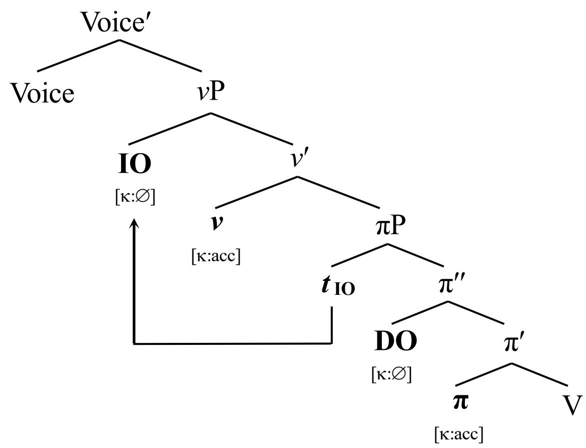
(59) a. *Who*$_i$ did you give the child **[a picture of** t_i**]**? (Direct object)
b. *Who*$_i$ did you give **[a picture of** t_i**]** to the child? (DP-PP frame)
c. * *Who*$_i$ did you give **[a child of** t_i**]** the picture? (Indirect object)

Our analysis, on the other hand, can predict all these facts, since it assumes that the finite subject, ECM subject, and indirect object DP move, but the single or direct object DP does not, as shown below. Thus, Koizumi's Agr-based account of the ASE relies on the dubious assumption that all DPs move.

(60) a. Agr-*less* single object VoiceP



b. Agr-*less* double object VoiceP



In summary, we have argued that Koizumi's (1995) approach to the ASE has both theoretically and empirically problematic aspects. Though the incompatibility of his Agr-based syntax with compositional semantics may be overcome somehow, it entails that DPs cannot stay in situ and must move to the edge of AgrPs, making the incorrect prediction that all of them should show freezing effects. Given this, the rest of this paper develops an Agr-less account of the ASE that eliminates the Agr-based output filter altogether to maintain the compositionality of interpretation. Specifically, we will derive the ASE by proposing a *syntactic input filter* that restricts the input search for Merge.

3. Explanation

3.1. Framework: Theories of Agreement and Derivation

Our account of the ASE is based on Chomsky's (2001) theory of Agree. In particular, we adopt Chomsky's view that Agree is a feature valuation operation conditioned by the relation of *feature matching*, and we propose that this relation plays a key role in deducing the ASE. In short, we will reformulate feature matching as a procedure that restricts the input search for Merge, as outlined below.

(61) Feature matching within one input to Merge restricts the search for the other input.

To elaborate on this proposal, this section introduces the basics of Agree and its developments based on the operation *Transfer* (e.g., Narita 2014). In doing so, we also provide a theory of *derivation* and *workspace* (e.g., Chomsky 2019) and then elucidate the structure of AdvP in line with Distributed Morphology (e.g., Corver 2021). These theories will jointly help understand the nature of our proposal.

First of all, Chomsky (2001) proposes that Agree applies to two LIs, a probe P and a goal G, and assigns the value of ^{v}f to ^{u}f, where ^{u}f is an unvalued feature in P and ^{v}f is a *matching* valued feature in G. Agree is therefore based on a feature matching relation *Match*, which holds between two LIs.

(62) For any lexical items X, Y, Match(X, Y) iff
  a. either X or Y has an unvalued feature ^{u}f, and
  b. the other has a valued feature ^{v}f of the same type as ^{u}f.

This formulation of Match reflects Chomsky's (2001: 6) proposal that it only holds for valued-unvalued pairs of LIs. For example, Match holds for valued-unvalued pairs, such as ($X_{[\kappa:\,acc]}$, $Y_{[\kappa:\,\varnothing]}$), but it does not hold for identical-value pairs, such as ($X_{[\kappa:\,acc]}$, $Y_{[\kappa:\,acc]}$).

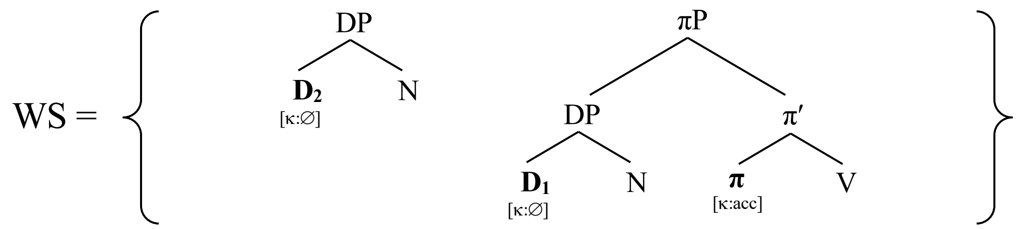
Importantly, we assume that the application of Match is not conditioned by any structural relations like *c-command*, so it may apply to LIs even if they exist within separate SOs (cf. Chomsky 2001). Still, this does not mean that LIs may hold Match anywhere. After all, Match is a syntactic process and it is not available for LIs stored in the lexicon. Thus, Match has at least the following condition to meet.

(63) For LIs to hold Match, they must occur in the workspace of the same derivation.

To begin with, we take a derivation as a sequence of syntactic operations (OPs) (cf. Collins 2017). Then, the workspace (WS) may be viewed as a kind of working memory that stores the outputs of earlier OPs and supplies them as inputs to later OPs. In other words, each derivation updates its own WS step by step and carries it as an input resource along with the lexicon Lex, as stated below.

(64) A derivation is a sequence of OPs $<OP_1, \ldots, OP_n>$, where
  a. each OP is a syntactic operation such as Merge, and
  b. each OP takes its input from the Lex or WS at its stage.

We now assume with Chomsky (1995, 2019) that the WS is minimal; that is, it is the set of all SOs that have been created by the current stage and excludes those which have been taken as inputs to Merge.[12] For example, suppose a derivation has created two complex SOs, say, an intermediate πP and an indirect object DP. Then, the WS at this stage consists of only these two outputs, as we see in the following.

(65)


The point here is that our theory regards this WS as containing two matching pairs of LIs, namely ($\pi_{[\kappa:\,acc]}$, $D_{1[\kappa:\,\varnothing]}$) and ($\pi_{[\kappa:\,acc]}$, $D_{2[\kappa:\,\varnothing]}$). Of course, the second pair members, π and D_2, do not have their projections in a c-command relation and do not even exist within the same SO. However, they still occur in the same WS, so they count as holding Match, according to the conditions in (62) and (63).

In contrast, the application of Agree is conditioned more severely. For one thing, P or its projection must c-command G, as we have assumed with Bošković (2007). It has also been proposed in the literature that Agree is further conditioned in terms of *Transfer*, which Chomsky (2004) posits as an operation that maps SOs to phonetic and semantic forms. For instance, adapting Narita's (2014) proposal, we assume the following idea (cf. Epstein and Seely 2002, Hiraiwa 2005, Richards 2007, Chomsky 2008).

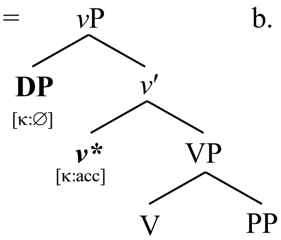
[12] We call this model of WS a *removal model*, since it requires Merge(X, Y) to add its output {X, Y} to the WS and remove the inputs X, Y from the WS. One alternative is Bobaljik's (1995) *union model*, where Merge adds its output to the WS and does nothing else, so never reducing the membership of the WS. For further relevant discussion, see Collins and Stabler (2016), Collins (2017), and Chomsky et al. (2019).

(66) Agree is a part of Transfer, and they apply at derivationally simultaneous points.

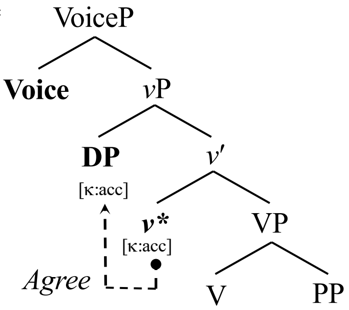
This Agree-in-Transfer model accords with recent work by Arregi and Nevins (2012), who analyze feature valuation as a post-syntactic process in the syntax-morphology mapping. This view is indeed a theoretical option, and also empirically reasonable in that the output of feature valuation must be determined morphologically. In any case, the Agree-in-Transfer model has a significant consequence on the timing of Agree; that is, its application within a SO is "delayed" until that SO undergoes Transfer.

The question then is when Transfer occurs. Under Chomsky (2004), it applies to *phases*, converting them in full or their complements into phonetic and semantic forms. Here, suppose that VoiceP is a phase, whose head we call a *phase head*. It then follows that Transfer occurs only when a phase head is merged into the structure, and in the Agree-in-Transfer model, this means that Agree does not apply until that timing. For example, consider the following derivation, which is creating a single object VoiceP.

(67) a. Merge(DP, *v*′) =



b. Merge(Voice, *v*P) =



At stage (a), *v** and DP hold Match in the sense that they make a valued-unvalued pair, but they cannot undergo Agree, since no phase head is introduced. At stage (b), however, they appear within the phase VoiceP, so they can undergo Agree when the sister of the phase head Voice is transferred. Thus, our theory allows a matching pair to undergo Agree only after it is embedded in a phase.[13] This result amounts to saying that a matching pair holds Match until a phase head is merged into the structure.

Finally, it is important to clarify the structure of AdvP in terms of feature matching. In the literature, it has been argued that adverbs are not simple LIs at the level of syntax (e.g., Rubin 1996, Alexeyenko 2015, Corver 2021), so we assume that they are decomposable like verbs. Specifically, under Marantz's (1997) theory of Distributed Morphology, we analyze an adverb into a root and a categorizer, and call the latter a *modifier head* μ. Thus, all adverbs like *secretly* are syntactically complex as shown below.

(68)

One may identify the suffix -*ly* as a realization of the head μ, but see Baker (2003), Corver (2014), and Alexeyenko (2015) for a different view on the status of -*ly*. What is important here is that, while DP contains syntactic features like φ-features and κ-features, AdvP (i.e., μP) does not contain any of them, so it

---

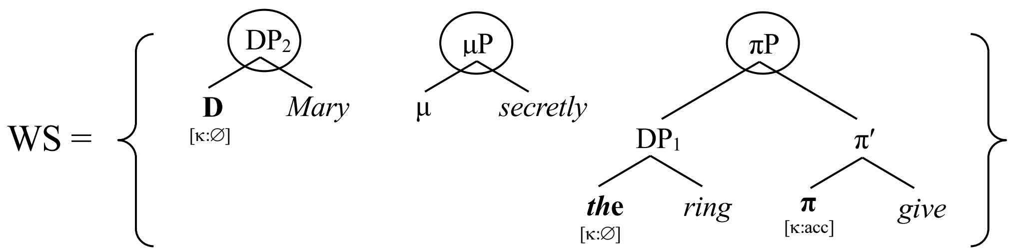
[13] In the literature, it has been suggested that PP is also a phase (e.g., Drummond, Hornstein, and Lasnik 2010, Bošković 2014, Narita 2014). If this is true, we assume that PP is also split, like $[_{pP}$ *p* $[_{PP}$ P DP]] (e.g., Svenonius 2003, 2010), and that the entire *p*P is a phase with the head P given a valued κ-feature. This means that P and DP make a matching pair, but they can also undergo Agree only after the phase head *p* is merged.

never holds Match with the head *v*, for example.[14] With this assumption, namely that AdvPs are complex SOs with no matchable features, we now make a proposal on the input search for Merge.

3.2. Proposal: A Matching Search Condition

Our proposal is built on the view of the WS as a working memory. Given that the input search of the WS is a computational process that activates its stored data, we assume that the search domain should be kept minimal, because the smaller it is, so is the computational load; that is, *minimal search* is a principle of computational efficiency, requiring the input search of the WS to select less embedded inputs among appropriate ones (cf. Chomsky 2013, 2015). Still, if we simply apply this principle to the input search for Merge, the accessible inputs in the WS will be only its *members*, since they are the least embedded SOs (i.e., roots). For example, suppose that a derivation has reached the following WS.

(69)



Here, the circled SOs are members of the WS (= {DP_2, μP, πP}), and the minimal search principle should make all non-member terms like DP_1 invisible as inputs to Merge, since they are more embedded SOs than the members. This means that, everything else being equal, the minimal input search for Merge always ends up being *member search*, and it would allow the mode of EM (i.e., Merge of roots), but not the mode of IM (i.e., Merge of a root and a non-root). Thus, we need a system that enables the input search to satisfy the minimal search principle without impairing the movement property of Merge.

What we then propose is that feature matching (i.e., Match) is in fact a procedure that activates a certain subset of the term SOs of the WS, and forces the input search for Merge to focus on that subset. More specifically, we suggest that the input search for Merge is restricted by the following condition.

(70) **Matching Search Condition (MSC)**
For any member X of the WS, if the head of X matches some LI inside X,
X can merge with any term Y of the WS, only if Y is in *the match set* for X.

This means that, if the head of X holds Match inside X, the search for the other input Y must shift its domain from the entire WS to the match set for X. What is the match set? In short, it consists of less embedded SOs in the WS whose heads match the head of X. More precisely, it is defined as follows.

(71) Y is in *the match set* for X iff
a. the head of Y matches the head of X, and
b. there is no SO such that SO has the same head as Y and dominates Y.

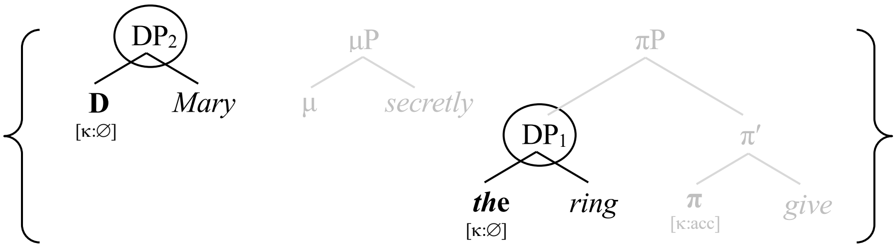
---

[14] We assume that all nominal expressions, including pronouns (e.g., *she*) and names (e.g., *Mary*), are syntactically complex and must take at least the form of $[_{DP}$ D N], where D can be null. On arguments for the syntactic complexity of pronouns and names, see Progovac (1998) and Matushansky (2008).

Note that (71b) serves as a "top-down" algorithm that picks up the least embedded candidate. Thus, if a head H projects a SO like [$_{\text{HP}}$ … [$_{\text{H}'}$ H …] and matches the head of X, then (71b) selects the highest projection HP as a member of the match set for X, and excluds any lower projections of H.

Let us now show how the MSC works. Imagine that Merge has taken πP in (69) as an input. Then, the MSC restricts the other input to this Merge application and requires it to come from the match set for πP, because the head $\pi_{[\kappa:\,\text{acc}]}$ of πP holds Match inside πP, namely with the head *the*$_{[\kappa:\,\varnothing]}$ of DP_1, Note that the match set for πP consists of two SOs, namely DP_1 and DP_2, since the head $\pi_{[\kappa:\,\text{acc}]}$ matches their heads *the*$_{[\kappa:\,\varnothing]}$ and $\text{D}_{[\kappa:\,\varnothing]}$. Given this, the match set for πP can be represented as follows.

(72) Match Set for πP =



The gray parts are actually not included in the match set. This means that the match set is a subset of the term SOs of the WS, and may also have a non-member of the WS as a member. Thus, though the input search for Merge is essentially member search, it can see a non-member of the WS by shifting its domain to the match set for some SO. In short, the MSC can make the mode of IM potentially available, and it counts as a "good" solution for the minimal search principle, preserving the potential of Merge.

It is then important to note that the MSC only limits the input search within the WS, and not within the Lex. This is because the MSC is a condition on working memory, and the Lex is a kind of long-term memory. In other words, the input search for Merge can get free access to the Lex. Thus, our theory offers two algorithms on the input search for Merge: [A] if the head of one input X holds Match inside X, the other input Y must be a member of the Lex or the match set; [B] if the head of X does not hold Match inside X, Y must be a member of the Lex or the WS. For our purpose, the more important algorithm is [A], which requires the head of Y to match the head of X unless Y is taken from the Lex.

3.3. Demonstration: Deriving the Anti-stacking Effect

We now demonstrate how the MSC derives the ASE. Recall that the ASE is an effect that excludes an AdvP at the edge of a κ-assigner head, i.e., [$_{\text{HP}}$ *AdvP [$_{\text{H}''}$ **DP**$_{[\kappa]}$ [$_{\text{H}'}$ $\text{H}_{[\kappa]}$ …]]]. Let us begin by considering the ASE pattern found in the single object VoiceP. The question is how the MSC prevents AdvP from occurring at the edge of *v*P, as shown below.

(73) John (✓*secretly*) **gave** (**secretly*) **the ring** (✓*secretly*) **to Mary** (✓*secretly*) yesterday.

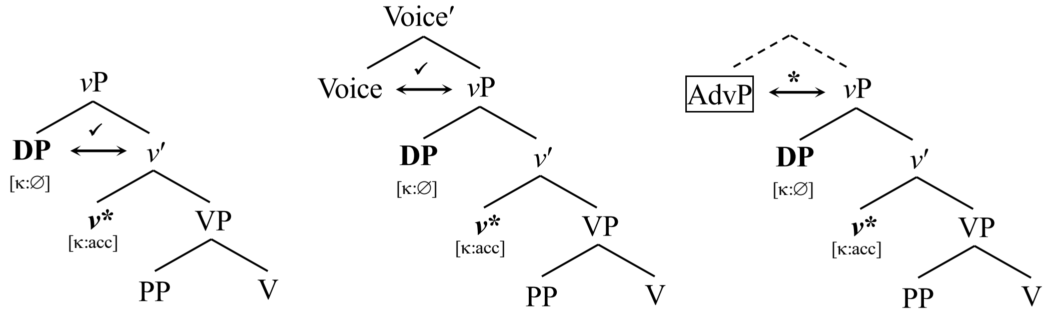
a. [$_{\text{VoiceP}}$ t_{John} [$_{\text{Voice}'}$ **Voice** [$_{v\text{P}}$ **DP**$_{[\kappa]}$ [$_{v'}$ $v^*_{[\kappa]}$ [$_{\text{VP}}$ **PP** [✓AdvP V]]]]]]
b. [$_{\text{VoiceP}}$ t_{John} [$_{\text{Voice}'}$ **Voice** [$_{v\text{P}}$ **DP**$_{[\kappa]}$ [$_{v'}$ $v^*_{[\kappa]}$ [✓AdvP [$_{\text{V}'}$ **PP** V]]]]]]
c. [$_{\text{VoiceP}}$ t_{John} [$_{\text{Voice}'}$ **Voice** [*AdvP [$_{v\text{P}}$ **DP**$_{[\kappa]}$ [$_{v'}$ $v^*_{[\kappa]}$ [$_{\text{VP}}$ **PP** V]]]]]]
d. [✓AdvP [$_{\text{Voice}''}$ t_{John} [$_{\text{Voice}'}$ **Voice** [$_{v\text{P}}$ **DP**$_{[\kappa]}$ [$_{v'}$ $v^*_{[\kappa]}$ [$_{\text{VP}}$ **PP** V]]]]]]

The leading idea is that the MSC disallows AdvP to merge to a SO if the head of the SO holds Match inside the SO; call this kind of SO a *matching center* (MC). Then, AdvP can merge to V, V′, and Voice″ in (73a, b, d), since they are not MCs; that is, the heads V and Voice lack κ-features and do not hold

Match inside their projections. The edge of *v*P, by contrast, is not available to AdvP as in (73c), since *v*P is a MC in the sense that the head *v** has a κ-feature and matches the head of DP inside *v*P.

To show how to rule out AdvP's position in (73c), let us examine the following derivation.

(74) a. Merge(DP, *v*′) b. Merge(Voice, *v*P) c. *Merge(AdvP, *v*P)

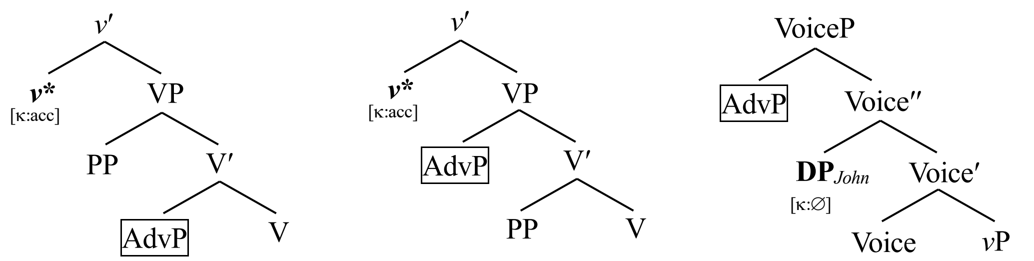


First of all, the structure *v*P in (74a) counts as a MC, since its head $v^*_{[\kappa:acc]}$ matches the head $D_{[\kappa:\varnothing]}$ of DP; they make a valued-unvalued pair in κ-features. Note that DP cannot have its κ-feature valued via Agree with the head *v* at this point. This is a consequence of the Agree-in-Transfer model, where Agree applies only at phase levels like VoiceP, and Match holds until that timing. Thus, the MSC applies to *v*P and the input to Merge with *v*P must be taken from the match set for *v*P or the Lex. This means that *v*P can only be merged with a matching SO, or a single LI such as Voice in (74b).

The question then is whether *v*P can merge with an AdvP. We argue that this is impossible for the following reasons. To begin with, recall that every AdvP is not a single LI but a SO that consists of a root and the modifier head μ, like {μ, *secretly*}, so the Lex cannot supply such a complex SO. Then, suppose that the WS includes an AdvP, which may be created before or after *v*P is created. Still, is the AdvP in the match set for *v*P? The answer is no, because the head of AdvP has no κ-feature and cannot match the head $v^*_{[\kappa:acc]}$. It therefore follows that *v*P cannot be merged with AdvP, as in (74c).

On the other hand, the V head, V′, and Voice″ can be merged with AdvP, as shown below.

(75) a. AdvP at the edge of V b. AdvP at the edge of V′ c. AdvP at the edge of Voice″



This is because V, V′, and Voice″ are not MCs (i.e., the heads V and Voice induce no Match). Thus, the MSC does not apply to them, and they can be merged with an AdvP whenever it exists in the WS.

We then turn to the ASE pattern in the double object VoiceP. Consider the following example.

(76) John (✓*secretly*) **gave** (**secretly*) **Mary** (**secretly*) **the ring** (✓*secretly*) yesterday.

a. $[_{VoiceP}$ … $[_{vP}$ $\mathbf{IO}_{[\kappa]}$ $[_{v'}$ $v_{[\kappa]}$ $[_{\pi P}$ $\boldsymbol{t}_{\mathbf{IO}}$ $[_{\pi''}$ $\mathbf{DO}_{[\kappa]}$ $[_{\pi'}$ $\pi_{[\kappa]}$ [✓AdvP V]]]]]]]

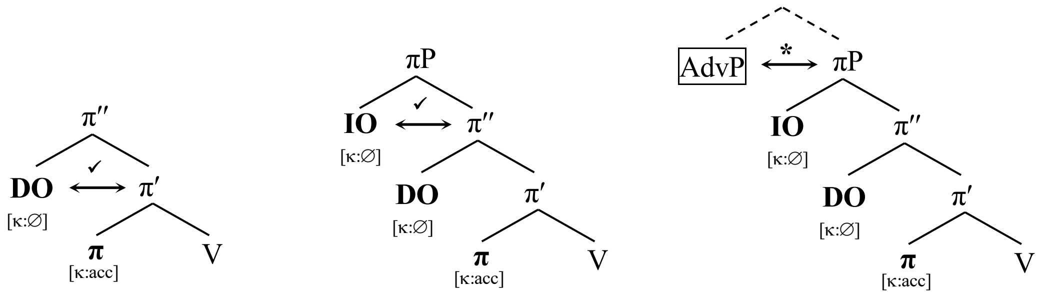
b. $[_{VoiceP}$ … $[_{vP}$ $\mathbf{IO}_{[\kappa]}$ $[_{v'}$ $v_{[\kappa]}$ [*AdvP $[_{\pi P}$ $\boldsymbol{t}_{\mathbf{IO}}$ $[_{\pi''}$ $\mathbf{DO}_{[\kappa]}$ $[_{\pi'}$ $\pi_{[\kappa]}$ V]]]]]]]

c. $[_{\text{VoiceP}} \ldots\ [\ \boxed{\text{*AdvP}}\ [_{vP}\ \mathbf{IO}_{[\kappa]}\ [_{v'}\ v_{[\kappa]}\ [_{\pi P}\ t_{\mathbf{IO}}\ [_{\pi''}\ \mathbf{DO}_{[\kappa]}\ [_{\pi'}\ \pi_{[\kappa]}\ \text{V}\ ]]]]]]]$

d. $[\ \boxed{{}^{\checkmark}\text{AdvP}}\ [_{\text{VoiceP}} \ldots\ [_{vP}\ \mathbf{IO}_{[\kappa]}\ [_{v'}\ v_{[\kappa]}\ [_{\pi P}\ t_{\mathbf{IO}}\ [_{\pi''}\ \mathbf{DO}_{[\kappa]}\ [_{\pi'}\ \pi_{[\kappa]}\ \text{V}\ ]]]]]]]$

The question here is why AdvP cannot be merged to πP or *v*P, as in (76b) and (76c). To answer this question step by step, let us examine the following derivation, which is creating the πP domain.

(77) a. Merge(DO, π′) b. Merge(IO, π″) c. *Merge(AdvP, πP)

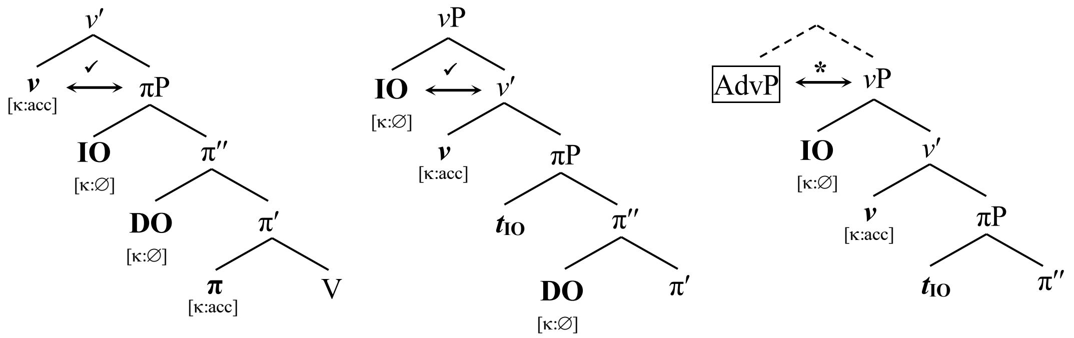


First of all, the structure π″ in (77a) counts as a MC, since its head $\pi_{[\kappa:\text{acc}]}$ matches the head $D_{[\kappa:\varnothing]}$ of the direct object DO. Thus, the MSC applies to π″ and the input to Merge with π″ must come from the Lex or the match set for π″, which includes at least DO. Note that π″ semantically requires the indirect object IO to merge to its edge, so suppose that IO is now created; i.e., the current WS consists of at least IO and π″, like {…, IO, π″, …}.[15] It is then clear that the match set for π″ must include IO, too, since the head $D_{[\kappa:\varnothing]}$ of IO matches the head $\pi_{[\kappa:\text{acc}]}$. This means that π″ can be merged with IO as in (77b).

Importantly, the resulting structure πP in (77b) also counts as a MC, because it contains SOs whose heads match the head $\pi_{[\kappa:\text{acc}]}$; they are IO and DO. Indeed, their heads $\pi_{[\kappa:\text{dat}]}$ and $D_{[\kappa:\varnothing]}$ match in that they make valued-unvalued pairs in κ-features. Thus, the MSC applies to πP, and the input to Merge with πP must be taken from the Lex or the match set for πP. It then follows that Merge cannot extend πP by selecting an AdvP as in (77c), since no AdvP can be taken from the Lex or the match set.

Let us now consider the following derivation, which is in turn creating the next domain *v*P.

(78) a. Merge(*v*, πP) b. Merge(IO, *v*′) c. *Merge(AdvP, *v*P)



To begin with, note that the structure *v*′ in (78a) is a MC, since it also contains IO and DO, whose heads

---

[15] More formally, we assume that our applicative head π is of type $\langle\langle \mathbf{v}, \mathbf{t}\rangle, \langle \mathbf{e}, \langle \mathbf{e}, \langle \mathbf{v}, \mathbf{t}\rangle\rangle\rangle\rangle$, given that the root V *give* is of type $\langle \mathbf{v}, \mathbf{t}\rangle$ (cf. Pylkkänen 2002, Bruening 2010, 2015, 2021). Thus, π′ is of type $\langle \mathbf{e}, \langle \mathbf{e}, \langle \mathbf{v}, \mathbf{t}\rangle\rangle\rangle$, and π″ is of type $\langle \mathbf{e}, \langle \mathbf{v}, \mathbf{t}\rangle\rangle$. It is now clear that neither π′ nor π″ is a potential target of modification by manner AdvPs, which we have analyzed as being of type $\langle\langle \mathbf{v}, \mathbf{t}\rangle, \langle \mathbf{v}, \mathbf{t}\rangle\rangle$.

match the head $v_{[\kappa:acc]}$. Thus, the MSC restricts the input to Merge with v' and it must come from the Lex or the match set for v', which consists of IO and DO. This means that v' and IO can be internally merged as in (78b), and this movement step results in *v*P, where $IO_{[\kappa:\varnothing]}$ c-commands the head $v_{[\kappa:acc]}$ and they can undergo Agree at a phase level.[16] What is important here is that *v*P also counts as a MC, since it contains the two matching SOs, IO and DO. It then follows that the MSC also applies to *v*P, and this restriction blocks a non-matching SO like AdvP from merging with *v*P, as shown in (78c).

In summary, we have shown how the MSC derives the ASE. The key idea is that no AdvP may be taken as an input to Merge if the head of the other input SO holds Match inside the SO. In this sense, our account of the ASE is derivational, ruling out the merge of AdvP *at the level of input*, and not *at the level of output*. The MSC is therefore a restriction compatible with compositional semantics, unlike Koizumi's (1995) Agr-based output filter, because it makes *v*P and πP only derivationally unavailable to the merge of AdvP. In short, the MSC is a proposal motivated both empirically and theoretically, which also solves the tension between the principle of minimal search and the potential of the operation Merge.

4. Verification

4.1. The Anti-stacking Effect at the Finite Left Periphery

Under our Match-based approach to the ASE, it should follow that no other κ-position than *v*P and πP can host AdvP, either. This prediction can be verified by looking at the left periphery of finite clauses, because there we find the head I, which assigns nominative κ to the DP at its edge (e.g., Chomsky 1981). In other words, we are now making the claim that no AdvP occurs at the edge of IP, as shown below.

(79) **The distribution of AdvPs around the finite subject**

[……… $[_{IP}$ *AdvP $[_{I''}$ **DP**$_{[\kappa:\varnothing]}$ $[_{I'}$ $I_{[\kappa:nom]}$ $[_{VoiceP}$ … t_{DP} …]]]]]

This ASE pattern is also what the MSC predicts, since the structure I″ counts as a MC in the sense that its head $I_{[\kappa:nom]}$ holds κ-feature matching with the head of the subject $DP_{[\kappa:\varnothing]}$. That is, the MSC applies to I″ and it cannot be merged with any non-matching SO such as AdvP. This is exactly the same kind of explanation as we have given to the restrictions on AdvP placement within *v*P and πP.

Still, the ASE pattern above does not appear to hold at first glance. For example, there are several classes of AdvPs that may appear between the complementizer *that* and the subject DP, such as *frequency AdvP* (e.g., *often*), *modal AdvP* (e.g., *probably*), *evaluative AdvP* (e.g., *foolishly*), and *evidential AdvP* (e.g., *apparently*). The partial distribution of these AdvPs at the left periphery is illustrated below.

(80) a. I know **that you** are *often* angry at your boyfriend.
b. I know **that** *often* **you** are angry at your boyfriend.

(81) a. I think **that he** will *probably* be able to solve the problem.
b. I think **that** *probably* **he** will be able to solve the problem.

(82) a. I know **that the student** has *foolishly* been skipping your class.
b. I know **that** *foolishly* **the student** has been skipping your class.

[16] In (78a), since DO is in the match set for v', it may internally merge to v' instead of IO, although this option must be blocked. Here, we adapt Chomsky's (2000, 2001) locality condition, assuming that the actual input to Merge is confined to members of the match set that are not c-commanded by any other members.

(83) a. I think **that John** is *apparently* in love with that woman.
b. I think **that** *apparently* **John** is in love with that woman.

In the following, however, we argue that the (b) examples are not true counterexamples to the MSC, and that the AdvPs can be analyzed as not occurring at the edge of IP. Specifically, we suggest that the AdvPs in the (b) examples occupy the edge position of an additional functional category, which exists between the complementizer *that* and the subject DP.

We begin by adopting Rizzi's (1997) *Split-CP Hypothesis*, decomposing the category of complementizer C into at least two heads. One is *finiteness* (Fin), which may take IP as its sister and expresses a rudimentary distinction for tense, mood, polarity, and agreement. The other is *clause typing* (Typ), which may take FinP and specify the illocutionary force of the IP as indicative, imperative, or interrogative. Under this architecture, therefore, IP may be dominated by two functional categories: FinP and TypP.

(84) **The fine structure of the finite left periphery**
$[_{\text{TypP}}$ Typ $[_{\text{FinP}}$ Fin $[_{\text{IP}}$ $\mathbf{DP}_{[\text{K}:\varnothing]}$ $[_{\text{I}'}$ $\text{I}_{[\text{K:nom}]}$ $[_{\text{VoiceP}} \ldots t_{\text{DP}} \ldots]]]]]$

Let us then adapt Rizzi's (1997: 312) suggestion that Typ and Fin may occur as a single head in English. Here, suppose that the Typ head may function as what Bhatt and Yoon (1991) call a *subordinator* and be realized as a complementizer like *that*. Given this, we propose the following structural distinction between root and non-root clauses; in root clauses, the two heads are fused as a single one, which we call *Force* (cf. Rizzi 1997), while in subordinate clauses, they are split, as shown below.[17]

(85) a. Root finite clauses: $[_{\text{ForceP}}$ Force (= Typ/Fin) $[_{\text{IP}}$ DP $[_{\text{I}'}$ I VoiceP $]]]$
b. Subordinate finite clauses: $[_{\text{TypP}}$ Typ $[_{\text{FinP}}$ Fin $[_{\text{IP}}$ DP $[_{\text{I}'}$ I VoiceP $]]]]$

This distinction allows us to make two uniform claims. The first one is that FinP always serves as the locus of *auxiliary inversion*; in root clauses, the inverted auxiliary (Aux) occupies the fused Typ/Fin, but in embedded clauses, it occupies the split Fin. The second claim is that TypP always serves as the locus of *wh-movement*; in root clauses, it targets the edge of the fused Typ/Fin, but in embedded clauses, it targets the edge of the split Typ. In the following, we lend empirical support to these two claims.

Our first concern is the syntax of FinP, and we suggest that whenever a finite Aux is inverted, it is located in the Fin head, contra Rizzi (1997: 303ff.). This idea is supported by Hopper and Thompson's (1973) observations on *negative inversion* (NI). NI is an Aux-inversion rule which preposes a negative phrase (*neg*-XP) to the left periphery, as Hopper and Thompson (H&T) show.

(86) a. I **have** *never in my life* seen such a crowd.
b. *Never in my life* **have** I seen such a crowd. (H&T 1973: 467)

What is then important is that NI may also apply in embedded finite clauses, as shown below.

---

[17] We assume that the bundling of the heads Typ and Fin is parametrized; it may vary by language where they are fused or split (cf. Bhatt & Yoon 1991). Thus, it is a theoretical possibility to obtain a language or dialect whose root clauses have the heads Typ and Fin split. See also footnote 19.

(87) a. I exclaimed **that** *never in my life* **had** I seen such a crowd.
b. It's true **that** *never in his life* **has** he had to borrow money. (H&T 1973: 474, 476)

Here, the inverted Aux occurs with the complementizer *that*, which clearly indicates that they occupy distinct positions. Given that Typ may serve as a subordinator that closes off the embedded clause, we assume that it hosts *that*, while Fin hosts the inverted Aux. Specifically, we suggest that the inverted Aux reaches the position of Fin by Aux-to-Fin raising, with the *neg*-XP placed at the edge of FinP.

(88) **The structure of embedded negative inversion**
$[_{\text{TypP}}$ **that** $[_{\text{FinP}}$ *neg*-XP $[_{\text{Fin}'}$ Fin-**Aux** $[_{\text{IP}}$ DP …]]]]

At this point, one might wonder whether FinP is an appropriate label for the category that applies NI. In fact, however, our analysis of NI is empirically motivated, because the applicability of NI correlates with the presence of finiteness.[18] This point can be made clear by Culicover's (2013) observations, according to which NI cannot apply within *to*-infinitival clauses, as shown below.

(89) a. I expected **that** *not once* **would** I see John.
b. * I expected *not once* **to** have seen John. (Culicover 2013: 225)

This restriction follows if we assume that FinP is (i) the locus of NI, and (ii) occurs only in finite clauses. Still, the latter point does not entail that TypP is also limited to finite clauses, since the Typ head may be a subordinator, which we assume is further divided into two sorts; one is for finite clauses, and the other is for *to*-infinitives. Thus, we may have at least two types of subordinate structures.

(90) a. Subordinate finite clauses: $[_{\text{TypP}}$ Typ $[_{\text{FinP}}$ Fin $[_{\text{IP}}$ …]]]
b. Subordinate *to*-infinitives: $[_{\text{TypP}}$ Typ $[_{to\text{P}}$ *to* $[_{v\text{P}}$ …]]]

Importantly, postulating the Typ head in *to*-infinitives makes one prediction, once we assume that TypP is the locus of *wh*-movement. That is, if *wh*-movement targets the edge of TypP, then it should hold that *to*-infinitives can apply *wh*-movement, just as finite clauses can. This prediction is borne out as shown below, suggesting that NI and *wh*-movement target different positions.

(91) a. I was wondering *what* I **should** do.
b. I was wondering *what* **to** do. (Culicover 2013: 225)

Let us then consider our second claim in more detail, namely that *wh*-movement (WM) targets the edge of TypP. Given that the root clause bundles the two heads Typ and Fin into Force (= Typ/Fin), our clam entails that WM and NI in the root clause both target the edge of the bundled head Force. Thus, we

[18] Also, FinP has been given different names in the literature, such as *Polarity Phrase* (PolP) (e.g., Culicover 1991), and PolP and FinP seem to be notational variants. For instance, Culicover argues that PolP is a locus of NI (Culicover 1991: 49ff.), while Rizzi notes that FinP is the category that may also express negation in such languages as Latin and Celtic (Rizzi 1997: 284). This means that PolP and FinP have an aspect in common; they are both a locus of polarity. Given this, it is natural to hold a unified perspective on them.

assume that the *wh*-XP and *neg*-XP in (92) both occur at the edge of Force, as shown in (93).

(92) a. *Who* **will** Robin see and Lee talk to?
b. *No one* **will** Robin see and Lee talk to. (Culicover 2013: 241)

(93) a. [$_{\text{ForceP}}$ *who* [$_{\text{Force}'\ (=\ \text{Typ/Fin}')}$ **will** [$_{\text{IP}}$ Robin …]]]
b. [$_{\text{ForceP}}$ *no one* [$_{\text{Force}'\ (=\ \text{Typ/Fin}')}$ **will** [$_{\text{IP}}$ Robin …]]]

In embedded clauses, by contrast, WM and NI can and must target different positions, namely the edge of Typ and Fin, because the two heads are split. For example, let us assume that the *wh*-XP and *neg*-XP in (94) occur at the edge positions of Typ and Fin, respectively, as shown in (95).

(94) a. I wonder *what* Robin saw.
b. I said **that** *not once* **had** Robin raised his hand. (Culicover 2013: 224)

(95) a. [$_{\text{TypP}}$ *what* [$_{\text{Typ}'}$ Typ [$_{\text{FinP}}$ Fin [$_{\text{IP}}$ Robin …]]]]]
b. [$_{\text{TypP}}$ **that** [$_{\text{FinP}}$ *not once* [$_{\text{Fin}'}$ **had** [$_{\text{IP}}$ Robin …]]]]]

Note that this analysis is consistent with our first claim that FinP is the locus of Aux-inversion. One possibility is that Aux-inversion occurs at least when the Fin head is the locus of WM or NI. If so, it follows that the embedded WM cannot result in Aux-inversion, as shown in (96b), because it targets the edge of the Typ head, and it is not applied by the Fin head itself, as shown in (97b).

(96) a. *What* **did** Robin see?
b. * I wonder *what* **did** Robin see. (Culicover 2013: 224)

(97) a. [$_{\text{ForceP}}$ *what* [$_{\text{Force}'\ (=\ \text{Typ/Fin}')}$ **did** [$_{\text{IP}}$ Robin …]]]
b. [$_{\text{TypP}}$ *what* [$_{\text{Typ}'}$ Typ [$_{\text{FinP}}$ **did** [$_{\text{IP}}$ Robin …]]]]

These properties of FinP and TypP make two further predictions, which are upheld by Haegeman's (2000a, b) data. First, the embedded clause should allow WM and NI to apply together, but force *wh*-XP to occur higher than *neg*-XP, since the target of WM (TypP) is higher than that of NI (FinP). The following contrast shows that this is correct; *wh*-XP can occur with *neg*-XP, but must precede it.

(98) a. Lee wonders ***why*** *in no way* **would** Robin volunteer.
b. * Lee wonders *in no way* ***why*** **would** Robin volunteer. (Haegeman 2000b: 28)

Second, it is also predicted that the root clause does not allow *wh*-XP and *neg*-XP to co-occur in any order, since they compete for the same edge position of the bundled head Force, which may attract either *wh*-XP or *neg*-XP. This prediction is also upheld by the following paradigm.[19]

---

[19] Still, there is speaker variation in the root-subordinate asymmetry like (98a) vs. (99a), as Maekawa (2006), Duffield (2015), and Radford (2018) note; some speakers allow *wh*-XP and *neg*-XP to co-occur in root clauses, too. A full account of this variation is beyond the scope of this paper, but a possible direction for the analysis is to assume that the speakers who finds no such asymmetry can also split the bundled head Typ/Fin in the root clause when they apply NI. See also footnote 17.

(99) a. * ***Why***, *in no way* **would** Robin volunteer?
b. * *In no way*, ***why*** **would** Robin volunteer? (Haegeman 2000a: 134)

In this way, our theory of the left periphery is empirically motivated. We therefore assume three points; (i) the root clause in English bundles Typ and Fin as Force, while they are split in the embedded clause; (ii) TypP is the locus of WM; and (iii) FinP is the locus of Aux-inversion, including NI. Particularly important for us is (iii), under which the inverted Aux is used to mark the left boundary of IP.

Given this last point, we are now able to show that the ASE also holds at the finite left periphery. To begin with, let us again consider the distribution of the AdvP *often*, which can occur between the complementizer and the subject DP. The relevant examples in (80) are repeated here.

(80) a. I know **that you** are *often* angry at your boyfriend.
b. I know **that** *often* **you** are angry at your boyfriend.

In order to capture the acceptability of the (b) example, we assume that FinP is an interpretable category related to tense, mood, or polarity, so it can host some types of AdvPs at its edge, as shown in (100a). The question then is whether the AdvP *often* can occur at the edge of IP, as shown in (100b).

(100) a. [$_{\text{TypP}}$ **that** [$_{\text{FinP}}$ $^{\checkmark}$*often* [$_{\text{Fin}'}$ **Fin** [$_{\text{IP}}$ **you** [$_{\text{I}'}$ I …]]]]]
b. [$_{\text{TypP}}$ **that** [$_{\text{FinP}}$ **Fin** [$_{\text{IP}}$ $^{*/\checkmark}$*often* [$_{\text{I}''}$ **you** [$_{\text{I}'}$ I …]]]]]

We argue that the (b) configuration is impossible. This can be shown by the distribution of the AdvP in embedded NI clauses. As the following examples illustrate, it cannot appear between the subject DP and the inverted Aux, which is located in Fin and marks the left boundary of IP.

(101) a. The fact is **that** not only **is this theory** *often* criticized, but it has been proven to be false.
b. * The fact is **that** not only **is** *often* **this theory** criticized, but it has been proven to be false.

The reality of this ASE pattern can be further demonstrated if we consider the distribution of AdvPs in root interrogative clauses, which also have Aux-inversion. The point is that each type of AdvP fails to intervene between the subject DP and the inverted Aux, as shown by the following paradigms.

(102) a. Why **are you** *often* angry at your boyfriend?
b. * Why **are** *often* **you** angry at your boyfriend?

(103) a. Which problem **will he** *probably* not be able to solve?
b. * Which problem **will** *probably* **he** not be able to solve?

(104) a. Whose class **has the student** *foolishly* been skipping?
b. * Whose class **has** *foolishly* **the student** been skipping?

(105) a. Which woman **is John** *apparently* in love with?
b. * Which woman **is** *apparently* **John** in love with?

For similar observations, see Postal (1974: 136), Johnson (1991: 579), Koizumi (1995: 28), Rizzi (1997:

303), Potsdam (1998: 331), and Ernst (2002: 431), although they suggest different analyses of the unacceptable (b) cases. In our view, their deviance indicates that no AdvP occurs at the edge of IP, given that the inverted Aux in the root clause is in the fused head Force (= Typ/Fin) and marks the left boundary of IP. This result is exactly what the MSC entails, making an additional argument in its favor.

4.2. The Anti-stacking Effect at the Infinitival Left Periphery

Let us then consider *to*-infinitival clauses introduced by the preposition *for*. In the literature, it has been observed that the presence of *for* requires the subject DP of infinitives to be overt, and that no AdvP may occur between *for* and the subject DP (e.g., Johnson 1991, Koizumi 1995, Rizzi 1997, Ernst 2002). For example, this is shown by the following distribution of *foolishly* at the infinitival left periphery.

(106) a. It is quite possible **for** [**the student**] to *foolishly* skip the next class.
b. It is quite possible **for** [**the student**] *foolishly* to skip the next class.
c. * It is quite possible **for** *foolishly* [**the student**] to skip the next class.

In the rest of this section, we will argue that this ordering restriction is also a case of the ASE and can be deduced by the MSC. Needless to say, the success in our attempt depends on a particular proposal on the structure of the *for* infinitival construction. Given this, let us begin by examining the nature of *for*, and then propose its syntax behavior in line with Pesetsky and Torrego (2001, 2004).

First of all, we assume the preposition *for* that embeds infinitives as a kind of complementizer. This view has been widely accepted since Bresnan (1972), and indeed *for* shares some properties with the complementizer *that* (e.g., Pesetsky and Torrego 2001, McFadden 2012). First, *for* and *that* can be optionally added or omitted for some matrix predicates (though their presence is preferred for others).

(107) a. I would like (**for**) him to buy the book.
b. I believe (**that**) he bought the book. (McFadden 2012: 141)

Second, *for* and *that* must be added when the clauses they introduce appear as the matrix subject.

(108) a. [*(**For**) him to buy the book] was unexpected.
b. [*(**That**) he bought the book] was unexpected. (McFadden 2012: 141)

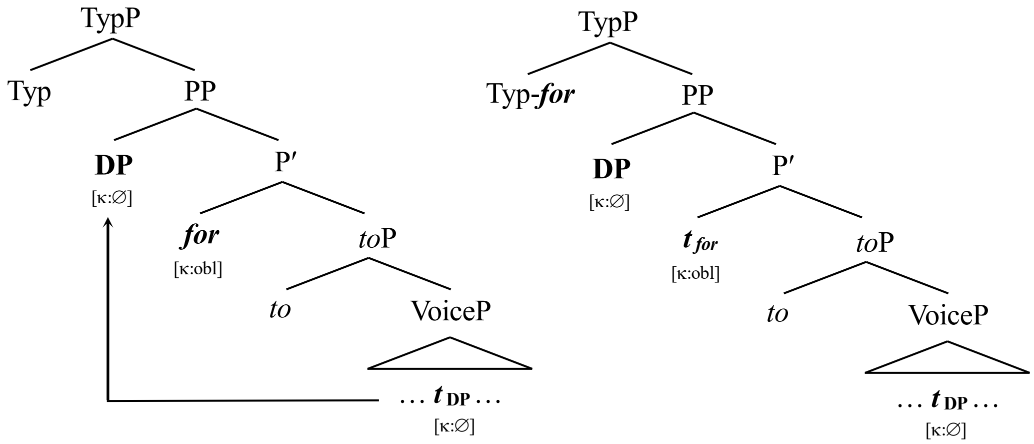
Third, *for* shows a sort of the *that*-trace effect, and it must be omitted if subject extraction occurs.

(109) a. *Who* would you like (***for**) __ to buy the book?
b. *Who* do you think (***that**) __ bought the book? (McFadden 2012: 141)

Given this parallel distribution of *for* and *that*, we take them as elements of the same class, namely, complementizers. Of course, *for* differs from *that* in disallowing AdvP to occur before the subject DP (e.g., *it is quite possible for* (**foolishly*) *the student to skip the next class*. vs. *I know that* (*foolishly*) *the student has been skipping your class*.), but this difference is exactly what we aim to explain.

We now propose a structure for the *for* infinitival construction. At the core of our claim is Pestesky and Torrego's (2001, 2004) view that *for* itself is not the complementizer head (Typ) but rather a preposition P that raises to form a complex head with Typ. While they identify the category P as a species of the category *Tense*, it suffices for us to represent *for* as a P head. Our point is that the complementizer use of *for* requires it to take a *to*-infinitive as its sister and undergo overt raising to Typ, as shown below.

(110) a. Movement to the edge of *for*-PP b. P-raising to the subordinator head Typ



Importantly, we assume with Chomsky (1981) that *for* acts as an oblique κ-assigner; in our terms, it has a valued κ-feature [κ: obl]. On Bošković's (2007) κ-valuation system, this means that the subject $DP_{[\kappa:\varnothing]}$ must move to the edge of PP so it can undergo Agree by c-commanding a matching goal $for_{[\kappa:obl]}$. While this movement step does not result in the actual word order, it is retrieved by P-raising. Thus, we correctly predict that present-day U.S. English excludes any configurations like (111a) and (111c).

(111) a. * It is quite possible **for** *to* [**the student**] skip the next class.
b. It is quite possible **for** [**the student**] *to* skip the next class.
c. * It is quite possible [**the student**] **for** *to* skip the next class.

Note that this analysis predicts two further facts. First, the subject DP on the right of *for* should also exhibit freezing effects, because it occurs there by movement, unlike the oblique DP in the prepositional use of *for*. This is a correct prediction, and the subject DP does not allow extraction out of it.

(112) a. Who_i are you looking **for [a picture of** t_i **]**?
b. * Who_i would you like **for [a picture of** t_i **]** to win the award?

Second, it should hold that *for* and the subject DP never move together, because they do not form a constituent at any derivational stage. As observed by Bresnan (1972), this prediction is also born out.

(113) a. [*Which woman*]$_i$ would you like (***for**) t_i to accompany you?
b. [(***For**) *which woman*]$_i$ would you like t_i to accompany you?

Thus, we should avoid assuming that *for* and the subject DP form a PP, where the DP is base-generated as the sister of *for*. The problem with such an analysis is that it has difficulty distinguishing the complementizer use of *for* from its preposition use, in which *for* can be moved together with its oblique DP.

(114) a. [*Which woman*]$_i$ did you buy this ring **for** t_i?
b. [**For** *which woman*]$_i$ did you buy this ring t_i?

Given these results, suppose that our analysis of the *for* infinitival construction is on the right track, where the raised P head *for* marks the left boundary of the κ-valuation domain for the subject DP. What is then important is that it allows us to regard the ordering restriction on AdvPs at the infinitival left periphery as a case of the ASE. For example, consider the distribution of *foolishly* again.

(115) It is quite possible **for** (**foolishly*) **the student** (✓*foolishly*) to (✓*foolishly*) skip the next class.

a. [$_{\text{TypP}}$ Typ-***for***$_{i}$ [$_{\text{PP}}$ **DP**$_{[\kappa]}$ [$_{\text{P}'}$ ***t***$_{i\,[\kappa]}$ [$_{to\text{P}}$ *to* [✓AdvP [$_{\text{VoiceP}}$ … V …]]]]]]

b. [$_{\text{TypP}}$ Typ-***for***$_{i}$ [$_{\text{PP}}$ **DP**$_{[\kappa]}$ [$_{\text{P}'}$ ***t***$_{i\,[\kappa]}$ [✓AdvP [$_{to\text{P}}$ *to* [$_{\text{VoiceP}}$ … V …]]]]]]

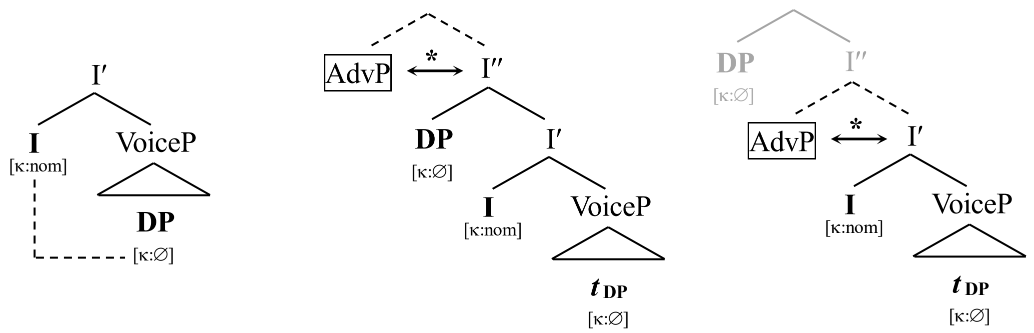
c. [$_{\text{TypP}}$ Typ-***for***$_{i}$ [*AdvP [$_{\text{PP}}$ **DP**$_{[\kappa]}$ [$_{\text{P}'}$ ***t***$_{i\,[\kappa]}$ [$_{to\text{P}}$ *to* [$_{\text{VoiceP}}$ … V …]]]]]]

As we can see in (115c), the AdvP cannot be merged at the edge of a κ-assigner head, here the P head *for*. Since this restriction is clearly a case of the ASE, we can derive it from the MSC as follows. That is, the structure PP counts as a MC, because its head P matches the head of the subject DP in κ-features, so the MSC applies to PP and forces it not to be merged with any non-matching SO such as AdvP.

4.3. Crosslinguistic Variation in the Anti-stacking Effect

Up to this point, we have shown that the ASE can be reduced to the MSC, which blocks AdvP from occurring right above a κ-position. As the reader may have noticed, however, this is not all that the MSC entails. In fact, it further predicts that no AdvP occurs right below a raised DP. To make this point clearer, let us examine the following derivation, which is creating a finite IP.

(116) a. Merge(I, VoiceP) b. *Merge(AdvP, I″) c. *Merge(AdvP, I′)



In (116a), the structure I′ counts as a MC, since it contains a SO whose head holds κ-feature matching with the head I, namely the subject DP within VoiceP. Thus, the MSC restricts the input search for Merge with I′, and I′ can only be merged with a LI, or a matching SO like the subject DP as in (116b). Crucially, this entails that I′ cannot be merged with any non-matching SOs, like AdvP as in (116c).

The MSC therefore predicts that no AdvP occurs between the raised DP and the I head. At first sight, this does not appear to hold, especially under the assumption that the I head is the position for finite Aux verbs such as copula *be*, aspect *have*, and modal *will*. For instance, the AdvPs *often*, *probably*, *foolishly*, and *apparently* can all appear between the subject DP and the finite Aux, as shown below.

(117) a. I know that **you** *often* **are** angry at your boyfriend.

b. I think that **he** *probably* **will** be able to solve the problem.

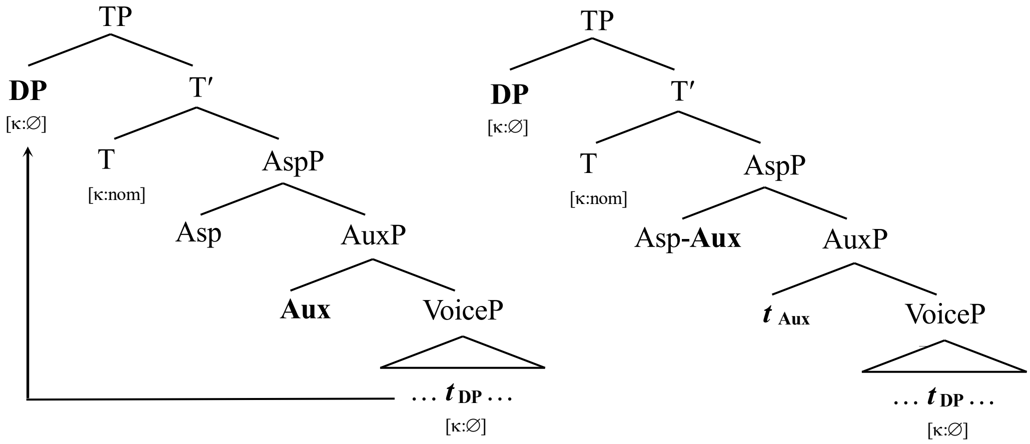
c. I know that **the student** *foolishly* **has** been skipping your class.

d. I think that **John** *apparently* **is** in love with that woman.

Then, does the acceptability of these examples indicate that the MSC makes a wrong prediction? Our answer is no, and the MSC is right as it is. What we will argue against instead is the standard view on the position of the finite Aux. In particular, we will suggest that, in English, the head which attracts the subject DP should be distinguished from the one which hosts the finite Aux.

First of all, built on Pollock's (1989) *Split-IP Hypothesis*, we assume with Kratzer (1998) and Hacquard (2006, 2009, 2010) that the category of I breaks down into at least two heads. One is *Aspect* (Asp), which takes as its sister a VoiceP or an AuxP containing it, and projects an AspP that denotes a predicate of times. The other head is *Tense* (T), which takes the AspP as its sister and denotes a reference time as the semantic argument for the AspP. With this architecture, we suggest that while T is the locus of nominative κ-valuation, Asp serves to host the finite Aux in English, as shown below.

(118) a. Movement to the edge of TP b. Aux-raising to Asp (when Asp is null)



Under this proposal, the MSC does not necessarily rule out the presence of an AdvP between the subject DP and the finite Aux, since it is still possible to analyze the AdvP as occurring at the edge of AspP, and not between the κ-matching elements, here the raised DP and the T head. Then, the actual prediction that the MSC makes is that no AdvP occurs above the finite Aux in a language where Aux-raising targets the T head. In the following, we make a crosslinguistic argument for this prediction, after we have elucidated how plausible it is to assume that the Asp head hosts the finite Aux in English.

As Kratzer (1998: 107) notes, the category Asp posited above should not be confused with *viewpoint aspect* such as expressed by the auxiliary *have*, which we take to be an instance of Aux verbs. Rather the Asp head falls into four types (i.e., $\mathrm{Asp}_{\mathrm{perfect}}$, $\mathrm{Asp}_{\mathrm{imperfect}}$, $\mathrm{Asp}_{\mathrm{perfective}}$, $\mathrm{Asp}_{\mathrm{imperfective}}$), and each type acts as a "tense" operator that takes the running time t_E of the event described by VoiceP and relates t_E to the reference time t_R denoted by the T head, which falls into two types (i.e., T_{present}, T_{past}). The temporal relations that the Asp heads each encode are informally characterized as follows.

(119) a. $\mathrm{Asp}_{\mathrm{perfect}}$: requires t_E to precede t_R. (e.g., If t_R is "now", t_E is set before "now")
b. $\mathrm{Asp}_{\mathrm{imperfect}}$: requires t_R to precede t_E. (e.g., If t_R is "now", t_E is set after "now")
c. $\mathrm{Asp}_{\mathrm{perfective}}$: requires t_E to be a part of t_R. (e.g., If t_R is "now", t_E has ended "now")
d. $\mathrm{Asp}_{\mathrm{imperfective}}$: requires t_R to be a part of t_E. (e.g., If t_R is "now", t_E is ongoing "now")

This means that the semantic values of Asp and T jointly determine the tense morphology and its interpretation. For example, Kratzer points out that the simple past morphology in English and German have different meanings. Given that you are looking at churches in Italy, consider the following discourses.

(120) a. Who **built** this church? Borromini **built** this church.
b. * *Wer* ***baute*** *diese Kirche*? *Borrromini* ***baute*** *diese Kirche*.
who built this church Borromini built this church
c. *Wer* ***hat*** *diese Kirche* ***gebaut***? *Borrromini* ***hat*** *diese Kirche* ***gebaut***.
who has this church built Borromini has this church built (Kratzer 1998: 106)

The English simple past tense in (120a) is accepted out of the blue, but the German counterpart in (120b) is not, and instead the present perfect tense must be used as in (120c). Thus, the English simple past may have a meaning that the German present perfect does, and this fact suggests that the simple past form in English can be used to spell out certain combinations of T and Asp; in (120a), $T_{present}$ + $Asp_{perfect}$, which is realized as the past morphology on the verb in English. In this sense, Asp is a functional head within the tense system, and our claim is that it is a head that may realize the finite Aux in English.

Note that the head $Asp_{imperfect}$ in (119b) is a “future” operator, so we assume that $Asp_{imperfect}$ may be realized as some modal verbs, while treating others as Aux verbs. This is a plausible assumption, because Roberts (1998) observes that some modal verbs (e.g., *might*, *must*) needs to be interpreted over the scope of negation *not*, while others (e.g., *can*, *need*) must be interpreted under it, as shown below.

(121) a. There *might***n't** be a unicorn in the garden. (*possible* > **not**)
b. There *can***'t** be a unicorn in the garden. (**not** > *possible*) (Roberts 1998: 115)

(122) a. You *must***n't** do that. (*obligation* > **not**)
b. You *need***n't** do that. (**not** > *obligation*) (Roberts 1998: 115)

These contrasts suggest that *can* and *need* are base-generated below *not* and undergo overt raising over it, and that they are interpreted in their base positions by reconstruction. In other words, if we assume that the base position of *not* is located below AspP, then it is an analytical possibility to identify, not T, but Asp, with the position that hosts the finite Aux through base-generation or head raising.

This claim has a crosslinguistic basis, which comes from the comparison of V-raising in English and French. Since Emonds (1978), it has been established that V-raising applies to both lexical and auxiliary Vs in French, but only the latter in English. Pollock (1989) then suggests that V-raising in both languages uniformly targets the T head. This claim is upheld by the distribution of manner AdvPs, given that they can only occur within VoiceP. Consider the following examples from Ernst (2002).

(123) a. * **Claude** <u>softly</u> **has** called his cat.
b. * ***Claude*** <u>*doucement*</u> ***a*** *appelé son chat*.
Claude softly has called his cat (Ernst 2002: 393)

However, there is a potential problem with this reasoning, as noted by Kayne (1989), Belletti (1990), and Ernst (2002). For instance, modal AdvPs can be put between the subject DP and the finite Aux in English, but this is impossible in French, as shown below.

(124) a. **Claude** <u>probably</u> **has** called his cat.
b. * ***Claude*** <u>*probablement*</u> ***a*** *appelé son chat*.
Claude probably has called his cat (Ernst 2002: 393)

Under Pollock's view, the contrast between (124a) and (124b) is mysterious, since the finite Aux in both languages occupies the same head T. Note that this sort of contrast is not limited to modal AdvPs, but it holds for other AdvPs. Consider the following data adapted from Ernst (2002: 386–387).

(125)	a.	(✓*Souvent*,)	***Jean-Pierre***	(**souvent*)	***a***	(✓*souvent*)	*parlé*	*à Marie*.
		often	Jean-Pierre	often	has	often	spoken	to Marie
	b.	(✓*Certainement*,)	***Jean-Pierre***	(**certainement*)	***a***	(✓*certainement*)	*parlé*	*à Marie*.
		certainly	Jean-Pierre	certainly	has	certainly	spoken	to Marie
	c.	(✓*Stupidement*,)	***Jean-Pierre***	(**stupidement*)	***a***	(✓*stupidement*)	*parlé*	*à Marie*.
		stupidly	Jean-Pierre	stupidly	has	stupidly	spoken	to Marie

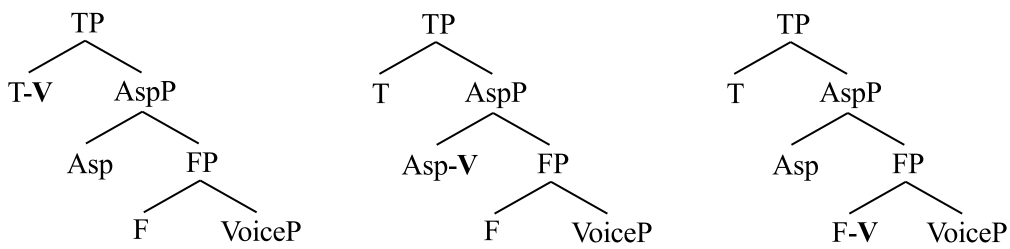
Thus, if we still assume that the difference between English and French in the distribution of AdvPs reflects that in the height of V-raising, and that V-raising in French targets the T head, then these facts lend the initial support to our claim. That is, the finite Aux in English should occupy a different position than T, as alluded to by Kayne (1989), Koizumi (1995), and Pollock (1997).

Our direction for the analysis seems to be right, and the variation across languages in the distribution of AdvPs should be partly attributed to the varying height of V-raising. This is indeed what Schifano (2018) and Roberts (2019) suggest in conducting a large-scale crosslinguistic survey. For instance, they demonstrate that the distribution of modal AdvPs varies across Romance languages. This is shown in the following paradigm, which is adapted from Roberts (2019: 349).

(126)	a.	***Antoine***	(**probablement*)	***confound***	(✓*probablement*)	*le poème*.	[French]
		A.	probably	confuses	probably	the poem	
	b.	***Andrei***	(✓*probabil*)	***greşeşte***	(✓*probabil*).		[Romanian]
		A.	probably	is-wrong	probably		
	c.	***Sergio***	(✓*probablemente*)	***confunde***	(**probablemente*)	*este poema*.	[Spanish]
		S.	probably	confuses	probably	this poem	

This variation can be captured if the height of the lexical V differs by language. For an illustration, let us assume that it differs as follows, given that there is a functional head F between AspP and VoiceP.

(127) a. French: **V** in T b. Romanian: **V** in Asp c. Spanish: **V** in F



This is a tentative analysis in leaving open the identity of the head F; see Schifano and Roberts for relevant discussion. Still, if we assume that modal AdvPs may not occur in VoiceP, we can explain why they cannot follow the V in Spanish as in (126c), since it marks the left boundary of VoiceP as in (127c). Then, it should hold that some AdvPs may occur between the V and its object DP in Spanish, because the V leaves VoiceP, unlike in English. This is correct, as shown by examples from Roberts (2019: 350).

(128)	a.	***Antoine***	(*toujours)	***confound***	(✓*toujours*)	*ce genre de poèmes*.	[French]
		A.	always	confuses	always	this type of poem	
	b.	***Maria***	(✓*mereu*)	***face***	(✓*mereu*)	*desertul*.	[Romanian]
		M.	always	makes	always	desert=the	
	c.	***Sergio***	(✓*siempre*)	***confunde***	(✓*siempre*)	*estos poemas*.	[Spanish]
		S.	always	confuses	always	these poems	

Thus, though the entire variation across Romance languages is more complicated than described here, it is assumable that the variation in AdvP placement is affected by the variation in V placement.

Built on this perspective, we now assume that the finite Aux in French occurs in the T head, while it occupies the Asp head in English. This distinction enables the MSC to explain why only French blocks the presence of an AdvP between the subject DP and the finite Aux. For example, let us return to (124a) and (124b), assuming that their structures can be represented as follows.

(129) a. (124a): [$_{\text{TP}}$ **DP**$_{[\kappa]}$ [$_{\text{T}'}$ T$_{[\kappa]}$ [$_{\text{AspP}}$ ✓*probably* [$_{\text{Asp}'}$ Asp-***has*** [$_{\text{AuxP}}$ … ***t***$_{\text{DP}[\kappa]}$ …]]]]]

b. (124b): [$_{\text{TP}}$ **DP**$_{[\kappa]}$ [$_{\text{T}''}$ *probablement [$_{\text{T}'}$ T$_{[\kappa]}$-Asp-***a*** [$_{\text{AspP}}$ ………… ***t***$_{\text{DP}[\kappa]}$ …]]]]

As we can see, the AdvP in French must be analyzed as occurring within TP, since the finite Aux is in the T head. This configuration is excluded by the MSC, because the T head holds κ-feature matching with the lower copy of the subject DP and makes the structure T′ count as a MC. That is, the MSC does not allow T′ to be merged with any non-matching SOs such as AdvP. This result collaborates the validity of the MSC, which correctly predicts that no AdvP occurs right below a raised DP, either.

5. Conclusion

In summary, this paper addressed whether the input search for Merge is totally free (Total Search) or partially restricted (Partial Search). We discussed this issue by considering which view offers a principled account of the Anti-stacking Effect (ASE), which says that no AdvP occurs at the edge of a κ-assigner head. Although Koizumi's (1995) output filter on Merge derives the ASE and allows us to maintain the view of Total Search, we argued that his Agr-based syntax is incompatible with compositional semantics and incorrectly predicts that all DPs exhibit freezing effects. To remove these problems, we built on the view of Partial Search and proposed the Matching Search Condition (MSC), an input filter on Merge that applies to an input SO whose head holds Match internally and blocks it from merging with non-matching SOs in the workspace (WS). We demonstrated that the MSC derives all the cases of the ASE found in the domains of VP and CP and even covers a crosslinguistic difference in the distribution of AdvPs, once the variation in the height of V-raising is taken into consideration. Thus, to the extent that the MSC is empirically supported, it should be concluded that the input search for Merge is partially restricted.

Our theory therefore argues against Chomsky's (2019) Recursion Principle, which allows Merge to take any SOs at any derivational stages. Indeed, the Recursion Principle causes a theoretical complication and leads Chomsky to posit another sort of restriction, namely, *output restrictions independent of the interface conditions.* One example is the principle of *Restrict Computational Resources* (Chomsky 2019: 275), which requires Merge not to increase the cardinality of the WS and plays a role in excluding "illegitimate" cases of Merge, such as *Parallel Merge* (e.g., Citko 2005). This principle is clearly an output restriction, since it is not until Merge applies that we can check whether that application expands the WS.

However, it is not totally clear why the computational system imposes output restrictions on itself, and moreover, "illegitimate" cases of Merge, if necessary, could be excluded by input restrictions like the minimal search principle (e.g., Larson 2015).[20] What is minimally required therefore is input restrictions and interface conditions, and the applicability of Merge should only be constrained in their terms.

References

Alexeyenko, Sascha. 2015. The syntax and semantics of manner modification: Adjectives and adverbs. PhD diss., University of Osnabrück.

Arregi, Karlos, and Andrew Nevins. 2012. *Morphotactics: Basque auxiliaries and the structure of Spellout*. Oxford: Oxford University Press.

Baker, Mark. 2003. *Lexical categories: Verbs, nouns, and adjectives*. Cambridge: Cambridge University Press.

Basilico, David. 2008. Particle verbs and benefactive double objects in English: High and low attachments. *Natural Language and Linguistic Theory* 26: 731–773.

Becker, Misha. 2002. English has two copulas. *University of Pennsylvania Working Papers in Linguistics* 7(2), Article 2.

Belletti, Adriana. 1990. *Generalized verb movement*. Turin: Risenberg and Sellier.

Belk, Zoë, and Ad Neeleman. 2017. AP adjacency as a precedence constraint. *Linguistic Inquiry* 48: 1–45.

Bhatt, Rakesh, and James Yoon. 1991. On the composition of comp and parameters of V-2. *Proceedings of the 10th West Coast Conference on Formal Linguistics*, *WCCFL* 10: 41–52.

Bobaljik, Jonathan. 1995. In terms of Merge: Copy and head movement. *MIT Working Papers in Linguistics* 27: 41–64.

Bošković, Željko. 2007. On the locality and motivation of move and agree: An even more minimalist theory. *Linguistic Inquiry* 38: 589–644.

Bošković, Željko. 2014. Now I'm a phase, now I'm not a phase: On the variability of phases with extraction and ellipsis. *Linguistic Inquiry* 45: 27–89.

Bošković, Željko. 2018. On movement out of moved elements, labels, and phases. *Linguistic Inquiry* 49: 247–282.

Borer, Hagit. 2005. *Structuring sense. Volume II: The normal course of events*. Oxford: Oxford University Press.

Bowers, John. 1993. The syntax of predication. *Linguistic Inquiry* 24: 591–656.

Bowers, John. 2010. *Arguments as relations*. Cambridge, MA: MIT Press.

Bresnan, Joan. 1972. Theory of complementation in English syntax. Ph.D. diss., Massachusetts Institute of Technology.

Bruening, Benjamin. 2001. QR obeys superiority: Frozen scope and ACD. *Linguistic Inquiry* 32: 233–273.

Bruening, Benjamin. 2010. Ditransitive asymmetries and a theory of idiom formation. *Linguistic Inquiry* 41: 519–562.

Bruening, Benjamin. 2015. Depictive secondary predicates, light verb *give*, and theories of double object constructions. Ms., University of Delaware.

Bruening, Benjamin. 2021. Implicit arguments in English double object constructions. *Natural Language and Linguistic Theory* 39: 1023–1085

Chomsky, Noam. 1981. *Lectures on government and binding*. Dordrecht: Foris.

Chomsky, Noam. 1993. A minimalist program for linguistic theory. In *The view from Building 20*, eds. Kenneth

[20] Larson (2015) also proposes to restrict the input search for Merge, but Larson's proposal crucially depends on the assumption that each derivation may have multiple WSs in parallel. This is an assumption we do not share. For us, each derivation may invoke only one WS, updating its contents stage by stage.

Hale, and Samuel Keyser, 1–52. Cambridge, MA: MIT Press.
Chomsky, Noam. 1995. *The minimalist program*. Cambridge, MA: MIT Press.
Chomsky, Noam. 2000. Minimalist inquiries: The framework. In *Step by step: Essays on minimalism in honor of Howard Lasnik*, eds. Roger Martin, David Michaels, and Juan Uriagereka, 89–155. Cambridge, MA: MIT Press.
Chomsky, Noam. 2001. Derivation by phase. In *Ken Hale: A life in languag*e, ed. Michael Kenstowicz, 1–52. Cambridge, MA: MIT Press.
Chomsky, Noam. 2004. Beyond explanatory adequacy. In *The cartography of syntactic structures. Volume 3: Structures and beyond*, ed. Adriana Belletti, 104–131. Oxford: Oxford University Press.
Chomsky, Noam. 2008. On phases. In *Foundational issues in linguistic theory*, eds. Robert Freiden, Carlos Otero, and Maria Zubizaretta, 133-166. Cambridge, MA: MIT Press.
Chomsky, Noam. 2013. Problems of projection. *Lingua* 130: 33–49.
Chomsky, Noam. 2015. Problems of projection: Extensions. In *Structures, strategies and beyond: Studies in honor of Adriana Belletti*, eds. Elisa Di Domenico, Cornelia Hamann, and Simona Matteini, 1–16. Amsterdam: John Benjamins.
Chomsky, Noam. 2019. Some puzzling foundational issues: The reading program. *Catalan Journal of Linguistics*, Special Issue: 263–285.
Chomsky Noam., Ángel Gallego, and Dennis Ott. 2019. Generative grammar and the faculty of language: Insights, questions, and challenges. *Catalan Journal of Linguistics*, Special Issue: 229–261.
Citko, Barbara. 2005. On the nature of Merge: External Merge, Internal Merge, and Parallel Merge. *Linguistic Inquiry* 36: 475–496.
Collins, Chris. 2002. Eliminating labels. In *Derivation and explanation in the minimalist program*, eds. Samuel Epstein, and Daniel Seely, 42–64. Oxford: Blackwell Publishers.
Collins, Chris. 2017. Merge(X,Y) = {X,Y}. In *Labels and roots*, eds. Leah Bauke, and Andreas Blümel, 47–68. Berlin: De Gruyter.
Collins, Chris, and Edward Stabler. 2016. A formalization of minimalist syntax. *Syntax* 19: 43–78.
Corver, Norbert. 2014. [Adverbial[$_{\text{Pred}}$ *ly*]]. In *Black book. A festschrift in honor of Frans Zwarts*, eds. Jack Hoeksema, and Dicky Gilbers, 47–63. Groningen: University of Groningen.
Corver, Norbert. 2021. Adverbial *-s* as last resort. *Natural Language and Linguistic Theory*. https://doi.org/10.1007/s11049-021-09527-w
Culicover, Peter. 1991. Polarity, inversion and focus in English. *Proceedings of the 8th Eastern States Conference on Linguistics*, *ESCOL '91*: 46–68.
Culicover, Peter. 2013. *Explaining syntax: Representations, structures, and computation*. Oxford: Oxford University Press.
Davidson, Donald. 1967. The logical form of action sentences. In *The logic of decision and action*, ed. Nicholas Rescher, 81–95. Pittsburgh: University of Pittsburgh Press.
Drummond, Alex, Norbert Hornstein, and Howard Lasnik. 2010. A puzzle about P-stranding and a possible solution. *Linguistic Inquiry* 41: 689–692.
Duffield, Nigel. 2015. Where not to put why, and why not? *The Journal of Konan University. Faculty of Letters* 27: 57–68.
Elliot, Patrick. 2018. Explaining DPs vs. CPs without syntax. *Proceedings of the 52nd Annual Meeting of the Chicago Linguistic Society*, *CLS* 52: 171–186.
Emonds, Joseph. 1978. The verbal complex V-V′ in French. *Linguistic Inquiry* 9: 151–175.
Epstein, Samuel, and Daniel Seely. 2002. Rule applications as cycles in a level-free syntax. In *Derivation and ex-*

planation in the minimalist program, eds. Samuel Epstein, and Daniel Seely, 65–89. Oxford: Blackwell.
Ernst, Thomas. 2002. *The syntax of adjuncts*. Cambridge: Cambridge University Press.
Folli, Raffaella, and Heidi Harley. 2007. Causation, obligation, and argument structure: On the nature of little v. *Linguistic Inquiry* 38: 197–238.
Georgala, Efthymia. 2012. Applicatives in their structural and thematic function: A minimalist account of multitransitivity. PhD diss., Cornell University.
Hacquard, Valentine. 2006. Aspects of modality. PhD diss., Massachusetts Institute of Technology.
Hacquard, Valentine. 2009. On the interaction of aspect and modal auxiliaries. *Linguistics and Philosophy* 32: 279–315.
Hacquard, Valentine. 2010. On the event relativity of modal auxiliaries. *Natural Language Semantics* 18: 79–114.
Haegeman, Liliane. 2000a. Inversion, non-adjacent-inversion, and adjuncts in CP. *Transactions of the Philological Society* 98: 121–160.
Haegeman, Liliane. 2000b. Negative preposing, negative inversion, and the split CP. In *Negation and polarity*, eds. Laurence Horn, and Yasuhiko Kato, 21–61. Oxford: Oxford University Press.
Hallman, Peter. 2015. Syntactic neutralization in double object constructions. *Linguistic Inquiry* 46: 389–424.
Harley, Heidi. 2002. Possession and the double object construction. *Yearbook of Linguistic Variation* 2: 29–68.
Harley, Heidi. 2013. External arguments and the Mirror Principle: On the independence of voice and v. *Lingua* 125: 34–57.
Heim, Irene, and Angelika Kratzer. 1998. *Semantics in generative grammar*. Oxford: Blackwell.
Hiraiwa, Ken. 2005. Dimensions of symmetry in syntax: Agreement and clausal architecture. PhD diss., Massachusetts Institute of Technology.
Hopper, Joan, and Sandra Thompson. 1973. On the applicability of root transformations. *Linguistic Inquiry* 4: 465–497.
Janke, Victoria, and Ad Neeleman. 2012. Ascending and descending VPs in English. *Linguistic Inquiry* 43: 151–190.
Jerro, Kyle. 2016. The syntax and semantics of applicative morphology in Bantu. PhD diss., University of Texas at Austin.
Johnson, Kyle. 1991. Object positions. *Natural Language and Linguistic Theory* 9: 577–636.
Kayne, Richard. 1989. Notes on English agreement. *CIEFL Bulletin* 1: 41–67.
Koizumi, Masatoshi. 1995. Phrase structure in minimalist syntax. PhD diss., Massachusetts Institute of Technology.
Kratzer, Angelika. 1996. Severing the external argument from its verb. In *Phrase structure and the lexicon*, eds. Johan Rooryck, and Laurie Zaring, 109–137. Dordrecht: Kluwer.
Kratzer, Angelika. 1998. More structural analogies between pronouns and tenses. *Proceedings of the 31st Semantics and Linguistic Theory Conference, SALT* 8: 92–110.
Kratzer, Angelika. 2006. Decomposing attitude verbs. Handout from a talk in honor of Anita Mittwoch on her 80th birthday, Hebrew University of Jerusalem.
Larson, Bradley. 2015. Minimal search as a restriction on Merge. *Lingua* 156: 57–69.
Larson, Richard. 1988. On the double object construction. *Linguistic Inquiry* 19:335–391.
Larson, Richard. 1989. Light predicate raising. Ms., State University of New York.
Lasnik, Howard. 1992. Case and expletives: Notes towards a parametric account. *Linguistic Inquiry* 23: 381–405.
Lasnik, Howard. 1995. Case and expletives revisited: On greed and other human failings. *Linguistic Inquiry* 26: 615–633.
Lasnik, Howard. 2003. *Minimalist investigations in linguistic theory*. London: Routledge.
Lasnik, Howard, and Myung-Kwan Park. 2003. The EPP and the Subject Condition under sluicing. *Linguistic In-*

quiry 34: 649–660.

Lasnik, Howard, and Mamoru Saito. 1991. On the subject of infinitives. *Papers from the 27th Regional Meeting of the Chicago Linguistic Society*, *CLS* 27(1): 324–343.

Legate, Julie. 2014. *Voice and v: Lessons from Acehnese*. Cambridge, MA: MIT Press.

Levinson, Lisa. 2010. Arguments for pseudo-resultative predicates. *Natural Language and Linguistic Theory* 28: 135–182.

Lohndal, Terje. 2014. *Phrase structure and argument structure: A case study of the syntax-semantics interface*. Oxford: Oxford University Press.

Maekawa, Takafumi. 2006. Configurational and linearization-based approaches to negative inversion. *Empirical Issues in Syntax and Semantics* 6: 227–247

Marantz, Alec. 1993. Implications of asymmetries in double object constructions. In *Theoretical aspects of Bantu grammar*, ed. Sam Mchombo, 113–150. Stanford: CSLI Publications.

Marantz, Alec. 1997. No escape from syntax: Don't try morphological analysis in the privacy of your own lexicon. *University of Pennsylvania working papers in linguistics* 4(2): 201–225.

Matushansky, Ora. 2008. On the linguistic complexity of proper names. *Linguistics and Philosophy* 21: 573–627.

McFadden, Thomas. 2012. *For* in English infinitives has nothing to do with Case. In *Linguists of tomorrow: Selected papers from the 1st Cyprus postgraduate conference in theoretical and applied linguistics*, eds. Kleanthes Grohmann, Aljona Shelkovaya and Dionysios Zoumpalidis, 573–627. Newcastle upon Tyne: Cambridge Scholars Publishing.

Moulton, Keir. 2015. CPs: Copies and compositionality. *Linguistic Inquiry* 46: 305–342

Narita, Hiroki. 2014. *Endocentric structuring of projection-free syntax*. Amsterdam: John Benjamins.

Oseki, Yohei. 2015. Eliminating pair-merge. *Proceedings of the 32nd West Coast Conference on Formal Linguistics*, *WCCFL* 32: 303–312.

Parsons, Terence. 1990. *Events in the semantics of English: A study in subatomic semantics*. Cambridge, MA: MIT Press.

Pesetsky, David. 1989. Language-particular process and the earliness principle. Ms., Massachusetts Institute of Technology.

Pesetsky, David. 1995. *Zero syntax*. Cambridge, MA: MIT Press.

Pesetsky, David, and Esther Torrego. 2001. T-to-C movement: Causes and consequences. In *Ken Hale: A life in language*, ed. Michael Kenstowicz, 355–426. Cambridge, MA: MIT Press.

Pesetsky, David, and Esther Torrego. 2004. Tense, case, and the nature of syntactic categories. In *The syntax of time*, eds. Jacqueline Guéron, and Jacqueline Lecarme, 495–537. Cambridge, MA: MIT Press.

Pollock, Jean-Yves. 1989. Verb movement, universal grammar and the structure of IP. *Linguistic Inquiry* 20: 365–424.

Pollock, Jean-Yves. 1997. Notes on clause structure. In *Elements of grammar: Handbook in generative syntax*, ed. Liliane Haegeman, 237–279. Dordrecht: Kluwer.

Postal, Paul. 1974. *On raising: One rule of English grammar and its theoretical implications*. Cambridge, MA: MIT Press.

Potsdam, Eric. 1998. *Syntactic issues in the English imperative*. New York: Garland Publishing.

Progovac, Ljiljana. 1998. Determiner phrase in a language without determiners. *Journal of Linguistics* 34: 165–179.

Pylkkänen, Liina. 2002. Introducing arguments. PhD diss., Massachusetts Institute of Technology.

Radford, Andrew. 2018. *Colloquial English: Structure and variation*. Cambridge: Cambridge University Press.

Richards, Marc. 2007. On feature inheritance: An argument from the phase impenetrability condition. *Linguistic Inquiry* 38: 563–572.

Richards, Norvin. 1997. What moves where when in which language?. PhD diss., Massachusetts Institute of Technology.

Rizzi, Luigi. 1997. The fine structure of the left periphery. In *Elements of grammar: Handbook in generative syntax*, ed. Liliane Haegeman, 281–337. Dordrecht: Kluwer.

Roberts, Ian. 1998. *Have*/*be* raising, move F, and procrastinate. *Linguistic Inquiry* 29: 113–125.

Roberts, Ian. 2019. *Parameter hierarchies and universal grammar*. Oxford: Oxford University Press.

Ross, John. 1967. Constraints on variables in syntax. PhD diss., Massachusetts Institute of Technology.

Rubin, Edward. 1996. The transparent syntax and semantics of modifiers. *Proceedings of the 15th West Coast Conference on Formal Linguistics*, *WCCFL* 15: 429–440.

Runner, Jeffrey. 1998. *Noun phrase licensing*. New York: Garland Publishing.

Schifano, Norma. 2018. *Verb movement in Romance: A comparative study*. Oxford: Oxford University Press.

Seely, Daniel. 2006. Merge, derivational c-command, and subcategorization in a label-free syntax. In *Minimalist essays*, ed. Cedric Boeckx, 182–217. Amsterdam: John Benjamins.

Stowell, Tim. 1981. Origins of phrase structure. PhD diss., Massachusetts Institute of Technology.

Svenonius, Peter. 2003. Limits on P: Filling in holes vs. falling in holes. *Nordlyd* 31: 431–445.

Svenonius, Peter. 2010. Spatial P in English. In *Mapping spatial PPs: The cartography of syntactic structures, Volume 6*, eds. Guglielmo Cinque, and Luigi Rizzi, 127–161. Oxford: Oxford University Press.

Vikner, Sten. 1995. *Verb movement and expletive subjects in the Germanic languages*. Oxford: Oxford University Press.

Vikner, Sten. 2001. Verb movement variation in Germanic and optimality theory. Habilitation, Universität Tübingen.

Wexler, Kenneth, and Peter Culicover. 1980. *Formal principles of language acquisition*. Cambridge, MA: MIT Press.