Paths: the ghost of features past

Elise Newman & Kenyon Branan

October 10, 2022

1 Introduction - moving toward Paths

This paper develops a novel theory of locality rooted in the notion of *path*. Long distance dependencies, on this approach, must be mediated by a sequence of local dependencies.

(1) **Path**
For a probe A to enter into a dependency with a licit goal B, there must be a path of local relationships between A and B.

We propose that, underlying this notion of *path* is a more general notion of *economy* in dependency formation. Paths, as we define them, allow the grammar enough information to know whether or not a search procedure, on which long-distance dependencies are contingent, will succeed or fail.

Chomsky (2004, et seq.) suggests that *probing* involves an operation of "minimal search". Minimal, for the purposes here, means that the search procedure will halt once a match has been found, the hope being that the right specification of the search procedure will capture Relativized Minimality effects (Rizzi 1990). A number of algorithms for minimal search have been proposed and discussed in the literature, see Branan and Erlewine (2021) for an overview and Preminger (2019), Ke (2019), Atlamaz (2019), Krivochen (2022), and Chow (2022) for more specific proposals. The basic idea is that nodes in a syntactic tree are sequentially "examined" to see if they are a match for what the probe is specified to look for, with the sequential search algorithm only being able to move to sisters or daughters of failed matches.

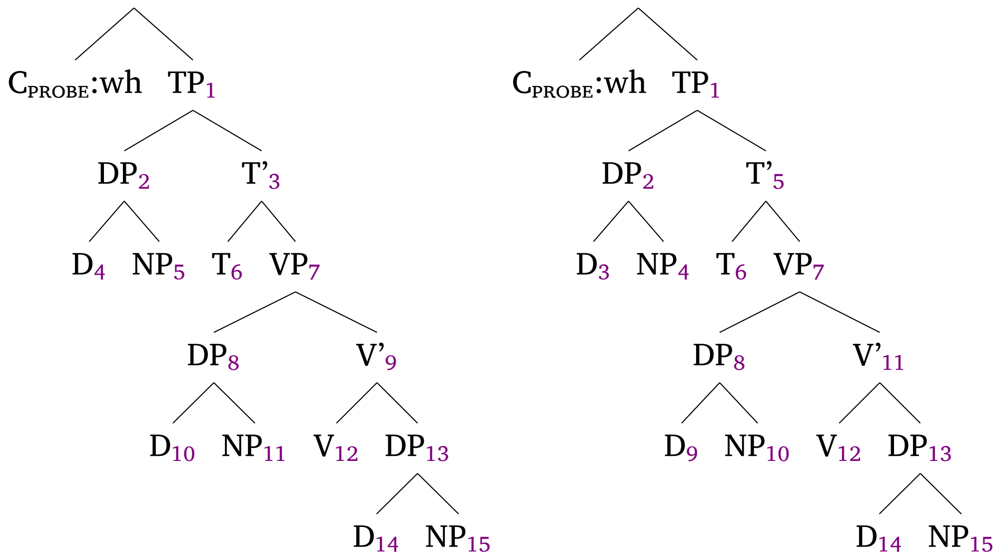
Why should search be minimal? One reason — as Chomsky suggests — might be for reasons of computational efficiency. Searching the tree involves *examining* a number of nodes to see if they are a match for the probe. Examining as few nodes as possible would be desirable, given that the process of examining a node to see if it is a match bears some computational cost.

With this in mind, consider a case like the following, involving an interrogative C seeking a matching [WH] element, which is not present in the tree. Every node in the tree must therefore be examined, regardless of the choice of algorithm.[1] In terms of com-

1. The numbering for the trees below corresponds to a pre-order breadth-first and depth-first search, in that order. Similar results obtain for the aforementioned search algorithms.

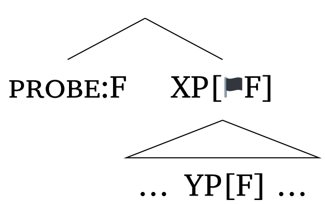
putational cost, this is the worst case scenario: every node in the tree must be examined, but doing so does not produce any observable change to the structure.

(2) **The least-efficient scenarios** *Two simple search algorithms*



We suggest that the grammar is designed to avoid costly failed searches of the type above.[2] In the abstract, we suggest that the grammar is endowed with a set of *flags* that provide (limited) information about the makeup of a constituent. For instance, in the case of a probe specified for a feature F, the daughters of a node will only be examined by the search procedure if the node itself bears [⚑F].

(3) **The probing configuration**

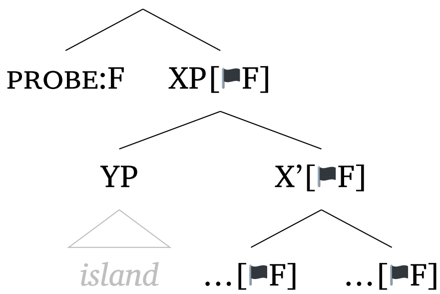


A desirable consequence of this is that it provides a fairly straightforward way of capturing *syntactic islands*. Islands, on this approach, would simply be phrases that lack a flag for the relevant sort of feature. As schematized below, the internal components of a

2. This question is ultimately orthogonal to the question of whether or not probing may fail without leading to a derivational crash (see Preminger (2014) for some discussion), and is in principle compatible with either view. Failed searches are consistently costly because they require the entire search space of a probe to be exhausted: each node must be examined to see if it matches the needs of the probe, and the examination of each node is that which bears the cost. In a world where probing may fail, knowing that a particular instance of it will fail allows the derivation to proceed to the next step without incurring the cost associated with search. In a world where failed probing leads to crash, knowing that a particular instance of it will fail allows the derivation to be thrown out without incurring the additional cost discussed above.

node lacking [⚑F] will not be subject to search. In the case below, probe-goal relationships for [F] will be impossible into YP, since it does not bear the relevant flag. Note that this entails a local relationship between a probe and its goal: every node in the sister of the probe that dominates the goal must bear a flag for a feature on that goal.

(4) **An island configuration**



This raises a number of questions, the most pressing of which this paper seeks to answer. In §2, we suggest that the *flags* in (4) are checked selectional features — presumably an independently necessary component of the grammar. Checked selectional features provide a record of the derivation — the presence of a checked selectional feature on maximal projection serves as a *flag* that either the specifier or complement of that phrase is of a particular sort, assuming that checking takes place under sisterhood. The chief innovation is an algorithm for determining whether or not checked selectional features are able to project past the maximal projection of the head they originated on. Crucially, this decision is *local*: it creates paths of local relationships between a probe and a licit goal, in the sense of (1). We show that the theory captures the basics of the classic CED: adjuncts and specifiers are, in the basic case, opaque for extraction, while complements are not. We show also that the theory avoids what we term the "escape hatch problem" for phase-based approaches to the CED, a stipulation which requires adjunct islands to both be phases and consistently lack an edge feature.

§3 and §4 show that the theory developed here has significant empirical bite. In §3, we show that the theory under development makes nuanced predictions about whether an adjunct will behave as an island or not: the local context that an adjunct appears in determines its opacity. We note that the presence of certain dependencies into a class of control adjuncts — such as *wh*-movement and parasitic gaps — consistently forces the control adjunct to receive an Obligatory Control interpretation, despite the Non-Obligatory Control interpretation being available in other contexts. In §4, we discuss two additional data points that support our theory. We show that specifiers, too, may be rendered transparent for extraction, based on cases of "melting" first discussed in Müller (2010). We show that the presence of an adjunct, in certain cases, may render the phrase it is adjoined to opaque for extraction, based on cases of long-distance scrambling in Balkar first discussed in Privoznov (2021). In §5, we discuss a number of open issues for the theory developed here, and sketch some possible answers to them.

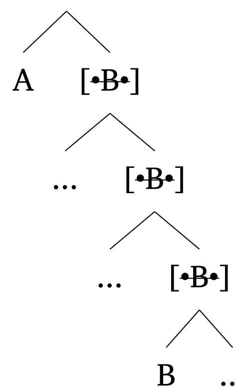
2 A theory of locality

As discussed in §1, we propose that a notion of *path* mediates Search. As Search underlies the establishment of long-distance dependencies, paths become preconditions for long distance dependencies by extension. More specifically, for a probe A to establish a dependency with a goal B, we propose that A's sister must bear a feature checked by B. An algorithm for projecting features checked by B from the head that selected B to the sister of A establishes a series of local relationships that links probe and goal.

(5) **Accessibility**
A probe A searching for a goal B may only initiate Search for B if there is a path from A to B.

(6) **Path** (shorthand)
There is a Path from A to B if A's sister bears a feature checked by B.

(7) A long-distance path from A to B

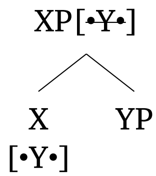


If for some reason A's sister does not bear a feature checked by B (i.e. because there is no local B), Search fails at the outset, without examining any nodes in the tree. Thus, Search never applies unnecessarily.

We begin by establishing some assumptions about clause construction, and show how a modified theory of feature projection creates long distance dependencies according to (5) and (6). Adopting the notation of Müller (2010), we represent the features that drive Merge as in (8). A head that selects for a YP, for example, might bear a feature [•Y•], which may be checked when that head (or a projection of it) merges with a YP. More concretely, a head with an unchecked [•Y•] feature that merges with a YP produces a projection bearing the checked version of that feature, [•~~Y~~•], as in (9). As will become important later, we follow Müller (2010) in assuming that these features drive *any* kind of Merge, representing not only external Merge, but movement (internal Merge) as well.

(8) **Merge features**
[•Y•] = an instruction to Merge with an element bearing Y

(9) **Selection for YP**



These checked features are inactive in the sense that they may no longer drive syntactic operations. However, contra Adger (2003) and Asudeh and Potts (2004), a.o., we suggest that they do not disappear from the derivation. Instead, we propose that they remain present throughout the computation to serve as a pseudo-record of selection.[3] On this view, XP provides more information to higher heads than just its own category feature. It also bears checked Merge features, which tell higher heads something about the elements inside XP, for instance that XP contains a YP in the case of (9).

So far, we have seen how checked features may be projected by a head to its own maximal projection, when it merges with elements it selects for. These features do not delete, and are thus visible to whatever subsequently merges with XP. According to the conditions in (5) and (6), for YP to be accessible to anything beyond XP's sister, [~~•Y•~~] must be able to project *past* XP. Only if [~~•Y•~~] projects past XP can it ever appear on the sister to a higher probe, making YP accessible to that probe.

We propose that feature projection past maximal projections is conditioned by the local context of that maximal projection. More specifically, maximal projections whose sisters are what we call *Indivisible Feature Bundles* get to project their checked selectional features to higher nodes, making their contents accessible to later operations (10). Indivisible features bundles are defined in (11) – they are essentially feature bundles whose features locally come from a single source. The intuition guiding this approach is the belief that language is binary: feature projection should only project two bundles of features at a time. If one sister is already projecting two feature bundles, the other cannot project at all. If one sister projects one or fewer feature bundles, the other can project one as well.

(10) **Feature projection**
A feature bundle {[~~•F•~~],[~~•G•~~]...} on a maximal projection may project iff its sister is an *indivisible feature bundle*.

3. Some might worry that having the computation store checked selectional features is costly, because it requires the derivation to carry around more information than if these features were deleted. However, this approach saves us from having a separate *deletion* operation, which would have to apply at every instance of Merge in order for these features *not* to stick around. We thus trade a derivational burden (the addition of deletion operations) for a storage one (the addition of stored information).

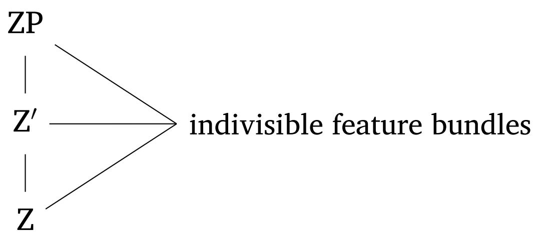
(11) **Indivisible feature bundle**:

a. a feature bundle that comes straight from the lexicon
→ e.g. a terminal node (Matushansky 2006), OR

b. a feature bundle that has projected to a node from only one daughter

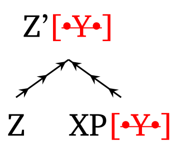
As an illustration, consider the tree in (12), in which a terminal node projects straight to a maximal projection without merging with anything else. The terminal node is an indivisible feature bundle because it comes straight from the lexicon. The Z′ node dominating it is also an indivisible feature bundle because it only projects from a single daughter. The ZP node that dominates them both is likewise an indivisible feature bundle, for the same reason. In this simple case, the head Z is functionally equivalent to the ZP that it projects (Chomsky 1995) – both are indivisible feature bundles.

(12) Indivisible feature bundles



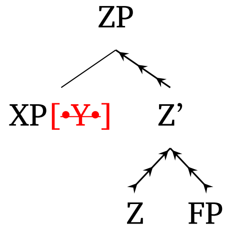
Recalling the XP maximal projection from (9), we can now calculate the predicted effects of context on whether XP gets to project its [•~~Y~~•] feature to higher nodes. If XP is the first-merged element with a head (othwerwise known as a *complement*), as in (13), the rule in (10) states that XP can project its [•~~Y~~•] feature – its sister is an indivisible feature bundle.

(13) XP projects [•~~Y~~•] to a higher node if it is a complement



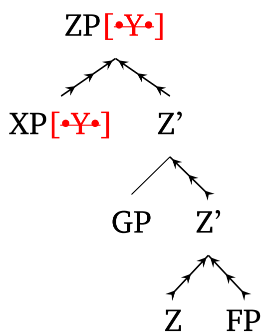
If XP is the second-merged element in ZP, i.e the first specifier of ZP (14), its sister is *not* an indivisible feature bundle. The Z′ sister to XP projects from two daughters: the terminal node (which always projects) and its sister (complements get to project, according to (13)). Since Z′ is not an indivisible feature bundle, XP does not get to project [•~~Y~~•] in this context, rendering YP inaccessible to operations external to XP. The theory thus accounts for basic CED effects: complements permit subextraction but first specifiers do not.

(14) XP does not project [•Y•] to a higher node if it is a first specifier



The theory makes a surprising prediction for third-merged elements, however. If XP merges as a second specifier (15), rather than a first specifier, its sister now only projects from *one* daughter. The sister to XP in this case only projects from one daughter because first specifiers cannot project, according to (14). The first node that dominates a first specifier is therefore an indivisible feature bundle according to (11b), which licenses projection of a second specifier.

(15) XP projects [•Y•] to a higher node if it is a *second* specifier



This approach therefore has an on-again off-again profile. If some maximal projection is allowed to project, it often creates a context in which the next merged maximal projection cannot project. If a maximal projection does not project, it often creates a context in which the next merged maximal projection can project, and so on. Thus, we expect the time of Merge to determine transparency for higher operations more than the complement-specifier distinction. We leverage this context sensitivity to explain the variable opacity of adjuncts and specifiers in different contexts.[4]

In sum, we propose that the distribution of checked features on nodes creates paths between probes and goals, where paths are a precondition for Search. A probe whose sister bears a feature checked by its goal may initiate Search for that goal, in which each successive node is examined for features checked by the goal until the goal is found. If

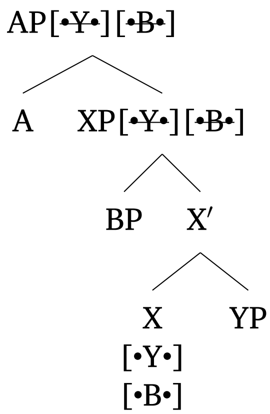
4. A question arises: what happens in the case of "simple" phrases, e.g. phrases that have themselves failed to select anything? One approach would be to deny the existence of simple phrases of this sort: on this view, every functional item would enter into some sort of selectional dependency with something else, while lexical items would minimally consist of a root and categorizing head (see Marantz (1997) for a proposal along these lines). Another approach would be that they fail to project features, which could potentially have consequences down the line if the phrase that they are a complement of later takes a specifier: the first specifier in this case should be allowed to project in the way a complement normally would. We leave investigation of these possibilities to future research.

the probe's sister has no relevant checked features, Search fails before it starts, avoiding unnecessary and costly searches. We proposed that the distribution of checked features is controlled by the rules of feature projection outlined here: maximal projections may project their checked features if their sisters are indivisible feature bundles but not otherwise. Successive projection of checked features creates paths.

Important to note is that [~~•Y•~~] is not equivalent to the YP that checked it. [~~•Y•~~] is a feature that was checked/rendered inactive by an element bearing Y. By contrast, YP is a phrase that can check some set of features on a probe, including [•Y•]. Thus, a probe whose sister bears [~~•Y•~~] has not "found" YP before searching – it must still search for the YP that checked the feature in order to satisfy the probe.

Lastly, though the presentation here only discussed a case where *one* checked feature was projected, we assume that feature projection is *wholesale* in general. What we mean by this is that multiple checked features on a phrase get projected together as a bundle – a maximal projection cannot selectively project some of its features but not others.

(16) **Projection is wholesale**



Because projection is wholesale, we expect maximal projections to be opaque or transparent to higher operations in a very general sense.[5] A transparent maximal projection is transparent for potentially multiple dependencies across itself – it projected every feature it had, so everything inside it that checked a projected feature is visible to higher heads. An opaque maximal projection is similarly opaque for every imaginable dependency – if a maximal projection projects no features past itself, there can be no paths leading into it. We will see that this all or nothing approach captures correlations between different dependencies that cross adjunct boundaries.

5. We could, of course, imagine more elaborate theories of feature projection that don't require wholesale feature projection of the sort here. The consequence of this would be that some domains would be transparent for some dependencies but not others (see Keine (2019) for some discussion of such patterns). We acknowledge this here as a point of interest for future work, but do not propose such elaborations here.

3 Adjunct (non-)islands

We now have a theory that predicts that maximal projections may be transparent for dependencies into them in certain configurations (for instance, when merged as a complement or to a phrase that already has a specifier). In this section, we discuss two cases where dependencies into adjuncts appear to be correlated, a phenomenon we term *correspondent transparency effects*.

First we discuss an observation from Truswell (2007) that wh-movement out of adjuncts tracks the obligatory/non-obligatory control distinction: control adjuncts that are transparent for wh-movement are obligatorily controlled, while control adjuncts that are opaque to wh-movement are non-obligatorily controlled. We argue that this correlation follows from our view of feature projection if both wh-movement and control are dependencies that employ Search.

Second, we observe that parasitic gaps also track the obligatory/non-obligatory control distinction. We argue that Nissenbaum (2000)'s independently proposed structures for parasitic gap constructions are configurations in which an adjunct should be transparent for multiple dependencies, such as binding of an operator and obligatory control. Thus the same explanation that accounts for correlations between wh-movement and control extend to parasitic gaps and control.

The obligatory/non-obligatory distinction is shown in (17).

(17) a. The flower$_i$ is open [PRO$_i$ to attract passing pollinators].
b. The door$_i$ is open [PRO$_{arb}$ to listen to confessions].

The non-agentive, inanimate subjects in (17) may corefer with an embedded PRO, as in (17a), or not, as in (17b). In the latter case, the embedded PRO is interpreted as referring to an arbitrary individual/group who might serve as a *listener* in this context. Insights from Chomsky (1981), Williams (1992), and Landau (2013, 2021) teach us that an inanimate PRO is sensitive to c-command by a controller, while animate PRO is not. To reflect this difference, we call (17a) a case of *Obligatory* control (henceforth OC), and (17b) a case of *Non-obligatory* control (henceforth NOC).[6]

McFadden and Sundaresan (2018) have argued that obligatory and non-obligatory control, despite appearances in (17), are in complementary distribution. On that view, a better description of (17) would be that (17a) is a case of genuine control, while (17b) is what happens when control cannot be established (i.e. an *elsewhere* construction). This view of the obligatory/non-obligatory control distinction is further motivated by the observation that OC adjuncts are transparent for wh-movement, while NOC adjuncts are opaque.

6. Interestingly, these control adjuncts show a Weak Island effect – they permit extraction of a DP but not an adjunct.

(i) *How did the flower open [in order to attract pollinators ____]
→ A: with a particular UV pattern

While we do not offer a theory of Weak Island-hood here, see Appendix A for some possible views of Weak Islands on the present theory.

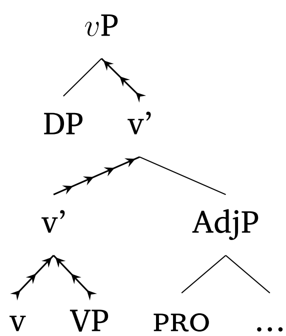
(18) a. What is the flower$_i$ is open [PRO$_i$ to attract ____]?
b. *What is the door$_i$ is open [PRO$_{arb}$ to listen to ____]?

The examples in (18) show that infinitival adjuncts can either be fully transparent for wh-movement and control or fully opaque. We now propose that this variable transparency of control adjuncts results from a structural ambiguity in their attachment sites. When the adjunct attaches below the subject in Spec vP, it is a first specifier, unable to project its features, which renders it opaque. When it attaches above the subject in Spec vP, it is a second specifier, permitted to project its features, which makes it transparent.

3.1 Adjunction sites and feature projection

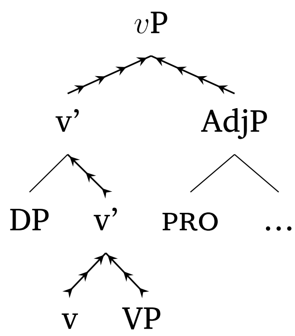
Following Landau (2021) and references there, we propose that controlled adjuncts are vP-level adjuncts. However, their exact adjunction position within vP is unspecified – we know they are not complements of v (VP is), but whether they attach above or below the base position of the external argument is undetermined. The feature projection algorithm makes different predictions for each choice: adjuncts that merge as a first specifier (below the subject) should not project, while adjuncts that merge as a second specifier (above the subject) should project.[7]

(19) a. Below the subject: sister projects from two daughters – adjunct can't project



b. Above the subject: sister projects from one daughter – adjunct can project

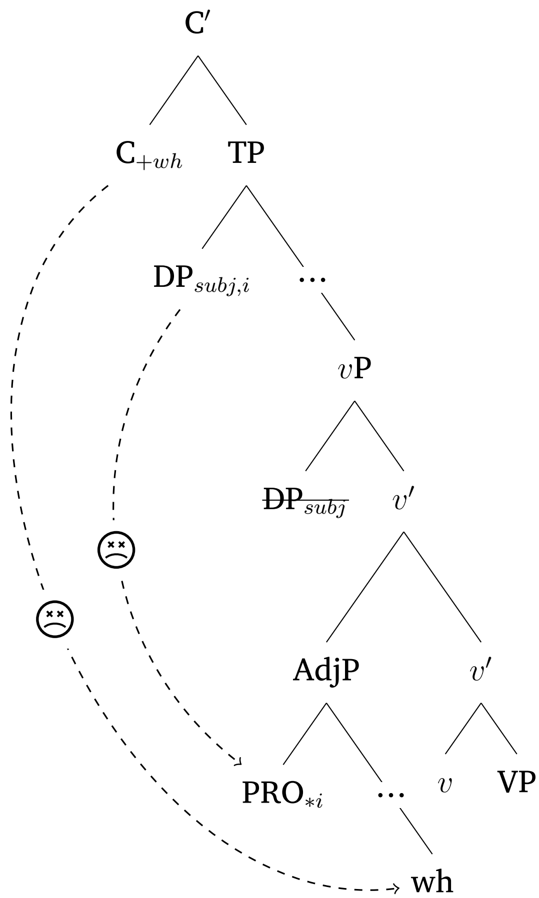
7. Crucially, because the adjunct has a (single) specifier, it only projects features from one daughter, and is therefore an indivisible feature bundle. As a result, adjunction does not block the vP from projecting its own features, regardless of whether vP is maximal at the time of adjunction. If the adjunct were *not* an indivisible feature bundle, we would expect adjunction in certain positions to block long distance dependencies within the matrix clause, which we do not observe here, but will discuss further in §4.2. As to what enforces the adjunct having a specifier in these cases: this could either be due to infinitival adjunct clauses being a certain size or because PRO moves to the edge of the adjunct clause for interpretability reasons (Heim and Kratzer 1998, p.226-228).



We suggest that variable projection from controlled adjuncts accounts for their variable transparency to OC and wh-movement. First, we propose that both *wh*-movement and OC control involve the establishment of a long-distance syntactic dependency through Search. *Wh*-movement arises when interrogative C searches its complement for a phrase bearing a [wh] feature, which is used to check a [•wh•] feature on itself. OC control likewise involves a syntactic dependency formation that is also contingent on successful Search, where the complement of a potential binder for PRO is subject to Search for PRO (see Ke 2019 for a complimentary proposal for reflexive binding). Consequently, there must be a *path* between PRO and its controller for OC to arise, and a path between interrogative C and a *wh*-phrase for movement to occur. When an adjunct projects its features, there are paths into it for every feature that it projects, as illustrated in (20) and (21).

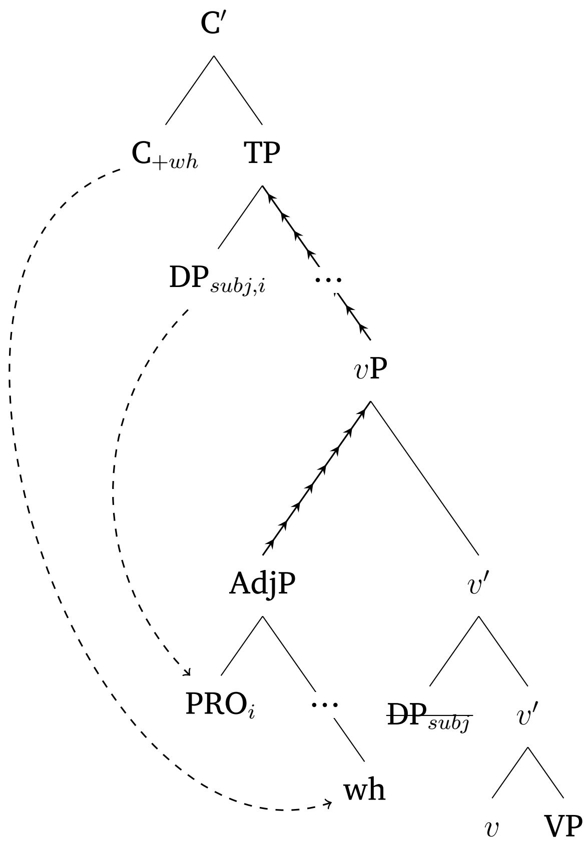
In (20), we see the continuation of the derivation in (19a); the matrix subject raises to Spec TP and an interrogative C is merged. According to (6), in order for the matrix subject to control adjunct PRO, its sister (T′) must bear a feature checked by PRO. Similarly, in order for interrogative C to attract a wh-element from inside the adjunct, its sister (TP) must bear a feature checked by the wh-element. In this case, the adjunct has merged as a first specifier, which prevents it from projecting its features to vP, let alone to T′ and TP. Thus, the lack of paths into the adjunct for any feature blocks both control into and wh-movement out of it.

(20)



In (21), we find the continuation of the derivation in (19b), where the adjunct merged as a second specifier. In this case, the adjunct was allowed to project its features to vP. When vP is merged as the complement to the next highest functional head, vP projects its features to the next highest node, which include those of the adjunct. Eventually, these features project up to T′ and TP, creating paths into the adjunct for control and wh-movement.

(21)



The adjunct is thus, in principle, transparent for subsequent Search. Both PRO and the *wh*-phrase within the adjunct should be visible, provided they have checked a feature in the adjunct that some higher head has initiated Search for. In the following subsection, we discuss what those features might be for both the *wh*-phrase and PRO, and discuss some broader implications of this for our theory. Having clarified the theory under development, we then return to our second case study: parasitic gap containing adjuncts.

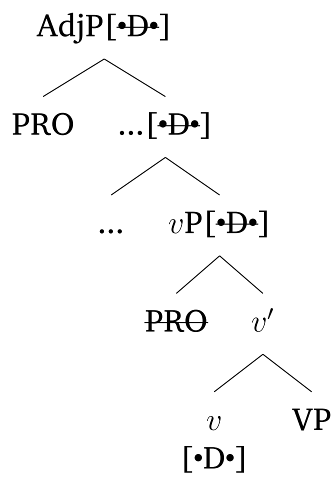
3.2 Dependencies through paths

On the theory developed here, the requirement for there to be a path between two elements "linked" through Search should be seen as a way to ensure that Search will succeed. We now explain how Search interacts with movement to account for the correspondent transparency effects. Recall, as discussed beforehand, that both internal and external Merge are licensed only when they check [•F•]s. Movement — or internal merge — requires an invocation of Search on the sister for some matching feature, followed by Merge of the result of Search at the root of the tree.

Not only must there be a path of checked features between the two elements in question, but the target of Search must have checked the sort of feature that Search is looking for. In other words, a probe with a feature [•X•] must find a path of [~~•X•~~] features to its goal, not just any path of features checked by its goal.

At this point, one might wonder *which* features actually establish these paths between the matrix subject/PRO and C/the wh-element, and how those features get checked/projected. For PRO, the answer is straightforward: PRO is presumably selected as the external argument of the adjunct clause. We can therefore imagine that it checks a [•D•] feature on adjunct v, which gets projected to the highest node dominating the adjunct clause. We have illustrated PRO as the highest specifier of AdjP on the assumption that PRO moves to the edge of its clause (Heim and Kratzer 1998). As long as control is mediated by a search for DPs, if that [~~•D•~~] feature projects to the sister of the matrix subject, the matrix subject may find and control PRO.[8]

(22) PRO checks [•D•] on v, which projects to AdjP



For wh-elements, the picture is slightly complicated by the fact that wh-features are not commonly viewed as being selected. When a wh-object merges, for example, we don't usually assume that it checks a [•wh•] feature as well as a [•D•] feature, in which case a [~~•wh•~~] feature should never project to the sister of any wh-probe. Assuming that wh-movement is mediated by the search for wh-features, what establishes the path between C and wh-elements, when wh-elements don't check [•wh•] features in their base positions?

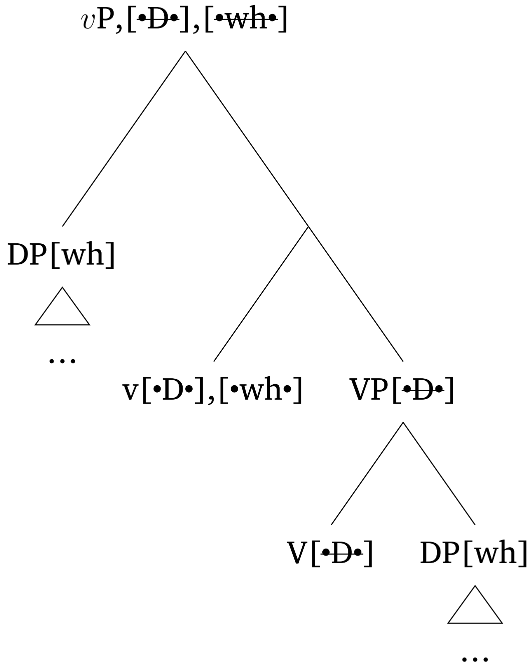
One possibility would be to propose that complement *wh*-phrases generally undergo a short step of [•D•]-driven movement to an intermediate position in the clause.[9] For concreteness, consider the case below. Here, *v* bears both a [•D•] and [•wh•] feature. Internal merge to satisfy [•wh•] is not possible: the complement of v does not bear a [~~•wh•~~] feature, so it may not be searched for [wh]. The complement does, however, bear a [~~•D•~~] feature, so it may be searched for an element bearing [D], in which case the object will be found. Subsequent merge of the object in spec,vP will check both the [•D•] as well as the [•wh•] on v. Subsequently, the [~~•wh•~~] will be able to project higher in the tree

8. An equivalent alternative is that whichever feature attracts PRO to the edge of the adjunct clause is what establishes the path between the matrix subject and PRO. If that feature is also [•D•], however, there is no meaningful difference between the two options. If some other feature is responsible for adjunct-internal movement of PRO, then some other feature could be responsible for the control path, but we won't speculate about what that feature could be here.

9. See Canac Marquis 1994 for a consonant proposal where object Ā-chains consistently involve a short step of A-movement within VP.

from this vP, creating a path between the wh-phrase in spec,vP and higher elements in the tree.

(23)



We tentatively adopt this suggestion, but acknowledge that this is not the only way out of the conundrum. We could, for instance, imagine denying that the problem arises in the first place: arguments on this view would generally be introduced as the specifiers of functional heads (see Ahn, to appear for a recent argument to this effect), which would always be potential bearers of a [•wh•]. Or we could propose that *wh*-elements are always themselves internally complex (see Cardinaletti and Starke 1994 for the proposal that pronominal elements are syntactically complex, Hagstrom 1998; Cable 2010 for consonant proposals about the syntactic complexity of *wh*-elements, and Nicolae and Scontras 2018 for an explicit proposal along these lines for Tagalog *wh*-questions). On this approach, DP-construction would involve checking of a [•wh•] feature contained within the *wh*-element itself, which would then be able to project up the tree. Given that these possibilities are not mutually exclusive, more work is ultimately needed to tease apart the predictions made by these theories, and determine which, if any, are unattested options for the grammar.

The proposal that wh-movement is mediated by DP-movement raises questions about wh-movement of non-DPs, such as PP arguments and adjuncts. Here, again, a number of ways forward present themselves.

(24) a. To whom did John first speak?
b. On which day did John first speak?

For adjuncts, a fairly straightforward analysis would be to propose that they consistently externally merge in spec,vP, at least in cases where they undergo *wh*-movement.

For argument PPs the way forward is less straightforward. One possibility is that wh-PP arguments, like adjuncts, often have the option of initially merging with a functional head like vP, ensuring that the PP argument checks a *wh* feature (see Newman 2021 for a proposal along these lines). Another possibility is that PP arguments are required to exit the VP for independent reasons (see Stowell 1981 for such a proposal). Subsequent Merge of a PP argument with vP consequently check v's [•wh•] feature in cases where the PP bears [wh].

In sum: both PRO and *wh*-elements must respectively check [•D•] and [•wh•] if they are to be visible for subsequent Search operations. In the case of PRO, this is relatively trivial: PRO checks a [•D•] when it is initially merged. In the case of *wh*-elements, this means that the *wh*-element must first undergo movement for independent reasons to an intermediate projection, with checking of [•wh•] on this intermediate position taking place as a side effect. Only after movement to such a position will there be a path of [~~•wh•~~] features to the *wh*-element, rendering it visible for subsequent search.

3.3 Parasitic gaps

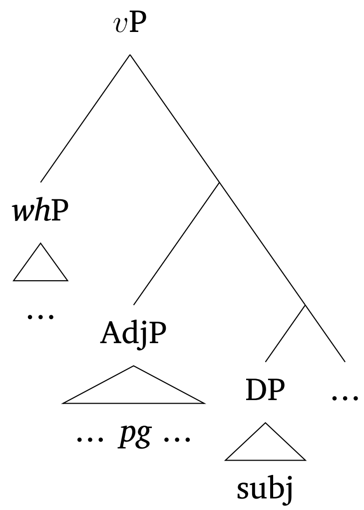
We have discussed how wh-movement out of adjuncts correlates with obligatory control into them, and have proposed that our theory of projection accounts for this correlation: adjuncts that project their features are transparent for multiple dependencies across them, while adjuncts that do not project their features are opaque for every imaginable dependency. We now observe that this correlation between wh-movement and control is not specific to wh-movement out of adjuncts. Parasitic gaps inside adjuncts show the same effect. Observe in (25) that parasitic gaps are possible in OC adjuncts, but not in NOC adjuncts.

(25) a. [What direction]$_i$ was the flower$_j$ opened to ~~what~~
[OP$_i$ PRO$_j$ in order to attract passing pollinators from ~~OP~~]?
b. * [What sort of person]$_i$ was the door$_j$ opened to ~~what~~
[OP$_i$ PRO$_{arb}$ in order to listen to confessions from ~~OP~~]?

Before, we saw that OC adjuncts are transparent as a consequence of the position that they are merged in, namely: as second specifiers. Adjuncts that are adjoined in such a position are transparent both for control as well as *wh*-movement, while the same adjunct merged in a different position will be opaque for both control and *wh*-movement, forcing an NOC interpretation of PRO.

Nissenbaum (2000) develops a theory of parasitic gap licensing that, juxtaposed with the proposals put forth here, straightforwardly captures the correlation in (25). For independent reasons, Nissenbaum proposes that parasitic gap containing adjuncts must merge in a particular position: immediately above the subject that controls PRO, and immediately below a position occupied by the *wh*-element which licenses the parasitic gap. In other words: parasitic gap containing adjuncts must be second specifiers of vP, schematized below.

(26) **Nissenbaum's parasitic gap licensing structure**



Recall that our theory of feature projection predicts second specifiers to be able to project their features to higher nodes, making them transparent to dependencies like wh-movement and control. The adjunct in (26) is therefore predicted to be transparent for control, because it is a second specifier. NOC adjuncts must not be in this position, since they block parasitic gap licensing, thus providing further motivation for the treatment of NOC as an elsewhere construction when an infinitival adjunct is opaque for a control dependency.

The fact that parasitic gap licensing is contingent on this structure also supports a view of operator binding as requiring Search, just like control and wh-movement. We follow Chomsky (1986), Larson (1988), Postal (1998), and Nissenbaum (2000) in assuming that parasitic gap constructions do not involve ATB wh-movement out of both matrix and adjunct clauses, but rather involve binding of an operator that moves adjunct-internally. By merging in second specifier position, the adjunct makes the features of PRO as well as the features of the operator accessible to higher elements. The configuration that licenses control thus should also license binding of the operator. If the adjunct merges in a different position, one which blocks feature projection, we expect both OC and operator binding to be blocked, as we find in (25b).

What we have seen, then, is that multiple kinds of dependencies which Search plausibly underlies — control and Ā-dependencies such as *wh*-movement and binding of null operators — are allowed into adjuncts only when those adjuncts appear in a particular context. Moreover the theory captures the fact that one and the same adjunct clause may be opaque or transparent, given that such clauses may merge as second specifiers or not. In §4 we discuss further implications of the theory we've developed, and compare it to other theories with comparable empirical coverage. In the section that follows, we discuss two other cases where the transparency of a domain appears to be contextually determined that our theory captures in a fairly straightforward manner.

4 Extensions

In this section we discuss two extensions of our theory to cases distinct from those discussed above. Both involve cases where the transparency of a domain for extraction is contingent on whether or not it is a second specifier. We first discuss *melting* Müller (2010), in which sub-extraction from a subject is licit only when some other phrase has passed through the position that introduces the subject. The second case, involving adverbial clauses in Balkar (Turkic; Russia), highlights a peculiar feature of the theory developed here: the presence of an adjunct clause may render the phrase it is merged with opaque.

4.1 Melting

Müller (2010) discusses a class of exceptions to the CED, which he calls *Melting* effects. He observes that external arguments in German and Czech are typically opaque to extraction, as expected for specifiers, according to the CED. However, he shows that scrambling an object to the left of the external argument has the effect of making the external argument transparent for extraction. In other words, object scrambling obviates the CED for transitive subjects. This is shown in (27) and (28) for German and Czech respectively. Wh-extraction out of the subject is only available when the object appears to its left.[10]

(27) *German wh-extraction* (ex.36)

a. *Was$_1$ haben [$_{DP3}$ t_1 für Bücher] [$_{DP2}$ den Fritz] beeindruckt?

what have for books.NOM the Fritz.ACC impressed

intended: "What kind of books impressed Fritz?"

10. Müller observes that this effect is not limited to extraction of a DP, but also of PPs.

(ii) *German PP extraction* (Müller 2010, ex.37)

a. *[$_{PP1}$ Über wen] hat [$_{DP3}$ ein Buch t_1] [$_{DP2}$ den Fritz] beeindruckt?

about whom has a book.NOM the Fritz.ACC impressed

intended: "About whom did a book impress Fritz?"

b. [$_{PP1}$ Über wen] hat [$_{DP2}$ den Fritz] [$_{DP3}$ ein Buch t_1] t_2 beeindruckt?

about whom has the Fritz.ACC a book.NOM impressed

"About whom did a book impress Fritz?"

(iii) *Czech PP extraction* (ex.44)

a. *[$_{PP1}$ O starých autech] oslovila [$_{DP3}$ kniha t_1] Petra$_2$.

about old cars fascinated book.NOM Petr.ACC

intended: "A book about old cars fascinated Petr."

b. (?)[$_{PP1}$ O starých autech] oslovila Petra$_2$ [$_{DP3}$ kniha t_1] t_2.

about old cars fascinated Petr.ACC book.NOM

"A book about old cars fascinated Petr."

b. Was$_1$ haben [$_{DP2}$ den Fritz] [$_{DP3}$ t_1 für Bücher] t_2 beeindruckt?
what have the Fritz.ACC for books.NOM impressed
"What kind of books impressed Fritz?"

(28) *Czech split DP constructions* (Müller 2010, ex.42)

a. *Stará$_1$ neudeřila [$_{DP3}$ žádná t_1] Petra$_2$.
old.NOM hit no.NOM Petr.ACC
intended: "No old one hit Petr."

b. (?)Stará$_1$ neudeřila Petra$_2$ [$_{DP3}$ žádná t_1] t_2.
old.NOM hit Petr.ACC no.NOM
"No old one hit Petr."

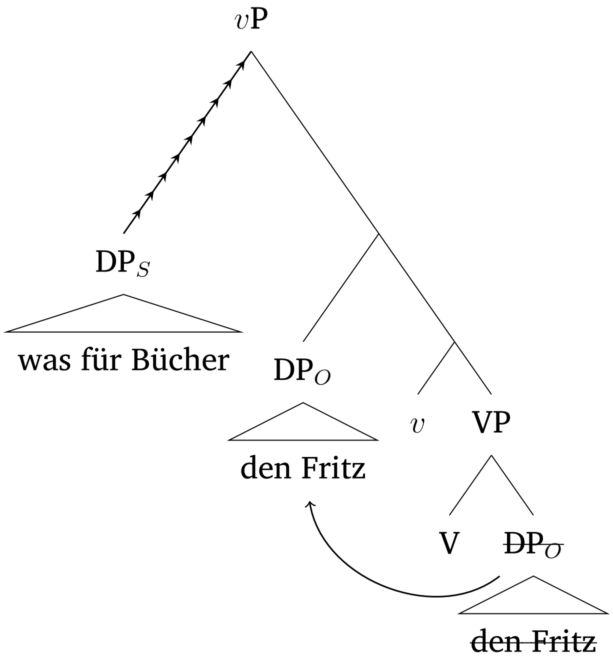
Importantly, Müller cites evidence from Grewendorf (1989) suggesting that the subject of a psyche verb like *beeindrucken* is a regular external argument in German, and not a VP-internal argument. Thus, it must be a specifier, making (27b) a true counterexample to the CED. What is surprising about (27) and (28) is that the exact same specifier (e.g. *was für Bücher*) can be opaque in (27/28a) but transparent in (27/28b), solely based on the position of the *object*. The surface position of the object presumably does not affect the specifier-hood of the subject, suggesting that island effects have more to do with local context than the complement/non-complement distinction.

Our theory provides a natural explanation for this effect on the assumption that object movement proceeds successive cyclically through the edge of vP, as discussed in §3.2.[11] Assuming that a scrambled object must stop in the edge of vP at some point in the derivation, the (b) examples in (27) and (28) differ from the (a) examples with respect to the total number of specifiers vP can have. When no scrambling takes place, the external argument is the only argument to ever occupy the edge of vP, while in scrambling derivations, vP has *two* specifiers at some point in the derivation.

Our theory predicts that first specifiers of vP should be opaque for extraction but second specifiers should be transparent. In non-scrambling derivations, the external argument is the first (only) specifier of vP, and is thus correctly predicted to be opaque. In scrambling contexts, by contrast, as long as the specifier configuration in (29) is allowed, the external argument may be a second specifier, which makes it transparent for extraction. We assume that Spec vP is only an *intermediate* landing site for the object – it eventually moves to a higher position to derive the surface word order OS, as shown in the full derivation in (30).

(29) A moving object can make the external argument a second specifier of vP, licensing (30)

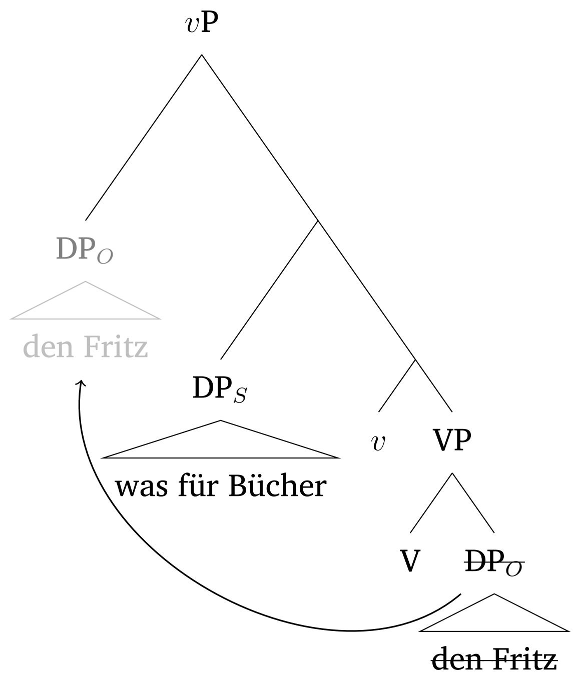
11. Following Legate (2003) and Sauerland (2003), we propose that the requirement to move successive cyclically through Spec vP transcends the A/Ā-distinction, thus side-stepping the question of whether scrambling has A or Ā-properties.



(30) $[_{CP}$ Was$_1$ haben $[_{DP2}$ den Fritz$]_2$ $[_{vP}$ $[_{DP3}$ t_1 für Bücher$]$ t_2 $[_{VP}$ t_2 beeindruckt $]$?

This proposed interaction between scrambling and extraction from specifiers is similar in spirit to Müller (2010)'s analysis, with some key technical differences. Müller presents a phase-based theory of the CED, in which phases can only produce escape hatches as long as they are incomplete. The last-merged element in a phase completes the phase, and blocks it from producing an escape hatch. As a result, his theory predicts that only the last-merged specifier of a phase is opaque for extraction. All earlier-merged material is transparent, including specifiers, because they merge early enough for an escape hatch to be produced. In non-scrambling contexts, the subject is the last-merged specifier of vP, while in scrambling contexts, Müller proposes that the object is the last-merged specifier of vP, making the subject transparent. In other words, his theory requires the opposite configuration of specifiers in vP in order to capture melting effects.

(31) Müller's vP in Melting contexts: only the highest specifier is opaque $\rightarrow$ highest specifier must be the scrambled object



Following Moltmann (1990), Grewendorf and Sabel (1999), McGinnis (1999), and Yoshida (2001, a.o.), with evidence from quantifier scope and the position of negation and adverbs, vP is not the final landing site for scrambled objects – Spec TP is. Regardless of the order of specifiers of vP, we therefore expect the object to be able to surface in a position that derives the surface word order OS. Both Müller's derivation and ours therefore make the same predictions here – the only difference is that our theory requires an earlier stage of SO order at Spec vP while Müller's requires an earlier stage of OS. Since we know of no diagnostics that could distinguish these two theories in German and Czech, it seems like both are equivalent analyses of melting.

However, these are *not* equivalent analyses of variable island-hood in general. We saw in §3 that parasitic gaps and OC are licensed for adjuncts that merge as second but not first specifiers, based on Nissenbaum (2000)'s proposed structure for parasitic gap constructions. Müller's theory is unable to account for such a pattern. In a parasitic gap construction, vP has three specifiers, with the matrix wh-phrase forming the outermost specifier. We would therefore expect all inner specifiers to be transparent according to Müller, contrary to fact.

Of course, Müller's theory is specifically about extraction, not necessarily operator binding or control, in which case the facts from §3 don't necessarily disprove his approach to melting. We could imagine that phase-hood is only relevant for certain kinds of dependencies, like extraction, in which case Müller's theory would have nothing to say about the distribution of operator binding and control. Either way, our theory has more empirical coverage because it captures both melting and the correspondent transparency effects of §3.

4.2 Balkar Converbs

Privoznov (2021) discusses a remarkable set of facts from Balkar, a Turkic language of Russia. The discussion centres around what he terms *converb clauses*, examples of which are given below. The subject of a converb may in principle be either PRO or an overt nominal.

(32) a. Aslan$_i$ [PRO$_1$ zɨr-la zɨr-laj] šorpa ete-j e-di
A. song-PL sing-CONV soup make-CONV AUX-3SG
"Aslan was making soup while singing songs."

b. [zašciq tabaq-la keltir-e] Fatima stol-ʁa azia sal-a edi
boy plate-PL set-CONV F. table-DAT food put-CONV AUX-3SG
"Fatima was setting the table while the boy was bringing plates."
Privoznov (2021, 7a, 8a p. 48)

Our point of interest is in a process that Privoznov terms *long-distance scrambling*. Balkar allows constituents to be fronted in a rather unrestricted manner, for reasons connected to the discourse. Getting straight to the point: scrambling may take place out of converbs that are themselves adjoined to an embedded finite clause, as shown below.

(33) a.✓ [$_{\text{matrix}}$ X [$_{\text{embedded}}$ [$_{\text{converb}}$ PRO … ~~X~~ …]]]

ol zɨr-ni Fatima [Kerim$_2$ zol-da [PRO$_2$ ____ zɨr-laj] bar-a
that song-ACC F. K. road-DAT sing-CONV go-CONV
e-di] de-gen-di
AUX-3SG say-PST-3SG
"Fatima said that Kerim$_2$ was walking down the road PRO$_2$ singing that song."

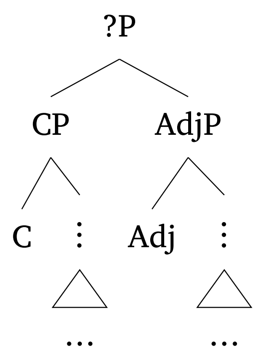
b.✓ [$_{\text{matrix}}$ X [$_{\text{embedded}}$ [$_{\text{converb}}$ DP$_{\text{subj}}$ … ~~X~~ …]]]

qart ana-sɨ-na men [[Kerim ____ boluš-a] zol-da ol alaj
old mother-3SG-DAT I K. help-CONV road-LOC 3sg thus
bar-ʁan] sun-a-ma
go-NZR think-PRS-1SG
"I think that with Kerim helping the old lady$_1$, she$_1$ was walking down the road."
Privoznov (2021, 16a, p. 52; 14d, p. 51)

The theory developed so far is able to account for (33a) in a fairly straightforward fashion: the features of elements within the control converb are able to project for the same reason that they may in English.[12] Namely, they are adjoined to vP, above the position occupied by the subject.

12. Privoznov (2021) draws somewhat different conclusions from these data about the position of control adjuncts in Balkar. To our knowledge, his theory is unable to account for the range of data discussed throughout this paper.

As for non-control converbs, we follow Privoznov (2021) in assuming that they are adjoined high in the clause, above any position occupied by the subject, as schematized below. This is reflected by the linear order of the converb clauses in the examples above: those with overt subjects appear at the left edge of the clause they modify, while those with PRO subjects appear clause medially. Privoznov (2021) discusses a number of of other diagnostics involving the scope of causative morphemes, NPI licensing, and variable binding by quantifiers, all of which suggest that converb clauses with overt subjects are adjoined above all other arguments in the clause they modify.

(34)


We have not encountered a structure of this sort so far in our discussion, and, in fact, they have unusual properties. Neither CP nor AdjP are, by the definition given in §2, indivisible feature bundles. The rules of projection, as stated before, allow the features of an element to project just when its sister is such a bundle.

(35) **Indivisible feature bundle**:
  a. a feature bundle that comes straight from the lexicon
  b. a feature bundle that has projected to a node from only one daughter

What happens in such a situation? We could imagine a number of scenarios: such structures might simply be disallowed, since they would end up bearing no syntactic features at all and thus subsequently be unable to be manipulated. What we would like to suggest is that the features either of CP or AdjP may in principle be projected in such a configuration. The choice is necessary, but arbitrary. This suggestion has empirical consequences, which are in fact borne out.

What we should expect, if this suggestion is on the right track, is that an embedded clause modified by an overt subject converb should be opaque if the converb itself is to be transparent, and vice versa. Interestingly, this appears to be on the right track. The presence of a converb modifying an embedded clause does not affect the transparency of the modified clause, regardless of whether or not the converb has an overt subject.

(36) a. zol-da$_1$ Fatima [Kerim __$_1$ [PRO ol zɨr-nɨ$_2$ zirla-j bar-a edi]
road-LOC F. K. that song-ACC
de-gen-di]
sing-CONV
"Fatima said that Kerim was walking by the road, while Kerim was singing that song."

b. üj-ge$_1$ men [[Fatima ešik-ni bezgi-ler-in-den teš-ip]
house-DAT I F. door-ACC hinge-PL-3-ABL take.off-CONV
Kerim tešek-ni ____$_1$ kijir-di] de-di-m
K. bed-ACC carry-PST.3SG say-PST-1SG
"I said that, Katima having taken the door off its hinges, Kerim carried the bed into the house." Privoznov (2021, 16c, p. 55; 18c, p. 56)

Privoznov (2021) shows that Balkar allows a process of *multiple long-distance scrambling*, where the scrambled elements need not be clausemates. An example of this is shown below: one scrambled element originates in the embedded clause, while the other originates in a control converb modifying the embedded clause.

(37) zol-da$_1$ ol zɨr-nɨ$_2$ Fatima [Kerim __$_1$ [PRO __$_2$ zirla-j bar-a edi
road-LOC that song-ACC F. K. sing-CONV go-CONV
] de-gen-di]
AUX.3SG
"Fatima said that Kerim was walking by the road, while Kerim was singing that song."

However, this process of multiple scrambling is restricted: the presence of an overt subject in the converb blocks scrambling from both the converb and clause it modifies at the same time.

(38) *ešik-ni$_1$ üj-ge$_2$ men [[Fatima __$_1$ bezgi-ler-in-den teš-ip] Kerim
door-ACC house-DAT I F. hinge-PL-3-ABL take.off-CONV K.
tešek-ni ____$_2$ kijir-di] de-di-m
bed-ACC carry-PST.3SG say-PST-1SG
"I said that, Katima having taken the door off its hinges, Kerim carried the bed into the house." Privoznov (2021, 16c, p. 55; 18c, p. 56)

In other words, we see exactly what we would expect given our theory of locality with the minor emendation suggested above. In a limited set of contexts, feature projection is obligatory but arbitrary. And in just these contexts, whether or not a domain is transparent for extraction determines whether or not its sister is opaque for extraction.

5 Discussion and conclusions

What we have seen so far is a novel theory of locality for which locality domains are determined by their local context. §2 develops a theory of feature projection that captures something like the classical CED. §3 shows that the theory is able to account for a number of exceptions to the classical CED, and furthermore explains a hitherto unexplained correlation between extraction from adjuncts and the possibility of a non-obligatory control interpretation for the adjunct in question. §4 discusses two extensions of the theory to other cases where whether or not movement out of a domain is tolerated appears to be determined by the context that domain appears in. Having motivated and developed this theory of locality, in this section we discuss a number of remaining issues, and sketch ways forward for future work.

5.1 Other approaches to the CED

Since Huang (1982), it has been common to treat specifiers and adjuncts as islands for extraction as a matter of definition. The CED, shown in (39), states that any non-complement should be opaque for extraction.

(39) *The Condition on Extraction Domains* (CED) (Huang 1982; Chomsky 1986; Cinque 1990; Manzini 1992):
Movement may not cross a barrier XP, unless XP is a complement.

The CED raises several questions: first, it is not exceptionless. We have just discussed examples of wh-extraction out of adjuncts and specifiers, both of which are clear violations of (39). These counterexamples refute the generality of (39), and suggest that we need a finer grained metric for island-hood besides the complement/non-complement distinction. Second, existing attempts to derive (39) face a conceptual disadvantage compared to the present theory.

A popular approach to the CED is to treat adjuncts and specifiers as subject to different rules than complements. For example, Uriagereka (1999), Johnson (2003), Sheehan (2013), and Privoznov (2021) suggest that non-complements must spell-out when they merge, rendering their contents inaccessible to further operations.

This approach requires some elaboration to theories of spell-out, given that complement clauses are also often proposed to spell-out at particular points in the derivation. Phases (including complement clauses) are typically assumed to be opaque to operations external to them due to their time of spell-out. However, unlike adjuncts/specifiers, phasal complements are thought to have an *escape hatch*. Elements that move to that escape hatch become accessible to later operations, despite the face that the phase has "spelled out". In order to capture the contrast between adjuncts/specifiers and complements, adjuncts/specifiers must therefore lack an escape hatch.

One could imagine several ways to encode the escape-hatch property on a phrase, such that complements have them but adjuncts/specifiers do not. For example, we could stipulate that complementation triggers spell-out of the *complement* of the phase head, while adjunction/specifier-Merge triggers spell-out of the entire phrase. Complements therefore have a specifier position which has not spelled out, while adjuncts/specifiers do not. Alternatively we could propose that edges of spelled-out phrases are always accessible, but that only certain heads have the ability to attract elements to their edge – adjuncts/specifiers routinely lack these edge features, in contrast to complements.

Both of these possibilities require us to stipulate a distinction between complements and non-complements in a way that the theory outlined in this paper does not. The present theory treats complements as first-merged elements with a head, specifiers as second-merged elements, and so on, reducing the number of primitive distinctions we need between different phrases. Moreover, the present theory is able to account for *variable* island-hood of adjuncts and specifiers, without stipulating special properties of those adjuncts and specifiers. Instead we propose that every phrase (complements and non-complements alike) is subject to the projection algorithm, which yields different results depending how many feature bundles are present on each node.

5.2 A tentatively typological view

in §3, we showed that the availability of adjunct extraction and parasitic gaps in English was tied to the position that an adjunct occupied in the clause. More specifically, we suggested that these types of gap containing adjuncts were possible only when the adjuncts were second specifiers of vP. This leads us to a relatively straightforward typological prediction: languages that disallow extraction *from* adjuncts should also disallow parasitic gaps within adjuncts. As we will see, such a picture does emerge from the data.

In a discussion of a semantic condition on extraction from adjuncts, Truswell (2011) notes that there is cross-linguistic variability in whether or not adjuncts may be extracted from that cannot be straightforwardly explained by the aforementioned condition. He notes that several languages — the *Class A* languages below — are at least as permissive as English in terms of whether or not extraction is allowed from adjuncts. Several other languages — the *Class B* languages below — are far less permissive than English, and do not allow extraction from adjuncts.

(40)

Class A	Class B
English	Dutch
Norwegian	French
Swedish	Greek
Spanish	

We are thus led to a straighforward expectation: all else being equal, Class A languages should allow parasitic gaps while Class B languages should not. It turns out that this is by and large on the right track: Engdahl's 1983 seminal article on parasitic gaps explicitly notes that the distribution of tolerable parasitic gaps in English mirrors that of Swedish and Norwegian. Campos (1991) likewise notes that Spanish appears to have parasitic gaps analogous to that of English.

The picture is markedly different for the Class B languages. Both French and Dutch allow parasitic gaps only in a small class of non-finite adjunct clauses, as discussed in Tellier (2001) and Bennis and Hoekstra (1985) respectively, and likewise generally do not tolerate extraction from the sorts of adjuncts that Class A languages do, as Truswell notes. Greek has likewise been noted to lack parasitic gaps entirely (Tsimpli 1995), and also generally does not allow extraction from adjuncts, again noted by Truswell. The facts ultimately deserve further investigation — in particular, we should like to know if extraction is possible from the non-finite clauses that allow parasitic gaps in Dutch and French — but are suggestive of the picture painted by the theory in question.

References

Abrusán, Márta. 2014. *Weak Island Semantics.* Vol. 3. OUP Oxford.

Adger, David. 2003. *Core syntax.* Oxford: Oxford University Press.

Ahn, Byron. to appear. "Mapping OUT-Argument Structure." *Syntax.*

Asudeh, Ash, and Christopher Potts. 2004. "Honorific marking: Interpreted and Interpretable." Paper presented at the Phi Workshop, McGill University.

Atlamaz, Ümit. 2019. "Agreement, Case, and Nominal Licensing." PhD diss., Rutgers University-School of Graduate Studies.

Bennis, Hans, and Teun Hoekstra. 1985. "Parasitic Gaps in Dutch." In *North East Linguistics Society,* 15:2.

Branan, Kenyon, and Michael Yoshitaka Erlewine. 2021. "Locality and Minimal Search."

Cable, Seth. 2010. *The Grammar of Q: Q-particles, Wh-Movement, and Pied-Piping.* Oxford University Press.

Campos, Héctor. 1991. "Silent Objects and Subjects in Spanish." *Current studies in Spanish linguistics,* 117–142.

Canac Marquis, Rejean. 1994. "A/A-bar Chain Uniformity." PhD diss., University of Massachusetts, Amherst.

Cardinaletti, Anna, and Michal Starke. 1994. "The Typology of Structural Deficiency: A Case Study of the Three Classes of Pronouns." *Clitics in the languages of Europe.*

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

———. 1986. *Barriers.* MIT Press (MA).

———. 1995. *The Minimalist Program.* Cambridge, MA: MIT Press.

———. 2004. "Beyond Explanatory Adequacy." In *Structures and Beyond,* edited by Adriana Belletti. Oxford University Press.

Chow, Keng Ji. 2022. "A Novel Algorithm for Minimal Search." *Snippets,* no. 42.

Cinque, Guglielmo. 1990. *Types of Ā-dependencies.* Cambridge, MA: MIT Press.

Engdahl, Elisabet. 1983. "Parasitic Gaps." *Linguistics and philosophy* 6 (1): 5–34.

Grewendorf, Günther. 1989. *Ergativity in German.* Dordrecht: Foris.

Grewendorf, Günther, and Joachim Sabel. 1999. "Scrambling in German and Japanese: Adjunction versus multiple specifiers." *Natural Language & Linguistic Theory* 17 (1): 1–65.

Hagstrom, Paul Alan. 1998. "Decomposing Questions." PhD diss., Massachusetts Institute of Technology.

Heim, Irene, and Angelika Kratzer. 1998. *Semantics in generative grammar.* Blackwell Publishing Ltd.

Huang, C.-T. J. 1982. "Logical relations in Chinese and the theory of grammar." PhD diss., MIT.

Johnson, Kyle. 2003. "Towards an etiology of adjunct islands." *Nordlyd* 31 (1).

Ke, Hezao. 2019. "The Syntax, Semantics and Processing of Agreement and Binding Grammatical Illusions." PhD diss.

Keine, Stefan. 2019. "Selective Opacity." *Linguistic Inquiry* 50 (1): 13–62.

Krivochen, Diego. 2022. "The Search for Minimal Search: A Graph-Theoretic View."

Landau, Idan. 2013. *Control in generative grammar: A research companion.* Cambridge University Press.

———. 2021. *A selectional theory of adjunct control.* MIT Press.

Larson, Richard K. 1988. *Light predicate raising.* Center for Cognitive Science, Massachusetts Institute of Technology …

Legate, Julie Anne. 2003. "Some Interface Properties of the Phase." *Linguistic Inquiry* 34(3):506–516.

Manzini, Rita. 1992. *Locality, a theory and some of its empirical consequences.* Cambridge, MA: MIT Press.

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. 2006. "Head Movement in Linguistic Theory." *Linguistic Inquiry* 37, no. 1 (January): 69–109.

McFadden, Thomas, and Sandhya Sundaresan. 2018. "Reducing pro and PRO to a Single Source." *The Linguistic Review* 35 (3): 463–518.

McGinnis, Martha. 1999. "A-Scrambling Exists!" *University of Pennsylvania Working Papers in Linguistics* 6 (1, Article 20).

Moltmann, Friederike. 1990. *Scrambling in German and the specificity effect. Ms.*

Müller, Gereon. 2010. “On Deriving CED Effects from the PIC.” *Linguistic Inquiry* 41 (1): 35–82.

Newman, Elise Sophia Bershad. 2021. “The (in) Distinction between Wh-Movement and c-Selection.” PhD diss., Massachusetts Institute of Technology.

Nicolae, Andreea, and Gregory Scontras. 2018. “How Does Who Compose?” In *Proceedings of the Eighteenth Meeting of the Austronesian Formal Linguistics Association (AFLA),* edited by Lauren Eby Clemens, Gregory Scontras, and Maria Polinsky, 125–139. Asia-Pacific Linguistics.

Nissenbaum, Jonathan W. 2000. “Investigations of Covert Phrase Movement,” Massachusetts Institute of Technology.

Pesetsky, David. 1987. “Wh-in-Situ: Movement and Unselective Binding.” *The representation of (in) definiteness* 98:98–129.

Postal, Paul Martin. 1994. “Contrasting Extraction Types.” *Journal of Linguistics* 30 (1): 159–186.

———. 1998. *Three investigations of extraction.* Vol. 29. mit Press.

Preminger, Omer. 2014. *Agreement and Its Failures.* MIT Press.

———. 2019. “What the PCC Tells Us about ‘abstract’ Agreement, Head Movement, and Locality.” *Glossa-an International Journal of Linguistics* 4 (13): 1–42.

Privoznov, Dmitry. 2021. “A Theory of Two Strong Islands.” PhD diss., Massachusetts Institute of Technology.

Rizzi, Luigi. 1990. *Relativized Minimality.* The MIT Press.

Sauerland, Uli. 2003. “Intermediate adjunction with A-movement.” *Linguistic Inquiry* 34(2):308–313.

Sheehan, Michelle. 2013. “The resuscitation of CED.” In *Proceedings of NELS,* 40:135–150.

Stanton, Juliet. 2016. “Wholesale Late Merger in Ā-movement: Evidence from Preposition Stranding.” *Linguistic Inquiry* 47 (1): 89–126.

Stowell, Timothy Angus. 1981. “Origins of Phrase Structure.” PhD diss., Massachusetts Institute of Technology.

Szabolcsi, Anna. 1997. “Strategies for Scope Taking.” In *Ways of Scope Taking,* 109–154. Springer.

Szabolcsi, Anna, and Frans Zwarts. 1993. “Weak Islands and an Algebraic Semantics for Scope Taking.” *Natural language semantics* 1 (3): 235–284.

Tellier, Christine. 2001. “On Some Distinctive Properties of Parasitic Gaps in French.” *CURRENT STUDIES IN LINGUISTICS SERIES* 35:341–368.

Truswell, Robert. 2007. “Extraction from Adjuncts and the Structure of Events.” *Lingua* 117 (8): 1355–1377.

Truswell, Robert. 2011. *Events, Phrases, and Questions.* OUP Oxford.

Tsimpli, Ianthi-Maria. 1995. “Focusing in Modern Greek.” *Discourse configurational languages,* 176–206.

Uriagereka, Juan. 1999. “Multiple spell-out.” *Current Studies in Linguistics Series* 32:251–282.

Williams, Edwin. 1992. “Adjunct control.” In *Control and grammar,* edited by Richard Larson, Sabine Iatridou, Utpal Lahiri, and James Higginbotham, 297–322. Springer.

Yoshida, Mitsunobu. 2001. “Scrambling in German and Japanese from a minimalist point of view.” *Linguistic Analysis* 30:93–116.

A Weak islands

The theory discussed so far is able to account for, among other things, the fact that *wh*-movement from certain adjuncts is in principle possible. Interestingly, however, not all types of *wh*-movement are possible: movement of an adjunct is considerably degraded when compared with movement of an argument. The answers in (41) are provided to force parses where both gaps originate within the control adjunct.

(41) **A weak island effect**
a. What did the flower open [in order to attract ____]
→ A: passing pollinators
b. *How did the flower open [in order to attract pollinators ____]
→ A: with a particular UV pattern

(41) is puzzling. The answer to this puzzle, in part, depends on whether or not we want our theory to be a general theory of weak islands. As the facts below suggest, weak islandhood is not straighforwardly connected to being a non-specifier: (42a) shows that complement clauses may be weak islands, while (42b) shows that the complement of a Neg head is comparably a weak island.

(42) **Complement weak islands**
a. *How do you regret that [John fixed the car ____]?
b. *How didn’t [Mary arrive at the party ____]?

For the theory at hand, we can state the weak island property as something like the following, stated below. This description is vague enough to allow for either a syntactic (see Pesetsky 1987; Cinque 1990; Rizzi 1990) or a semantic (see Szabolcsi and Zwarts 1993; Szabolcsi 1997; Abrusán 2014, a.o) approach to weak island-hood.

(43) **Weak islandhood**
Operations making reference to a path defined by [~~•wh•~~] are barred from certain domains.

For us, given the generalization about weak islands above, the puzzle is why (41a) is acceptable. We sketch here a theory that will allow (41a), making use of certain assumptions about *wh*-movement first developed in §3.2, and having much in common with antecedent proposals about escape from weak islands originating in Rizzi (1990).

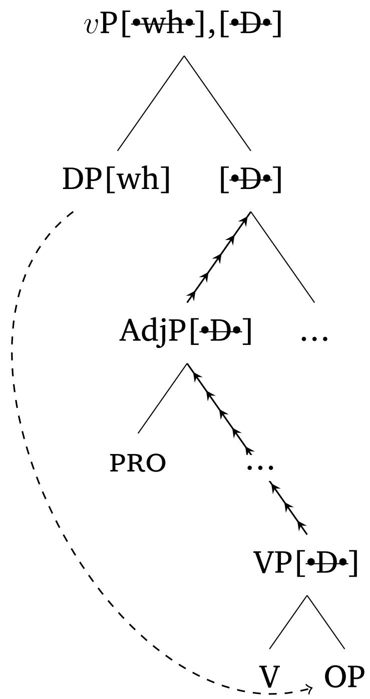
The core idea is that *wh*-movement out of weak islands is contingent on a feature [~~•D•~~] instead of a feature [~~•wh•~~]. Recalling our discussion of [wh] feature projection from §3.2, we saw that projection of a [~~•wh•~~] checked by the object was contingent on movement to an intermediate position motivated by checking of [•D•].

Wh-movement of objects, then, must involve two well-formed paths: one between the *wh*-phrase and its base position, defined by the feature [~~•D•~~], and another between the *wh*-phrase and the final landing site, defined by the feature [~~•wh•~~]. This, notably, contrasts with *wh*-movement of adjuncts, which do not occupy a position where they check [•D•].

Much work on *wh*-movement, at least in English, suggests that movement of *wh*-arguments may involve either a "true" movement dependency, or binding between the moved *wh*-element and something like a null pronominal (see Pesetsky (1987), Rizzi (1990), Postal (1994), and Stanton (2016) for discussion along these lines). Notably, the same is not true for movement of *wh*-adjuncts, where the binding strategy is not generally available. We suggest that the binding strategy — which only arguments may make use of — involves Search through a domain bearing [~~•D•~~], rather than [~~•wh•~~], and that it is this distinction which allows *wh*-movement of arguments to avoid being blocked by (43).

Consider the structure below, where AdjP is taken to be a weak island as in (41), and thus subject to (43). Following Rizzi (1990) and Postal (1994), a null element — represented here by OP — may in principle be merged in a position where it checks [•D•], provided it is subsequently bound by a *wh*-phrase of some sort. Binding requires a path of [•D•] features between the binder and bindee, similar to the binding of PRO and null operators discussed earlier in this paper. As we see below, the *wh*-phrase may in principle be generated in spec,vP of the matrix clause, so long as the adjunct containing OP appears in a position from which it may project its features. The *wh*-phrase may bind OP from this position, via the path of [•D•] features between the two. The *wh*-phrase is also Merged in a position where it checks a [•wh•] feature in the matrix clause, creating a path of [~~•wh•~~] to matrix spec,CP that does not traverse a weak island.

(44)



In contrast, such a derivation is not available for adjunct *wh*-phrases such as *how*, as in the case of (41b). Such an adjunct could be merged in spec,vP of the matrix clause, but it would be unable to bind a comparable OP in the adjunct clause. Such an adjunct could also be merged in spec,vP of the adjunct clause, but subsequent movement of the adjunct would run afoul of (43).