Optimal Linearization: Word Order Typology with Violable Constraints

Abstract. Insofar as linearization is a post-syntactic phenomenon pertaining to phonological forms, it is desirable that it be modelled in a violable constraint framework parallel to other PF-branch phenomena. This paper presents such a model, with the goal of capturing the Final-Over-Final Constraint (FOFC), a typological generalization that head-final phrases embed only other head-final phrases while head-initial phrases may embed phrases of either headedness. The specific model proposed here, termed Optimal Linearization, is couched in Optimality Theory and comprises a set of three interacting violable constraints for mapping syntactic structure to linear order. One constraint, ANTISYMMETRY, closely mimics the action of Kayne's familiar Linear Correspondence Axiom. An opposing constraint HEADFINALITY penalizes deviations from an idealized head-final order. Lastly, a domain-specific constraint HEADFINALITY-α enforces head-final ordering only within one constituent, allowing for FOFC-respecting disharmonic orders. Optimal Linearization has several advantages beyond typology, including offering insight into the leftward direction of specifiers as the emergence of an unmarked preference for head-finality. Linearizing at PF also allows us to model cases where phonological or prosodic properties of words affect their order; this is illustrated with the example of Irish pronoun postposing.

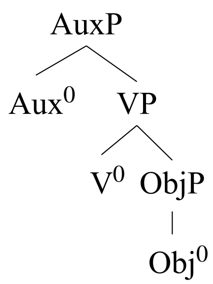
1. Introduction

Even with all the advances in syntactic theory over the last few decades, the relationship between syntactic structure and linear order remains somewhat mysterious. The Headedness Parameter, which states that all heads are either on the left or the right of their phrase, has two significant flaws: First, recent work has made it clear that the specifier position, insofar as it can be coherently defined on purely syntactic grounds, is always linearized to the left of its head (e.g. Kayne 1994a; Abels & Neeleman 2012:a.o), but the Headedness Parameter offers no explanation as to why this should be the case; we must instead simply stipulate it. Put another way: Head-final languages are fully head-final in that the head follows both complement and specifiers, but in head-initial languages the head precedes only the complement, not the specifier. The Headedness Parameter offers no explanation for this striking asymmetry.

The second failure of the Headedness Parameter is an empirical one: Depending on one's assumptions, the model either wildly undergenerates or wildly overgenerates. The undergeneration case is commonly known: If we assume that the headedness parameter can't be set for individual heads, we predict that all phrases in a language will have identical headedness. This is an easily falsifiable prediction. German is a frequently-studied example of a language with mixed-headedness; casting our net a bit more broadly, WALS (Dryer & Haspelmath 2013) lists 66 languages in which the relative ordering of the verb and its object differs from the ordering of adposition and noun. This is a small percentage of the sample, to be sure, but it represents only one of the ways that a language might display mixed-headedness; whatever model we use, it clearly must rule in these mixed cases. On the other hand, if we allow languages to set the Headedness Parameter differently for each individual phrase type, we miss the very real typological gaps in headedness relations between phrases. The best known of these is the Final-Over-Final Constraint (Sheehan et al. 2017): If a phrase is head-final, then its complement will be as well; if a phrase is head-initial, its complement may have either headedness. This is illustrated with a schematic tree in (1); any part of a tree with the same geometry will have the same word order prediction. If we are allowed to set the Headedness Parameter individually for each phrase, we predict 4 possible orderings for this tree; empirically, though, the Final-Over-Final Constraint (FOFC) rules out the order in which VP is head-initial but AuxP is head-final:

(1) **The Final-Over-Final Constraint:** The complement of a head-final phrase is also head-final.

(2) a.



b.

	Aux Initial	**Aux Final**
V Initial	Aux V Obj (e.g. English)	* V Obj Aux
V Final	Aux Obj V (e.g. Bambara)	Obj V Aux (e.g. Hindi)

The FOFC has been extensively discussed in the literature, notably in a recent book by Sheehan, Biberauer, Roberts, & Holmberg (2017); evidence for the constraint is presented there and in the references contained therein. I will present a small sample of the evidence here, however, coming from WALS (Dryer & Haspelmath 2013). WALS does not code directly for the kind of disharmonic orders that interest us here, but it does include a proxy: Feature 94A covers the placement of "adverbial subordinators", a subset of complementizers, with respect to their embedded clause; we can take this as tracking the order of C and its complement S. We can then look at the relationship between these embedding complementizers and the headedness of the language overall (as measured by Featurer 95A, "Relatioinship between the order of Object and Verb and the order of Adposition and Noun Phrase"). The results are tabulated in (3).

(3) **The FOFC in WALS**:

	C S	**S C**
Head-Initial	258 (87%)	1 (0.001%)
Head-Final	37 (13%)	85 (99.99%)

As can be seen, languages in which a head-final C embeds an otherwise head-initial clause are vanishingly rare[1], with only one such language listed in WALS.[2] This provides evidence for only a small subset of the range of cases covered by the FOFC, and the reader is directed to the existing literature for exemplification of the other cases. Nonetheless, it can be seen that the FOFC is at least a very strong trend and likely a universal.[3]

The FOFC not only rules out setting the Headedness Parameter individually by phrase, but in fact rules the middle-ground option as well: If we are only allowed to set the Headedness Parameter separately for certain domains (for example, for the vP phase and the CP phase), we still have no explanation for why there should be an asymmetry between head-final phrases and head-initial phrases in terms of what they embed. Something stronger than the Headedness Parameter is needed.

More contemporary approaches to linearization fair no better. Most of these take Kayne's Linear Correspondence Axiom as a starting point and map asymmetric c-command to linear precedence (see e.g. Fox & Pesetsky 2005). This has the benefit of explaining why specifiers are always on the left — they do always c-command the head — but at the cost of ruling out head-final phrases entirely. Analyses based on the LCA typically resort to hypothesizing complex movement in the syntax with little independent support. Leaving aside the stipulative nature of such an account, it is difficult to see how it could explain the FOFC: If surface-head-final phrases are generated by moving a complement higher in the structure, why shouldn't we be able to move a surface-head-initial complement in such a way as to violate the FOFC?[4]

In this paper, I will propose a model of linearization which provides some explanation for the two issues discussed above, namely: Why are specifiers always on the left? And, why are FOFC-violating orders virtually unattested? I start from the perspective that linearization is a PF phenomenon (Kayne 1994b; Chomsky 1995) and should be modelled the same way we model other phonological processes, namely with violable constraints. This accords with a growing body of

[1] Difference of proportions: $\chi^2 = 227.8$, df = 1, $p < 0.0001$.

[2] The one language listed is Buduma (Lukas & Nachtigal 1939), a Chadic language.

[3] The low percentage of C S languages which are head final in this data (13%) is a sampling artifact — head-final languages are under-represented in Feature 94A generally. Note that the disharmonic case does comprise 30% of the head-final languages in this sample.

[4] In fact, Abels & Neeleman (2012) demonstrate that, in the absence of restrictions on what movements are possible, the LCA is too permissive even to account for many of the generalizations which originally motivated it.

evidence that phonological or prosodic factors sometimes play a role in determining word order; for example, Bennett et al. (2016) show that prosodically-light pronouns in Irish may be postposed, while Serbo-Croatian 2nd position clitics have long been argued to be positioned relative to a prosodic, rather than syntactic, constituent (e.g. Halpern 1992; Bošković 2001:and many others). I will join López (2009); Elfner (2012), and others in proposing that we can model these cases of PF displacement with Optimality Theory (Prince & Smolensky 1993/2004) by having constraints on linearization come into competition with prosodic markedness constraints. In contrast to these earlier models, however, I propose that the mapping from syntactic structures to linear strings occurs fully post-syntactically: Rather than proposing a single "word order faithfulness" constraint penalizing deviance from a pre-specified order, I propose a family of constraints which enforce certain relationships between syntactic structure and word order, working together to derive the correct output. Modelling linearization in this way has the benefit of making clear, well-defined typological predictions, in the form of factorial typology: Different rankings of constraints should predict all and only the classes of word order actually observed.

I will call this general approach Optimal Linearization, and will demonstrate that, given the right constraint set, we can predict the typological gap described as the FOFC while still offering a coherent explanation for why specifiers are always left.[5] My proposed constraint set models word order typology as arising from the competition of two core constraints: One, HEADFINALITY, encodes a general preference for heads and their non-maximal projections to follow their sisters. The other, ANTISYMMETRY, encodes a competing preference for syntactic objects higher in the tree to be linearized earlier in the string; it closely mimics the effect of the familiar LCA (Kayne 1994b). These are both violable constraints; in some cases satisfaction of one constraint will entail violation of the other. Competition of these two constraints will derive the two harmonic word orders (head-initial and -final). Within this framework, the leftward position of specifiers occurs not because the specifier c-commands the head, but rather because the terminals within the specifier *fail* to c-command it; specifiers are therefore placed on the left as the grammar tries to achieve the

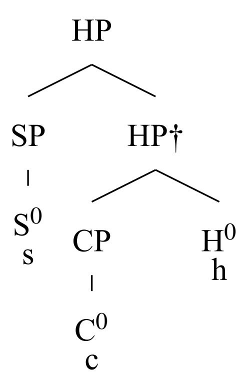
[5] In particular, I aim to capture the ordering of specifiers and complements; I will not take up the positioning of adjuncts here. See section 6 for thoughts on how this system might be extended to address the ordering of adjuncts.

"most head-final" ordering possible with heads still preceding their complement. Finally, a third constraint HEADFINALITY-α is identical to HEADFINALITY except that it considers only the order of those heads dominated by some node α. The addition of this constraint allows us to derive exactly those disharmonic orders compatible with the FOFC. Finally, I'll show in section 5 that Optimal Linearization is capable of accounting for those cases where phonological or prosodic factors seem to play a role in the determination of word order, and at least in the Irish case discussed by Bennett et al. (2016) fairs better than the previous model.

2. Harmonic word orders

I'll introduce Optimal Linearization by illustrating how it models a subset of the complete typology. In particular, I will start by considering only the "harmonic" word orders — those word orders that are consistently head-initial or head-final in all phrases. Intuitively, we want the Optimal Linearization procedure to take a syntactic structure like (4a) and produce one of the two orders in (4b) (and no others). (The nodes have been named corresponding to their structural position — so the specifier is SP, the head is HP, and the complement is CP.)

(4) a.



b. **Head-Initial:** shc

c. **Head-Final:** sch

In the Headedness Parameter model, these two orders are controlled by a single parameter. In a violable-constraint framework, it's more natural to have them controlled by two constraints: When one constraint (call it HEADFINALITY) is dominant, the output will be the head-final order *sch*; when the other constraint (call it ANTISYMMETRY) is dominant, the output will be the head-initial order *shc*. Further, we want this to extend to all phrases — that is, if there is more material in SP or CP, we want those phrases to be linearized the same way as HP. The goal of this section will be to define the constraints HEADFINALITY and ANTISYMMETRY to achieve exactly this result.

Before getting to the constraints themselves, however, I first need to introduce the rest of the Optimal Linearization model.

2.1 Some housekeeping

Before getting into the constraints themselves, it's worth taking a second to formalize what exactly the complete model looks like.[6] The general architecture of OT involves two core components: GEN takes an *input* and generates from it a number of *candidates* (i.e. potential outputs); EVAL takes the input and candidate set and, using a set of ranked violable constraints, selects a winner, which is the output of the model overall. Any given language is taken to have a fixed ranking of constraints. Taken together, GEN, EVAL, and the ranked constraints are a function from the possible inputs in the language to the possible outputs.

In Optimal Linearization, the input to GEN is the output of the narrow syntax, i.e. a phrase marker produced by some particular theory of syntax. While Optimal Linearization is compatible with a variety of syntactic theories, I will use structures compatible with Merge-based derivations and the Minimalist Program generally (Chomsky 1995). I will assume that the candidates created by GEN are strings composed of whatever phonologically-contentful Vocabulary Items are produced by the Spell-Out of the set of syntactic terminals in the input. I'll refer to these vocabulary items generically as "words". The set of candidates produced by GEN will be the full set of possible orders of words, so if there are n syntactic terminals mapped to phonologically-contentful words, there are $n! = n(n-1)(n-2)...$ candidates from which a single unique winner will be selected. Phonologically null syntactic terminals remain part of the input to the linearization component, but are never present in any of the candidates.

As a matter of notational convention, I will use capital letters to denote syntactic terminals (A,

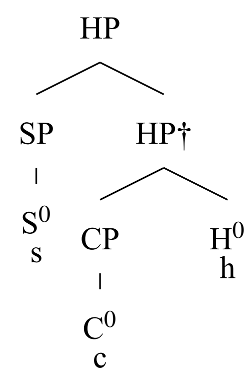
[6] While I endeavor to introduce the formal mechanisms of OT in this text, readers unfamiliar with the framework are referred to McCarthy 2002 for a more complete introduction.

B) and lower case letters to refer to the words corresponding to them (a, b). In addition, I will reserve the letters {X, Y, Z} for variables ranging over syntactic labels; letters from the beginning of the alphabet denote specific syntactic objects. The symbol $<$ denotes string precedence, so $x < y$ means some word x precedes some word y. As a last notational convention, I will draw all syntactic trees in a head-final fashion; remember, however, that syntactic trees have no order!

2.2 HEADFINALITY

Having dispensed with the preliminaries, let's now turn to the derivation of head-final orders. This will be accomplished by a constraint HEADFINALITY which, given the input (5a), prefers the order in (5b) to all other possible orders (5c).[7]

(5)



a. $\rightarrow$ sch

b. *shc, *csh, *chs, ...

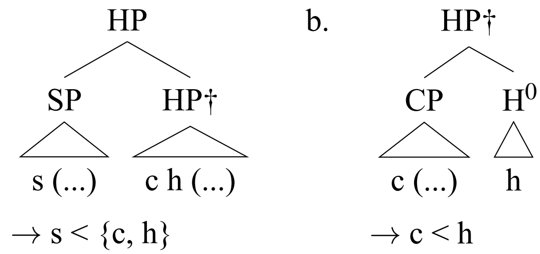
Let's think about what properties the winning order *sch* has that the other possible orders don't. First, it orders the specifier *s* before everything that isn't the specifier; any order that doesn't have *s* initial will be dispreferred. Put another way, the correct output has HP† following its sister. Second, the correct output orders the complement c before the head h; any order that has $h < c$ will be dispreferred. Put another way, H^0 follows its sister.

By visualizing each branching node separately, it can be seen that these two ordering conditions share a structural description. One ordering relation relates the daughters of HP to each other; the other relates the daughters of HP† to each other. In each case, the daughter that shares a label with

[7]Optimal Linearization requires that we be able to distinguish phrasal nodes from non phrasal nodes, but "bar-levels" have no special status in this theory; as such, I've labelled all non-terminal nodes as "XP", here and in all other trees. However, for expositional reasons it will be convenient to have unique labels for each node; accordingly, I've marked the phrasal, non-maximal nodes with †.

the node in question (HP† for HP; H for HP†) is set to follow the daughter that doesn't (SP for HP; CP for HP†).

(6) a. b.



It's going to be useful to have a pair of terms that distinguish these two structural relations. I'm going to call the daughter that shares a label with its parent the **descendant** or **endogenous daughter**; the one that doesn't share a label with it I'll call the **in-law** or **exogenous daughter**. Put another way, when two nodes undergo Merge, the one which projects becomes the descendant and the one that doesn't becomes the in-law. Specifiers and complements will always be in-laws of the nodes immediately dominating them; heads and their non-maximal projections will always be descendants.

Intuitively, then, HEADFINALITY is a constraint that prefers orders in which, for every branching node, the material dominated by its in-law precedes all material dominated by its descendant. Optimality Theory constraints are generally stated in terms of the output configurations they disprefer, i.e. the configurations which incur violations of the constraint. Putting HEADFINALITY into that form:

(7) HEADFINALITY : Assign one violation for each branching node XP dominating a pair of terminal nodes X^0 & Y^0 such that:

a. Y^0 is dominated by the in-law of XP;

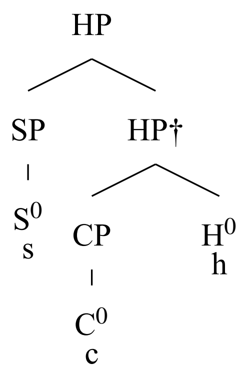
b. X^0 is not dominated by the in-law of XP; and[8]

c. $x < y$.

[8] If X^0 is dominated by XP but not dominated by the in-law of XP, then it is by definition dominated by the descendant of XP. Once we turn to linearizing movement structures in section 3, we will encounter cases in which a particular head is dominated by both the in-law and the descendent of XP; defining the constraint as shown here will prevent it from giving contradictory orders in these cases.

I'll illustrate the action of this constraint in an OT tableau. The candidate orders are listed in the leftmost column; the next column lists which branching nodes incur violations of HEADFINALITY. In this input, there are only two branching nodes and so the constraint scores a maximum of two violations. The arrow indicates the winning candidate *sch*, the only candidate which scores no violations.

(8) a.



b.

	(a)	HEADFINALITY
	shc	*H†
→	sch	
	csh	*HP
	chs	*HP
	hcs	*HP *HP†
	hsc	*HP *HP†

While this is a simple example, it serves to illustrate the action of HEADFINALITY generally. The constraint will linearize any XP in the same fashion as HP in this example — with everything contained in the specifier foremost, and X^0 final.

2.3 ANTISYMMETRY

The constraint HEADFINALITY suffices for deriving harmonically head-final word orders. In order to derive the head-initial orders we need a constraint that opposes HEADFINALITY. That is, we want some constraint ANTISYMMETRY such that the same tree in (8) is mapped to the order *shc* whenever ANTISYMMETRY$\gg$ HEADFINALITY. It may at first seem tempting to make ANTISYMMETRY the inverse of HEADFINALITY — that is, have it require the descendant to proceed the in-law. However, this won't work, as head-initial orders and head-final ones are not symmetric: In both orders, the specifier must precede everything that follows it. We need to look for something else that will create head-initial orders than just the reverse of HEADFINALITY.

I propose that we take a (metaphorical) page from Kayne's (literal) book (1994) and make ANTISYMMETRY a constraint that enforces correspondence between asymmetric c-command

and precedence. Unlike Kayne, however, I will only consider relationships between terminal nodes. This frees us from making stipulations about segments & categories, and will also have some other benefits that I will make clear momentarily. Intuitively, then, the constraint that we're looking for is one that penalizes words that occur in the opposite order as the asymmetric c-command relation between their terminals. More formally:

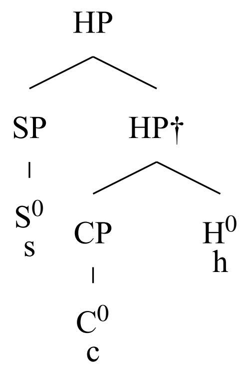
(9) ANTISYMMETRY: Assign one violation for each pair of terminal nodes X^0 & Y^0, where:

a. X^0 asymmetrically c-commands Y^0; and

b. $y < x$.

This constraint ranges over pairs of nodes that stand in an asymmetric c-command relation. In the basic spec-head-comp structure we've been investigating so far, there is only one such pair: The head H^0 asymmetrically c-commands everything in CP (namely C^0). As such, ANTISYMMETRY will score a maximum of one violation whenever $c < h$. However, ANTISYMMETRY will not order the specifier S^0 with respect to either of the other heads — while the phrase SP asymmetrically c-commands both h and c, S^0 itself does not. How, then, will the system order the specifier? Conveniently, we already have a constraint which accomplishes this: HEADFINALITY requires that HP be linearized such that everything in SP precedes everything in HP†. In a violable constraint system like OT, low-ranked constraints remain active even when dominated by a higher ranked constraint; even when ANTISYMMETRY$\gg$ HEADFINALITY, then, HEADFINALITY is still active and can enforce the leftward position of the specifier. I've presented this in tableau form below. ANTISYMMETRY eliminates the three candidates in which $c < h$; of the three that remain, only one fails to incur a violation of HEADFINALITY for HP, namely the one that orders the specifier on the left.

(10) a.



b.

	(a)	ANTISYMMETRY	HEADFINALITY
→	shc		*HP†
	sch	$*h < c$	
	csh	$*h < c$	*HP
	chs	$*h < c$	*HP
	hcs		*HP *HP†
	hsc		*HP *HP†

This is a case of "the emergence of the unmarked" (McCarthy & Prince 1994): The lower-ranked constraint acts to select the winner exactly when the higher-ranked one fails to choose. In this case, the higher-ranked ANTISYMMETRY doesn't select between the different placements of the specifier *s* within the string — it only requires that the head precede its complement. The fact that the specifier is on the left in the winning candidate is a reflection of the system choosing the "most head-final" order among those compatible with the order $h < c$. Optimal Linearization thus gives us new insight into a previously-mysterious fact about word order typology, namely that specifiers are always left-most even in otherwise "head-initial" languages. Put another way, it has always been somewhat problematic that so-called head-initial languages are never fully head-initial, but rather always require specifiers to precede the head. Optimal Linearization lets us understand this fact as a preference for head-finality emerging even in otherwise head-initial languages.

So far we've considered only a single, abstract tree where the specifier and the complement contain only a single word. Hopefully it is clear that adding more words to either specifier or complement will behave in the expected way: HEADFINALITY will provide pressure to linearize all the specifier material before head & complement and also all the complement material before the head; ANTISYMMETRY, likewise, will provide pressure to linearize the head before all the complement material — the head, after all, does asymmetrically c-command all of its complement. The same general pattern of linearization will be replicated within each XP, just as we'd expect. There is one class of syntactic structure not yet accounted for, however, namely structures involving movement. This is what I'll turn to in the next section.

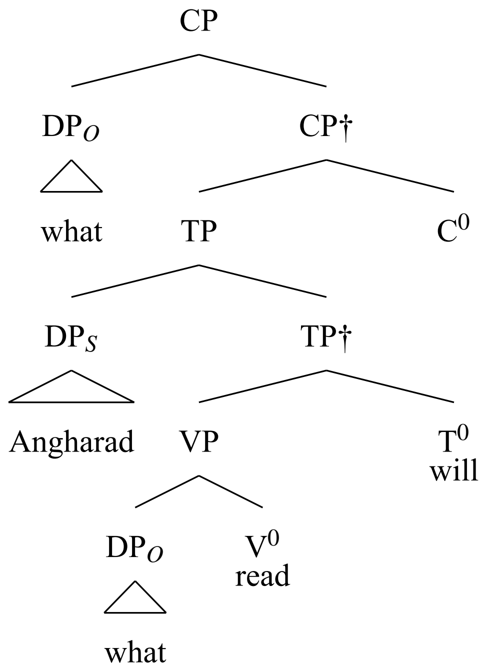
3. Linearizing movement

One of the goals of any linearization algorithm must be to explain why moved items appear in the location that they do (and only that one). That is: Once an XP has moved, what prevents it from being linearized according to its base position? And what prevents it from being spelled out twice, once according to each position? In most traditional theories of linearization there is an operation of "copy-deletion" which applies before linearization and transforms the tree at PF such that moved items are only in one position. However, Johnson (2016) outlines some possible undesirable consequences of introducing this extra transformation between the syntax and the linearization. Instead, I propose to keep to the original intuition that it is linearization itself that forces moved items to be spelled out in a particular location. The input to Optimal Linearization, then, will still have moved items in all of their positions. I will assume for the moment that GEN only creates candidates that have exactly one word for each (phonologically-contentful) syntactic terminal, even if that terminal has multiple copies — that is, GEN can't distinguish individual copies of a moved item. This prevents moved items from being linearized in multiple positions (a.k.a. multiple spellout). This may or may not be a desirable assumption, as multiple spellout of movement chains has been proposed as an analysis of resumption (Sichel 2014:e.g.) and verb-doubling predicate clefts (Koopman 1984; Kandybowicz 2006; Cable 2004:e.g.). If we want to capture these phenomena using multiple spellout, we would need to relax this restriction on GEN but then add additional constraints to enforce single spellout in all but the relevant contexts. Such a project is beyond the scope of this paper, so for the moment I'll use the constrained version of GEN.

With that in mind, let's consider what we want the Optimal Linearization constraints to do in the case of movement structures. I'll use English *wh*-movement as an illustrative example; (11) presents a simplified structure for an object *wh*-question.[9]

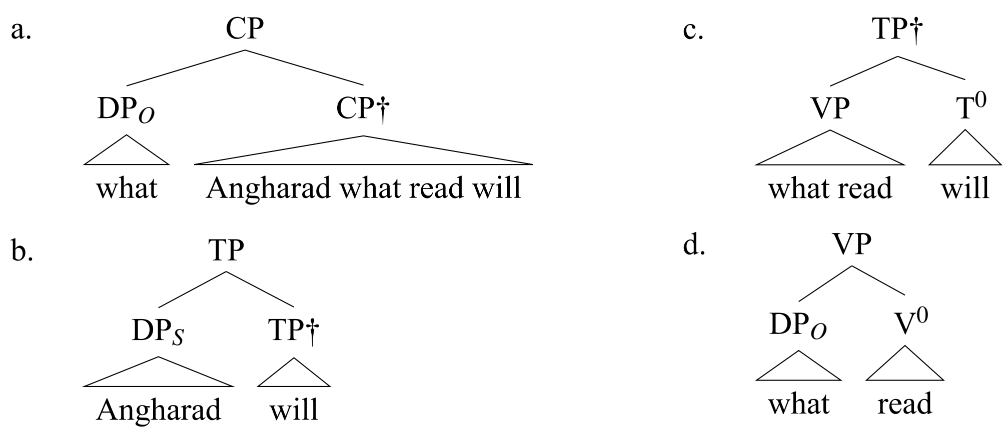
[9] More specifically, this is an embedded question. For expository reasons, I've omitted subject-auxiliary inversion and have shown the subject being base-generated in spec,TP.

(11)



Let's first consider how we want HEADFINALITY to treat the moved item. Recall that HEADFINALITY scores violations based on branching nodes. There are 5 branching nodes in (11), but one of them (CP†) has a branch with no phonologically-contentful words (C^0) and so will never score a violation. The remaining 4 branching nodes are as follows:

(12)



At once we can see that there's a problem. HEADFINALITY will score a violation for any branching node for which material in its descendant precedes material in its in-law. (12a) shows that the constraint will score a violation for CP if *Angharad* (which is in the descendant CP†) precedes *what* (which is in the in-law DP$_O$). (12c), however, shows that the constraint will score a violation for TP whenever *what* (which is in the descendant TP†) precedes *Angharad* (which is in the in-law DP_S). This produces produces a contradictory ordering for this tree.

Of course, the problem is that the constraint as defined can't distinguish between the 'high' and 'low' positions of the moved item. We want *what* to be linearized according to its higher position[10] , namely spec,CP. In other words, we want the constraint HEADFINALITY to consider DP_O only when it is evaluating the node CP; the contents of DP_O should not be relevant for the linearization of any lower branching node. In order to accomplish this, I will borrow from Abels (2003) the idea of *total domination*. Intuitively, some node X dominates a node Y only if it dominates all copies of Y. Formally:

(13) X **totally dominates** Y iff all copies of Y are dominated by a copy of X.

In (11), DP_O is totally dominated by only two items: itself (total domination is reflexive) and CP. All of the other terminal nodes are totally dominated by everything which (non-totally) dominates them — in the absence of movement, domination and total domination are identical. This allows us to revise our definition of HEADFINALITY to linearize the moved item according to its highest position:

(14) HEADFINALITY (revised): Assign one violation for each branching node XP **totally dominating** a pair of terminal nodes X^0 & Y^0 such that:

a. Y^0 is dominated by the in-law of XP;

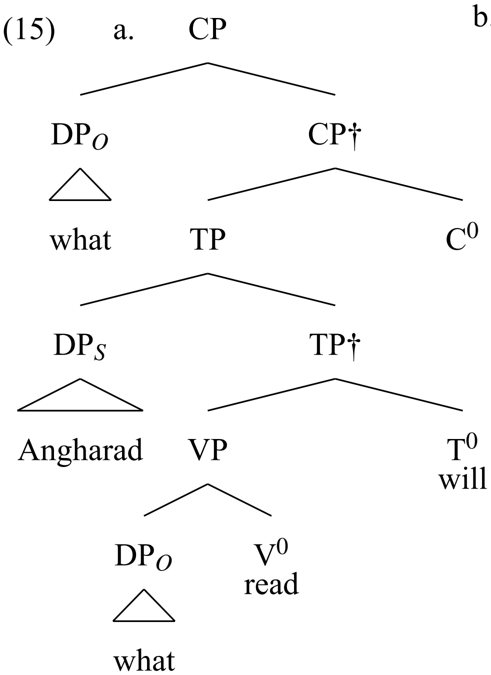
b. X^0 is not dominated by the in-law of XP; and

c. $x < y$.

[10] This may not always be true if for instance *wh-in situ* languages covertly raise the *wh* item (Watanabe 1992; Cole & Hermon 1998:e.g.) — in covert movement in general it seems that the linearization scheme must pick the lower copy. Fully accounting for these facts is beyond the scope of this paper, but we might propose that for instance there are two versions of each of the Optimal Linearization constraints, one which sees the lower copy and one the higher; the ranking of these versions relative to each other would determine whether movement overt or covert. Further refinement would be needed to ensure that overt and covert movement could coexist in the same language.

Because CP is the only branching node which totally dominates *what* in (11), the only way for *what* to violate HEADFINALITY is for it to follow anything contained in CP but not in DP_O, i.e. any word in CP†other than itself. As such, *what* (and in fact all of DP_O, if it were larger) will be linearized leftmost, in accordance with its moved position. I've illustrated this in the tableau in (15); space does not permit me to include all 24 candidate orders, so I've chosen a representative set. The winning candidate is a fully head-final pseudo-English.[11]

(15) a.



b.

	(a)	HEADFINALITY
→	what Angharad read will	
	Angharad what read will	*CP
	what Angharad will read	*TP†
	Angharad will read what	*CP *TP†

Of course, to achieve the correct head-initial order for English we need to consider ANTISYMMETRY. Here, we face a similar problem: V^0 still asymmetrically c-commands everything (non-reflexively) dominated by DP_O, and so ANTISYMMETRY will exert pressure for *read* < *what* as though *wh*-movement had never occurred. Again, what we want is a notion of *total c-command* parallel Abels (2003): V^0 fails to c-command *what* in all of its positions, and therefore won't be ordered before it. Total c-command is easy to formalize:

[11] Here we see the relevance of defining HEADFINALITY such that the material in the in-law must precede the material 'not in the in-law' (as opposed to 'in the descendent'), as mentioned in fn. 8: *what* is contained in both CP's in-law and descendent. If the constraint were defined in terms of the descendent, it produce the nonsensical ordering of *what* > *what*. The problem gets worse if the moved item has multiple words, for example if DP_O were *which book*: Here the constraint would both require *which* > *book* (since *which* is in the in-law and *book* is in the descendent) and *book* > *which* (since the reverse is also true).

(16) a. X **totally c-commands** Y iff:

(i) X does not dominate Y; and

(ii) everything that totally dominates X also totally dominates Y.

b. X **asymmetrically totally c-commands** Y iff X totally c-commands Y and Y does not totally c-command X.

In (11), V^0 does not totally c-command DP_O: for one, V^0's immediate mother VP does not totally dominate DP_O. In fact, there is nothing that totally c-commands the moved item. All that remains, then, is to update our definition of ANTISYMMETRY to use total c-command:

(17) ANTISYMMETRY(revised): Assign one violation for each pair of terminal nodes X^0 & Y^0, where:

a. X^0 asymmetrically **totally c-commands** Y^0; and

b. $y < x$.

Again, I've illustrated the action of this constraint in a tableau; as before, it fails to order any specifier, but HEADFINALITY emerges to accomplish that.

(18)

	(a)	ANTISYMMETRY	HEADFINALITY
	what Angharad read will	*$will < read$	
	Angharad what read will	*$will < read$	*CP
→	what Angharad will read		*TP†
	Angharad will read what		*CP *TP†

With this last modification to the constraints, Optimal Linearization will now linearize all moved phrases according to their highest position.[12]

[12] A reviewer asks to what extent the winning candidate is affected by details of the syntactic analysis, in particular by the addition or subtraction of functional material; for example, in (11) I have omitted vP; how would the linearization change if it were included? If the additional material is phonologically contentful, then the resulting candidates will be different and no direct comparison is possible; on the other hand, if the additional material is phonologically null, it will have no affect on the linearization whatsoever: Because only contentful words are present in the output candidates

4. Disharmonic word orders

Up to this point, I've restricted my attention to only the two harmonic word orders. There is a third order compatible with the Final-Over-Final Constraint: A head-initial phrase can embed a head-final one (but not the reverse). For example, German embedded clauses have a head-initial complementizer but are otherwise head-final[13] (19a); for an example lower in the clause, verbal auxiliaries in many of the Mande languages Kastenholz (2003) precede the VP, while the verb itself follows its complement (19b).

(19) a. German:

... dass Fritz mich gesehen hat.
that Fritz me seen has

"...that Fritz has seen me."

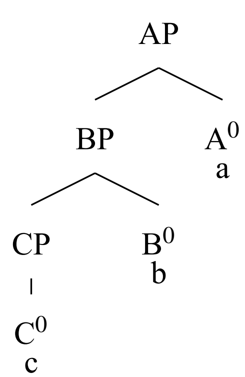
b. Bambara: (Dumestre 1984, via Kastenholz 2003)

fíɲɛ́ yé tásuma jíidi
wind PERFECT fire increase

"The wind made the fire flare up."

Abstractly, the FOFC-compliant disharmonic order follows the schema in (20): AP is linearized in a head-initial fashion, while BP is head-final.

(20) a.



b. **Disharmonic order:** acb

(by assumption), no violations will ever be scored involving a node dominating no contentful material. In essence, linearization operates on a "flattened" structure with null heads (and their immediate projections) are removed; this is reminiscent of the way the MATCH constraints as defined in Elfner (2012); Bennett et al. (2016) flatten syntactic structure to prosodic structure.

[13] Under the most common analyses of V2, matrix clauses are also an example of a mixed-headed order; I'll stick to embedded clauses here in order to avoid the complexities of head movement.

At present, the Optimal Linearization constraint set includes just two constraints, giving a maximum of two rankings / language classes. In order to allow for the disharmonic order, we'll need to add an additional constraint. I propose that this constraint is a relativized version of HEADFINALITY which only considers those nodes (reflexively) dominated by some node α. For example, in (20), α is BP; the constraint would score a violation for BP (which does reflexively dominate itself) if $b < c$, but would not consider the ordering of a at all. This leaves ANTISYMMETRY free to order AP head-initially.

This constraint captures the core generalization of the FOFC: head-finality "propagates down" the tree such that any node dominated by a head-final node will also be head-final itself. Formally, HEADFINALITY-α is defined nearly identically to HEADFINALITY except for a clause specifying its domain of application:

(21) HEADFINALITY-α : Assign one violation for each branching node XP **dominated by** α and totally dominating a pair of terminal nodes X^0 & Y^0 such that:

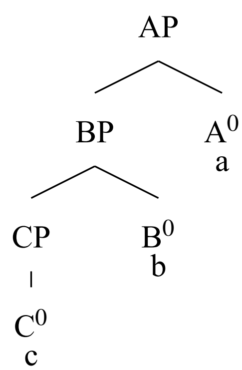
a. Y^0 is dominated by the in-law of XP;

b. X^0 is not dominated by the in-law of XP; and

c. $x < y$.

HEADFINALITY-α and HEADFINALITY are in a subset ("stringency") relationship: HEADFINALITY-α will always assign a strict subset of the violations assigned by HEADFINALITY. In practical terms, this means that whenever they are ranked "together" (i.e. both above or both below ANTISYMMETRY), their effects will be indistinguishable. Only under the ranking HEADFINALITY-$\alpha \gg$ ANTISYMMETRY$\gg$ HEADFINALITYwill they give rise to the disharmonic order. This is illustrated in the tableau in (22):

(22) a.



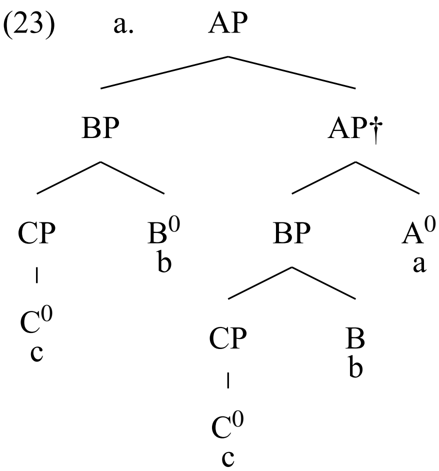
b.

	(a)	HF-BP	ANTISYM	HF
	abc	*BP		*AP *BP
	bac	*BP	$*a < b$	*AP *BP
	bca	*BP	$*a < b, *a < c$	*BP
	cba		$*a < b, *a < c, *b < c$	
	cab		$*a < c, *b < c$	*AP
→	acb		$*b < c$	*AP

Undominated HEADFINALITY-α effectively divides the syntactic structure into two domains: everything below α is linearized purely by HEADFINALITY-α , while everything above it is linearized by the combination of ANTISYMMETRY and HEADFINALITY, just as in the harmonic word order case. It's worth taking a moment to demonstrate that this applies even when movement is involved. There are two relevant cases: Movement of α itself, and movement of some phrase within α to a position outside of it. In both cases, we want the moved item to be head-final within itself, but positioned in a head-initial fashion with respect to the rest of the clause.

The case where α itself moves is illustrated in (23), where BP has moved to the specifier of AP. Both copies of BP (reflexively) dominate themselves, and so both are linearized head-finally; likewise, both copies of CP are dominated by a copy of BP, and so CP would also be linearized head-finally (if there were any other material in it). The only change is that A^0 no longer totally c-commands B^0 and C^0, so ANTISYMMETRY will fail to order it before them; instead, the general constraint HEADFINALITY will emerge to order the specifier on the left.

(23) a.

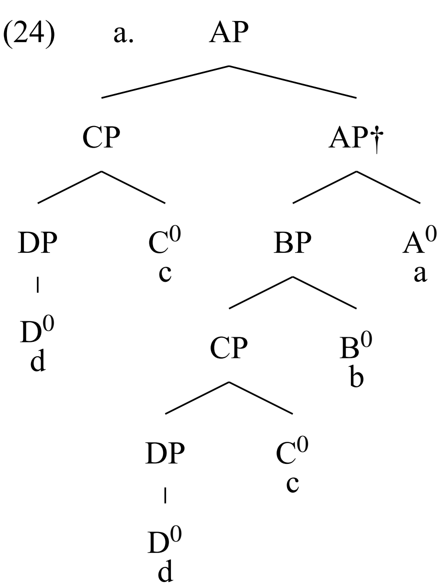


b.

(a)	HF-BP	ANTISYM	HF
abc	*BP		*AP *BP
bac	*BP		*AP *BP
bca	*BP		*BP
→ cba		$*b < c$	
cab		$*b < c$	*AP
acb		$*b < c$	*AP

Movement from within α requires a slightly larger tree to fully see. In (24), $\alpha = \text{BP}$ as before; this time, the complement of BP has moved up to the specifier of AP. Once again, HEADFINALITY-BP applies within CP, which is dominated (though not totally dominated) by BP; only the general HEADFINALITY orders the material in CP with respect to *a* and *b*, however, putting the moved item on the left.

(24) a.



b.

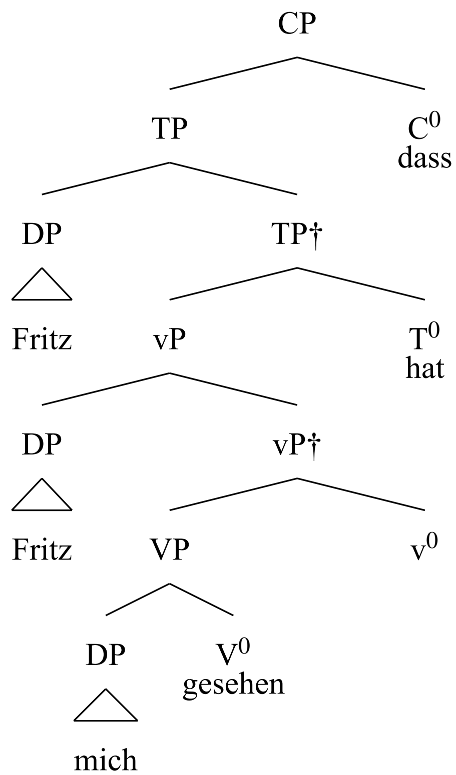
(a)	HF-αBP	ANTISYM	HF
→ dcab			*AP†
cdab	*CP		*AP† *CP
adcb	*CP		*AP *AP†
abdc			*AP *AP† *BP
dcba		$*a < b$	

I'll close this section by illustrating how the constraints described here linearize embedded clauses in German. German is a well-known case of a disharmonic word order: Complementizers are on the left, but the rest of the clausal spine is head-final. Thus, the domain of head-finality is TP;

that is, HEADFINALITY-TP is undominated. I've given a simplified syntactic structure in (25). [14]

(25) a. ... dass Fritz mich gesehen hat.
that Fritz me seen has
"...that Fritz has seen me."

b.



(26)

	(25b)	HF-TP	ANTISYM	HF
→	dass Fritz mich gesehen hat		3: *V < O, *Aux < V, *Aux<O	*CP
	Fritz mich gesehen hat dass		7: ... *C<S, *C<O, *C<V, *C<Aux	
	dass Fritz hat gesehen mich	*TP, *VP	0	*TP, *VP, *CP
	dass Fritz hat mich gesehen	*TP	1: *V<O	*TP, *CP

As shown in (26), the constraint HEADFINALITY-TP eliminates all candidates in which any head below TP is not final within its phrase. ANTISYMMETRY further eliminates those candidates where C^0, the only head not in the domain of head-finality, is not initial. The interaction of these

[14] For the purposes of this illustration, I'm ignoring the morphology of the verb itself.

two constraints derives the correct disharmonic word order.

5. PF effects on word order

Optimal Linearization is the proposal that linearization of syntactic structures, insofar as it is a post-syntactic phenomenon, should be calculated using a violable constraint framework. The focus of this paper so far has been on using Optimal Linearization to model word-order typology, in particular the FOFC, but an additional benefit of modelling linearization at PF is that it allows word order to interact with PF-specific aspects of language like phonology and prosody. There is a growing body of evidence to suggest that these PF factors can have an influence on word order: For example, 2nd position clitics in Serbo-Croatian sometimes seem to interrupt syntactic constituents, but never interrupt prosodic ones (e.g. Halpern 1992; Bošković 2001:and many others). Anttila et al. (2010) has shown that prosodic factors can influence the choice of syntactic frame for English ditransitives; more recently, Shih & Zuraw (2017) shows that segmental features influence adjective-noun order in Tagalog. López (2009) proposes that Clitic Right Dislocation in Romance is motivated by a prosodic constraint on phrasing the verb together with its extended projection. Along similar lines, Edmiston & Potsdam (2016) argues that right-extraposition of clauses follows from prosodic markedness constraints. Clemens (2016) argues that pseudo noun incorporation in Niuean is the result of a constraint requiring the verb to be prosodically phrased together with its argument; this analysis is extended to Mayan VOS orders by Clemens & Coon (2016). All of these cases seem to represent what we might call "PF displacement": PF-specific properties of language can create deviations from the word order expected on the basis of syntax.

Optimal Linearization is far from the first proposal for violable linearization at PF. López (2009); Edmiston & Potsdam (2016); Clemens (2016), and Clemens & Coon (2016) all rely on violable linearization in some form. Perhaps the most developed model of violable linearization is the one used by Bennett et al. (2016) to model Irish pronoun postposing, which is one of the clearest cases of PF displacement. Their model relies on a single word order faithfulness constraint penalizing deviations from a target order specified by the syntax. This is sharply different from Optimal Linearization: While both rely on violable constraints to enforce word order, the Bennett et al. (2016) model assumes that that order is given to PF by the syntax; Optimal Linearization instead calculates the word order directly at PF via the competition of multiple constraints. The primary benefit of the Optimal Linearization approach is that it models word order typology generally: If linearization is governed by a single faithfulness constraint, we gain no insight into why different languages are faithful to different word orders. It's worth taking a moment, however, to demonstrate that Optimal Linearization is in fact capable of modelling PF displacement, in particular the Irish pronoun postposing case; in fact, I argue that Optimal Linearization actually fairs better in this specific case than the word-order faithfulness model.

Elfner (2012), expanded by Bennett et al. (2016), show that Irish light object pronouns often appear far to the right of where object DPs would generally be expected, with no detectable difference in semantic or pragmatic import. For example, in (27) the pronominal object appears after the clause-final adjunct:

(27) Fuair sé ___ óna dheartháir an lá cheana é
get.PAST he from.his brother the-other-day it
"He got it from his brother the other day." (Bennett et al. 2016:171)

Bennett, Elfner, & McCloskey (2016) present convincing evidence that this displacement lacks the signature of a syntactic movement process. For one, this displacement seems to be optional, and doesn't correspond to any semantic or pragmatic effect. For another, the displacement often has rather implausible landing sites. An example of this is given in (28), where a light expletive pronoun has been displaced into the middle of a conjunction; if this were syntactic movement, it would be lowering movement and would seemingly violate the Coordinate Structure Constraint.

(28) is cuma ___ 'na shamhradh é nó 'na gheimhreadh
COP.PRES no.matter PRED summer it or PRED winter
"It doesn't matter whether it's summer or winter." (Bennett et al. 2016:183)

A final property of pronoun postposing that makes it a poor example of syntactic movement is that it affects only light, stressless pronouns; stressed pronouns appear in their expected position. This leads Elfner (2012) to propose that the postposing is a kind of prosodic repair: A constraint STRONGSTART (Selkirk 2011) militates against phonological phrases which begin with a light (sub-minimal word) element; this constraint outranks some relevant constraint enforcing linearization, and the result is that light pronouns are pronounced later in sentence in order to achieve a more harmonic prosody. The definition of STRONGSTART given by Bennett et al. is in (29); paraphrased, it will assign one violation for each node in the prosodic parse that is at least as big as a word but which begins with something smaller than a word. Stressless pronouns are argued to be light clitics rather than prosodic words, and hence are affected by STRONGSTART.

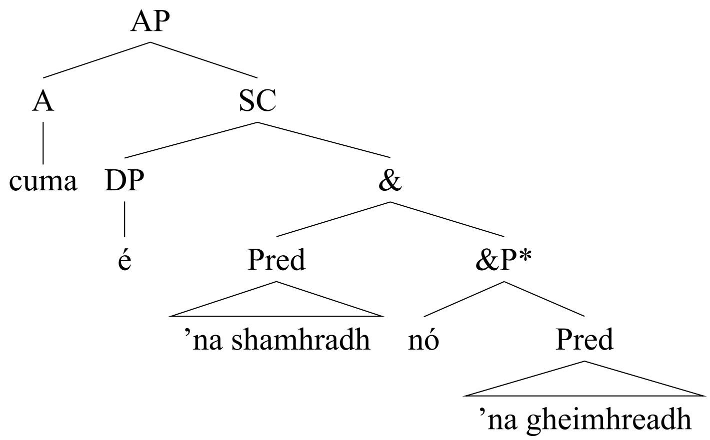
(29) STRONGSTART: Prosodic constituents above the level of the word should not have at their left edge an immediate sub-constituent that is prosodically dependent [i.e. smaller than a word]. (Bennett et al. 2016:198).

This notion of PF displacement captures the properties of pronoun postposing elegantly. Crucially, it introduces the idea that linearization is violable — the same insight on which Optimal Linearization is built. The particular implementation, however, assumes that the structure has already been ordered by PF; the authors are purposely vague about how this is arrived at, but propose a faithfulness constraint NOSHIFT penalizing deviations from the underlying order:

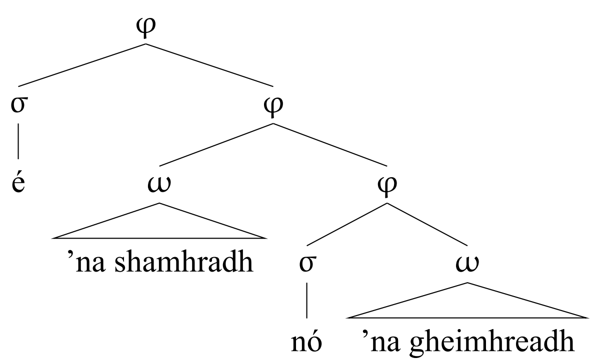
(30) NOSHIFT: If a terminal element α is linearly ordered before a terminal element β in the syntactic representation of an expression E, then the phonological exponent of α should precede the phonological exponent of β in the phonological representation of E.
(Bennett et al. 2016:202)

To illustrate how this enables them to account for pronoun shift, let's consider the schematized syntactic structure of (29) given by Bennett et al. The details of their prosodic analysis are beyond the scope of this paper, but the "faithful" prosody they predict is given in (32); the weak pronoun *é* winds up at the left edge of the phrase corresponding to the small clause.

(31) Syntactic structure of (28) (Bennett et al. 2016:184):



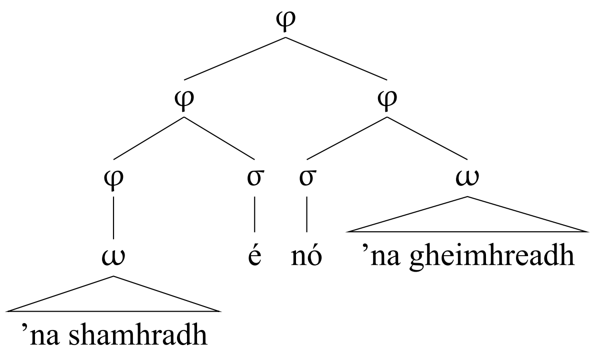
(32) Partial prosodic structure of (28) (Bennett et al. 2016:216):[15]



This structure, preferred by NOSHIFT and the other constraints enforcing prosodic phrasing, fares poorly with STRONGSTART: The highest phonological phrase has a sub-word element as its leftmost daughter. If STRONGSTART dominates NOSHIFT, a postposing structure like (33), in which no φ begins with a σ, is preferred:

[15] In this and all prosodic structure examples, φ represents a phonological phrase, while ω represents a prosodic word. These are two of the commonly-assumed levels of the prosodic hierarchy (Ito & Mester 2012). In tableaux, I will use parentheses to represent prosodic phrasing, and will leave prosodic word status unmarked.

(33) STRONGSTART-respecting word order:

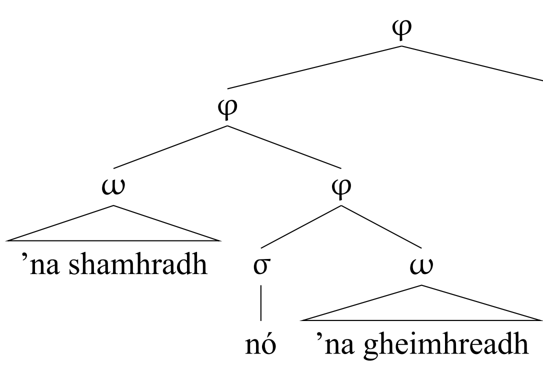


(34)

(31)	STRONGSTART	NOSHIFT
(é (’na shamhradh (nó ’na gheimhreadh))) = (32)	1	0
→ (((’na shamhradh) é) (nó ’na gheimhreadh)) = (33)	0	1

This is the desired result. Bennett et al. (2016) acknowledge that there is another possible output, namely the one in which the pronoun is postposed all the way to the end of the clause; they note that, in general, the landing site of pronoun postposing can be arbitrarily far to the right, and that the different landing sites are in free variation. They state that their proposed analysis correctly predicts the alternative structure in (35):

(35) Alternative ordering of (28) (Bennett et al. 2016:218):



However, it is not clear from their proposal that this result is, in fact, predicted. The NOSHIFT constraint, as written, assigns additional violations for each pair of syntactic elements which get reordered. But what counts as a syntactic element? If the answer is “all syntactic terminals” or even "all XPs", the result should be that additional violations will be assigned the further right the pronoun is displaced. Put another way, in the winning order of (34), the pronoun has only changed orders with the first predicate; in the order in (35) it has changed orders with the disjunction and the second predicate as well, and so NOSHIFT should assign additional violations. The result is that the candidate with minimal linear displacement should always win (modulo other prosodic factors); this is illustrated below.

(36)

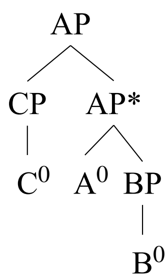
	(31)		STRONGSTART	NOSHIFT
	(é ('na shamhradh (nó 'na gheimhreadh)))	= (32)	1	0
→	((('na shamhradh) é) (nó 'na gheimhreadh))	= (33)	0	1
	(('na shamhradh) ((nó 'na gheimhreadh) é))	= (35)	0	3

Empirically, this seems to be the wrong prediction in the Irish case, and in fact Bennett et al. never show more than one violation of NOSHIFT being assigned to any given candidate. The definition of NOSHIFT given is deliberately intended to cover a number of possible ways of arriving at the desired linearization; given this, we might understand Bennett et al. to be assuming some linearization scheme which assigns at most one violation for postposing this pronoun.[16]

Optimal Linearization is such a scheme. While Irish is generally head-initial and so should have ANTISYMMETRY ≫ HEADFINALITY, I've shown that the ordering of specifiers is controlled by HEADFINALITY. That constraint crucially assigns violations by counting branching nodes in the syntax which are not linearized head-finally, rather than by counting pairs of words. Take the simplified example in (37). HEADFINALITY will assign a single violation whenever AP is not linearized head-finally, i.e whenever *a or b* precedes *c*. No further violations are assigned as *c* is displaced rightward — the first two candidates each receive only one violation.

[16]This is somewhat difficult to accomplish with a single constraint. The violable linearization scheme given in Bennett et al. 2016 is essentially a "string edit distance" function, i.e. a function that calculates how many changes would need to be made to one string of characters in order to produce another. In this system, some linearization (i.e. a string) is given by the syntax, and NOSHIFT scores each candidate on how "distant" it is from the target linearization. A distance function based on swapping characters in the string will always run into the problem described above: Every swap incurs additional penalties, and so there will always be pressure for extremely local displacement.

(37) a.



b.

	(a)	HEADFINALITY
	bac	1
	bca	1
→	cba	0

HEADFINALITY, then, is the tool with which to analyze the Irish postposing case: No additional violations are assigned as the pronoun is displaced further rightward. If both STRONGSTART and ANTISYMMETRY dominate HEADFINALITY, we achieve the correct result.

(38)

	(28)		ANTISYM	SS	HF
	(é (’na shamhradh (nó ’na gheimhreadh)))	= (32)	0	1	0
→	(((’na shamhradh) é) (nó ’na gheimhreadh))	= (33)	0	0	1
→	((’na shamhradh) ((nó ’na gheimhreadh) é))	= (35)	0	0	1

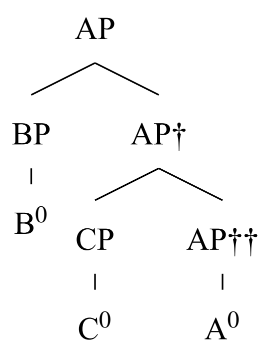
Both of the winning candidates in (38) respect both STRONGSTART and ANTISYMMETRY. Both violate HEADFINALITY in that some element of the conjunction precedes the pronoun, but crucially they both violate this equally and so both emerge as winners. Thus, the Optimal Linearization constraints fare better than the plain NOSHIFT. More needs to be done to understand how these constraints interact with other prosodic faithfulness and markedness constraints (e.g. the MATCH constraints), but Optimal Linearization is at least compatible with PF displacement, and in at least this case is preferable to having a single “linearization faithfulness” constraint.

6. Conclusion

Optimal Linearization is the proposal that linearization is accomplished at PF by a set of violable constraints which make reference to the syntactic structure I’ve shown that this model is capable of making detailed predictions about word order typology; I’ve also shown that it gives new insight into the asymmetric positioning of specifiers, allowing us to understand it as an emergence of an unmarked preference for head-finality. Finally, I’ve shown that this approach allows us to account for cases of PF displacement and in fact fairs better than approaches based on a single word order faithfulness constraint.

There is one notable aspect of linearization which has not been taken up here, namely the ordering of adjuncts. The constraints as presently defined will treat adjuncts identically to specifiers. For example, in (39), take CP to be some modifier phrase adjoined to AP. Similar to the specifier case, C^0 neither c-commands nor is c-commanded by any other head in this structure, and so ANTISYMMETRY is silent on its ordering; HEADFINALITY will prefer to order AP† head-finally, i.e. with $c < a$. Similar logic results in $b < c$. From this we can generalize that adjuncts will universally be linearized before their head but after the specifier, regardless of constraint ranking.

(39) a.



b.

	(a)	ANTISYMMETRY	HEADFINALITY
	abc	0	2
	bac	0	1
→	bca	0	0
	cba	0	1
	cab	0	1
	acb	0	2

This is not a desirable result, insofar as right-adjunction is quite common. Perhaps more interestingly, adjuncts are known to be extremely variable in their distribution (Ernst 2001), both across and within specific kinds of adjuncts. Untangling this complex distribution will require other factors beyond the three constraints presented here. In some cases, the complex distribution of adjuncts has been taken to reflect more complex syntactic structure (as in e.g. Cinque 1999). In other cases, it seems that the syntactic (or possibly prosodic) weight controls whether adjuncts are on the left or the right of their head, as in English examples like *a big dog* vs. *a dog bigger than me*. Roberts (2017) presents evidence that the positioning of adjuncts is, in fact, subject to the FOFC, so the constraints presented here still have a role to play in any analysis of their distribution, but considerably more refined tools will be needed.

Additionally, there is one aspect of the FOFC that these constraints do not capture: it only applies within certain domains. For example, German DPs appear to be head-initial, even though they are often contained inside the head-final TP; more generally, DP-internal ordering and the ordering of elements in the clausal spine seem to be independent of one another as far as the FOFC is concerned. Biberauer et al. (2014) codify this by restricting the FOFC to looking at heads within one Extended Projection (Grimshaw 1991). Optimal Linearization is certainly compatible with such a notion; one possible analysis would involve a stringent version of HEADFINALITY that is relativized not to some node but rather to an entire Extended Projection — for instance, in the case of German, one that examined only nodes in the verbal EP. There's also another possible explanation: Perhaps linearization precedes by phase (as in e.g. Fox & Pesetsky 2005), with the possibility that the linearization constraints are ranked differently for the DP-phase and the CP-phase. Without committing to this particular analysis, I will leave further investigation of these options aside for now.

This is far from the first time that PF constraints have been proposed which make reference to the syntax. There is a large family of "prosodic faithfulness" constraints which enforce correspondence between syntactic and prosodic structures. For example, the MATCH constraints (Selkirk 2011) ensure that syntactic constituents are matched by prosodic constituents that dominate the same set of terminal nodes. These constraints must have access to the syntactic structure, and in fact must even have access to the labelling of syntactic nodes in order to distinguish words, phrases, and clauses. Similarly, Clemens (2014) proposes the constraint ARG-φ, which penalizes prosodic structures in which heads and their arguments are not phrased together; this constraint needs access to selection relations.

The Optimal Linearization constraints fit this pattern: They use c-command, dominance, and labelling to choose between differently-ordered candidates. In so doing, they accomplish three main things. First, it captures the same empirical facts about linearization that are encoded in the classical Headedness Parameter, but does so using a constraint-based model consistent with how other PF-branch phenomena are treated. This frees us from having to stipulate properties like the placement of specifiers, instead allowing these properties to emerge from constraint interactions. Second, Optimal Linearization additionally allows for the disharmonic orders consistent with the FOFC without needing to stipulate any new syntactic principles — we can build syntactic trees exactly as before while still deriving the correct orders. And finally, Optimal Linearization provides a model for interactions between linearization and phonological or prosodic markedness that, in at least some cases, fits the empirical data better than a single word-order faithfulness constraint does.

References

Abels, Klaus. 2003. Successive cyclicity, anti-locality, and adposition stranding. Doctoral Dissertation, University of Connecticut.

Abels, Klaus, & Ad Neeleman. 2012. Linear asymmetries and the LCA. *Syntax* 15:25–74.

Anttila, Arto, Matthew Adams, & Michael Speriosu. 2010. The role of prosody in the english dative alternation. *Language and Cognitive Processes* 25:946–981.

Bennett, Ryan, Emily Elfner, & James McCloskey. 2016. Lightest to the right: An apparently anomalous displacement in Irish. *Linguistic Inquiry* 47:169–234.

Biberauer, Theresa, Anders Holmberg, & Ian Roberts. 2014. A syntactic universal and its consequences. *Linguistic Inquiry* 45:169–225.

Bošković, Željko. 2001. *On the nature of the syntax-phonology interface: Cliticization and related phenomena*. Amsterdam: Elsevier Science.

Cable, Seth. 2004. Predicate clefts and base-generation: Evidence from Yiddish and Brazilian Portuguese. Unpublished manuscript.

Chomsky, Noam. 1995. *The Minimalist Program*. Cambridge, Massachusetts: MIT Press.

Cinque, Guglielmo. 1999. *Adverbs and functional heads: a cross-linguistic perspective*. New York: Oxford University Press.

Clemens, Lauren Eby. 2014. Prosodic noun incorporation and verb-initial syntax. Doctoral Dissertation, Harvard University.

Clemens, Lauren Eby. 2016. Argument-φ: A prosodic account of pseudo noun incorporation. In *Proceedings of NELS 46*, ed. Brandon Prickett & Christopher Hammerly, 217–226.

Clemens, Lauren Eby, & Jessica Coon. 2016. Deriving verb initial word order in Mayan. Presentation at NELS 47.

Cole, Peter, & Gabriella Hermon. 1998. The typology of wh-movement. Wh-questions in Malay. *Syntax* 1:221–258.

Dryer, Matthew S., & Martin Haspelmath, ed. 2013. *WALS Online*. Leipzig: Max Planck Institute for Evolutionary Anthropology. URL http://wals.info/.

Dumestre, Gérard. 1984. *Dictionairre bambara-français*. Paris: Institut National des Langues et Civilsations Orientales.

Edmiston, Daniel, & Eric Potsdam. 2016. Linearization at pf: Evidence from malagasy extraposition. Presented at NELS 47, University of Massachusetts Amherst.

Elfner, Emily. 2012. Syntax-prosody interactions in Irish. Doctoral Dissertation, University of Massachusetts Amherst.

Ernst, Thomas. 2001. *The syntax of adjuncts*, volume 96. Cambridge University Press.

Fox, Danny, & David Pesetsky. 2005. Cyclic linearization of syntactic structure. *Theoretical linguistics* 31:1–45.

Grimshaw, Jane. 1991. Extended projection. Unpublished manuscript, Brandeis University.

Halpern, Aaron Lars. 1992. Topics in the placement and morphology of clitics. Doctoral Dissertation, Stanford University Ph. D. dissertation.

Ito, Junko, & Armin Mester. 2012. Recursive prosodic phrasing in Japanese. *Prosody matters: Essays in honor of Elisabeth Selkirk* 280–303.

Johnson, Kyle. 2016. Towards a multidominant theory of movement. Lectures presented at ACTL, University College.

Kandybowicz, Jason. 2006. On fusion and multiple copy spell-out: The case of verb repetition. In *The copy theory of movement on the PF side*, ed. Jairo Nunes & Norbert Corver. Oxford University Press.

Kastenholz, Raimund. 2003. Auxiliaries, grammaticalization and word order in Mande. *Journal of African Languages and Linguistics* 24:31 – 53.

Kayne, Richard S. 1994a. *The antisymmetry of syntax*, volume 25 of *Linguistic Inquiry Monographs*. MIT Press.

Kayne, Richard S. 1994b. *The antisymmetry of syntax*. Cambridge, Massachusetts: MIT Press.

Koopman, Hilda. 1984. *The syntax of verbs*. Dordrecht, The Netherlands: Foris Publications.

López, Luis. 2009. Ranking the linear correspondence axiom. *Linguistic Inquiry* 40.2:239–276.

Lukas, J., & G. Nachtigal. 1939. *Die sprache der buduma im zentralen sudan: auf grund eigener studien und des nachlasses von g. nachtigal*. Abhandlungen für die Kunde des Morgenlandes. Deutsche morgenländische gesellschaft, Kommissionsverlag F.A.Brockhaus. URL https://books.google.com/books?id=ZzQpOwAACAAJ.

McCarthy, John J. 2002. *A Thematic Guide to Optimality Theory*. Cambridge University Press.

McCarthy, John J, & Alan S Prince. 1994. The emergence of the unmarked: Optimality in prosodic morphology. In *Proceedings of the North East Linguistic Society 24*. GLSA (Graduate Linguistic Student Association), Dept. of Linguistics, University of Massachusetts.

Prince, Alan, & Paul Smolensky. 1993/2004. *Optimality Theory: Constraint interaction in generative grammar*. Blackwell Publishers.

Roberts, Ian. 2017. The Final-over-Final Condition and Adverbs. In *The Final-over-Final Condition*, ed. Michelle Sheehan, Theresa Biberauer, Ian Roberts, & Anders Holmberg, chapter 6, 97 – 120. MIT Press.

Selkirk, Elisabeth. 2011. The Syntax-Phonology Interface. In *The Handbook of Phonological Theory*, ed. John Goldsmith, Jason Riggle, & Alan Yu, volume 2, chapter 14, 435–483. Oxford: Blackwell Publishing.

Sheehan, Michelle, Theresa Biberauer, Ian Roberts, & Anders Holmberg. 2017. *The Final-over-Final Condition: A Syntactic Universal*, volume 76. MIT Press.

Shih, Stephanie S., & Kie Zuraw. 2017. Phonological conditions on variable adjjective and noun word order in Tagolog. *Language* 93:317 — 352.

Sichel, Ivy. 2014. Resumptive pronouns and competition. *Linguistic Inquiry* 45:655–693.

Watanabe, Akira. 1992. Subjacency and S-structure movement of *wh*-in-situ. *Journal of East Asian Linguistics* 1:255–291.