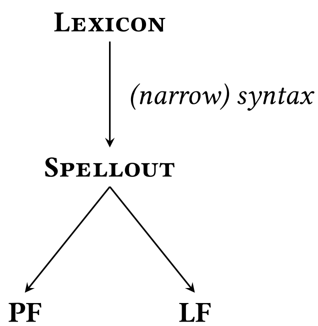
Williams 2013: GSGM

LING 252 · Ethan Poole · 27 January 2022

1 The architecture

- ***The standard view: Y-model***
  Syntax is generative, while PF and LF are interpretive:

(1)


⁕ ***Generative Semantics, Generative Morphosyntax (GSGM)***

- PF and LF are constructed in tandem; both are generative.
- The functional sequence F is a "clock" that times events in the derivation; it only indirectly shapes clause structure.
- Each F_i calls one semantic function and one or more syntactic functions.

⇒ There is no structure that gets interpreted (in the Spellout sense).

- ***Combine***
  The syntactic function is Combine, which puts together two syntactic objects. It involves two components: the PRODUCT and the product's PLACEMENT.

(2) Combine(X, YP)
  product:
    if M has a value:
      product = [X + Search(M-value, YP)]
    else:
      product = X
  placement:
    if P = XP:
      placement = [product + YP]
    else if P = X^0:
      placement = [. . . [product + Search(P-value, YP)] . . .]

- ***Parameters of Combine(X, YP)***
  - M = the kind of element to search for in YP, to then combine with X

(3)
  a. M = _stem ⇝ find closest stem; prefix X to it
  b. M = root_ ⇝ find closest root; suffix X to it
  c. M = nonlexical verb ⇝ find closest nonlexical verb; combine with X

- P = the position of the product in or w.r.t. YP

(4) a. P = _XP ⇝ place before closest phrase

b. P = X^0_ ⇝ place after closest head

⇒ Like Representation Theory, GSGM is a *derivational* theory. Most of its explanations involve derivational timing.

- ***Comparisons***
  - Not Generative Semantics: syntax is generative, not interpretive
  - Not CG: no parallelism between syntactic and semantic functions
  - Not GPSG: no reduction of movement to PS-rules or feature percolation
  - GSGM is most similar to Montague Grammar, in which syntax and semantics stand in a homomorphic relation and thus each syntactic rule is paired with one semantic rule.
  - Combine operates on workspaces, like Merge in Chomsky's recent work.

2 Combine

⁕ ***The gist***
Combine is a unified replacement for (i) Merge, (ii) head movement and (iii) morphological readjustment rules.

2.1 V-to-T vs. Affix Hopping

- Consider V-to-T movement and Affix Hopping:

(5) a. **French: V-to-T movement**

Jean T+embrasse souvent [___ Marie]
Jean T+kiss often Marie
'Jean often kisses Marie'

b. **English: Affix Hopping**

John ___ often [T+kiss Mary].

- ***The standard analysis***[1]
  - V-to-T movement happens in the syntax → generative
  - Affix Hopping happens at PF → interpretive

[1] There is, of course, the alternative analysis were head movement happens at PF, and thus both operations are part of the interpretive component.

- ***Commonalities***
This analysis misses the commonalities between the two operations:
  - Both obey the HMC:[2]

[2] French gloss and translation are, unfortunately, my own . . .

(6) a. **French**

*Jean T+embrasse avoir [___ Marie]
Jean T+kiss have Marie
Intended: 'Jean kissed Marie'

b. **English**
*John ___ have often [kiss+T Mary].

– Both are clause-bounded:

(7) a. John said Mary has left.

b. *Has John said Mary ___ left?

– Both violate the Extension Condition, as they do not target the root.

⇒ On the standard analysis, these commonalities are effectively an accident, which must be stipulated.

⇒ Crucially, being completely clause-bounded distinguishes V-to-T movement and Affix Hopping from phrasal movement.

⁕ ***Williams' analysis***

– What V-to-T movement and Affix Hopping share is that they create the same morphosyntactic object (the PRODUCT):

(8) **Both French and English**
. . . T . . . V . . . → . . . [T + V]

– They differ in the PLACEMENT of this product:

(9) a. **French**
COMBINE(T, [$_{VP}$ Adv V . . .]) = [$_{VP}$ [**T+V**] [Adv . . .]]   P = XP

b. **English**
COMBINE(T, [$_{VP}$ Adv V . . .]) = [$_{VP}$ Adv [**T+V**] . . .]   P = X^0

– There is no violation of the Extension Condition in either case: VP is enlarged by one element; there is never any structure [T VP].

– Semantically, they both apply a semantic function to the pair (T, VP).[3]

[3] Williams does not speculate on what the semantic function is here, but it seems compatible with either pronominal or quantificational tense.

- The architecture more schematically, where MS is a morphosyntactic object and "MS" is its meaning:

(10)
"MS_3" = Semantic Function ("MS_1", "MS_2")
/
Old workspace: {(MS_1, "MS_1"), (MS_2, "MS_2")…}
\
MS_3 = Combine ("MS_1", "MS_2")
New workspace: {(MS_3, "MS_3")…}

2.2 Product constitution

- ***No M value → Merge***
If M has no value, then the product is the first argument of COMBINE, placed in accordance with the P value. When P = XP, this is equivalent to MERGE:

(11) COMBINE(that, TP) = [$_{CP}$ that TP]   M = ∅; P = XP

⇒ When M has a value, it can lead to displacement (i.e. movement).

- ***When M = root***
  - In English, tense attaches to the verb root, not the stem, because it can trigger root allomorphy:[4]

(12) a. { PAST, [$_{VP}$ retake it] } → { [$_{VP}$ re [take+PAST] it] } ⇝ *retook*
b. { PAST, [$_{VP}$ retake it] } ↛ { [$_{VP}$ [retake+PAST] it] } ⇝ **retaked*

  - The French simple past exhibits the same pattern:

(13) a. { PAST, voir } → { voir+PAST } ⇝ *vit*
b. { PAST, [pre voir] } → { [pre [voir+PAST]] } ⇝ *prévit*

- ***When M = stem***
  In French, unlike the simple past, the future attaches to the verb *stem*, and thus derivational affixes bleed root allomorphy:

(14) a. { FUT, [pre voir] } → { [[pre voir]+FUT] } ⇝ *prévoira*
b. { FUT, [pre voir] } ↛ { [pre [voir+FUT]] } ⇝ **préverra*
c. { FUT, voir } ↛ { voir+FUT } ⇝ *verra*

- ***Parametrizing for directionality***
  We can account for suffix vs. prefix by parameterizing M with directionality:

(15) a. _stem ⇝ prefix to stem
b. _root ⇝ prefix to root
c. stem_ ⇝ suffix to stem
d. root_ ⇝ suffix to root

2.3 Placement

- ***Possible values of P***

(16) a. $\_X^0$ ⇝ place before closest head
b. _XP ⇝ place before closest phrase
c. $X^0\_$ ⇝ place after closest head
d. XP_ ⇝ place after closest phrase

- Note: SEARCH is defined in a way to only traverse from a head to its functional complement, i.e. down the clausal spine.

- ***Stacking vs. smallest head***
  - Consider the different outcomes for two affixes where M = root and $\text{Affix}_1 > \text{Affix}_2$:

(17) a. **Stacking**
[Affix_1 [Affix_2 root]]
b. **Smallest head targeted**
[Affix_2 [Affix_1 root]]

[4] In Halle and Marantz (1993, 1994), this kind of root allomorphy ('ablaut' in their terms) is handled by readjustment rules operating over strings, and so can handle this pattern as well.

⇒ Williams assumes the following:

(18) a. M = root ⇒ smallest head
 b. M = stem ⇒ stacking
 c. P = X^0 ⇒ smallest head
 d. P = XP ⇒ stacking

- ***In Williams 2013, but not here***
 1. How this system derives the preference for mirroring
 2. How this system derives U20[5]

[5] Essentially a reimplementation of Cinque (2005).

3 Bounding

3.1 Relative bounding

⁎ ***F-clock***
F(unctional)-structure constitutes a "clock" that times Combine operations:[6,7]

[6] This idea builds on Representation Theory (Williams 2003).

[7] I don't know what "A det VP" means.

(19) a. F-structure as spine of the clause b. F-structure as derivational "clock"

- ***Intervention***
At any given point in the derivation, the minimality interveners are determined by what has happened *morphologically* so far. Not every F_i will introduce an intervener.

- ***Example: Auxiliary raising***[8]
 – English allows up to four stacked auxiliaries, the ordering of which is fixed:[9]

(20) a. Alex might have been being chased.
 b. Alex could have been being interviewed.

(21) **English auxiliary ordering**
modal > perfect > progressive > passive > verb phrase

 – We account for this fact by positing a series of functional heads:

(22) a. $[_{TP}$ Alex $[_{\overline{T}}$ **might** $[_{AspP}$ **have** $[_{EvtP}$ **been** $[_{VoiceP}$ **being** $[_{VP}$ chased $]]]]]]$
 b. $[_{TP}$ Alex $[_{\overline{T}}$ **might** $[_{AspP}$ ∅ $[_{EvtP}$ **be** $[_{VoiceP}$ **being** $[_{VP}$ chased $]]]]]]$
 c. $[_{TP}$ Alex $[_{\overline{T}}$ **might** $[_{AspP}$ ∅ $[_{EvtP}$ ∅ $[_{VoiceP}$ **be** $[_{VP}$ chased $]]]]]]$
 d. $[_{TP}$ Alex $[_{\overline{T}}$ **might** $[_{AspP}$ ∅ $[_{EvtP}$ ∅ $[_{VoiceP}$ ∅ $[_{VP}$ chase Maria $]]]]]]$

 – If T is not filled with a modal, the highest instance of *have* or *be* raises to T.
 – **Problem:** The null variants of Asp, Evt, and Voice should not be allowed to raise to T, as this would overgenerate:

[8] Williams' example is subject-auxiliary inversion in polar questions, but this case can be straightforwardly handled by the movement targeting only T. Auxiliary raising presents the same issue, but isn't amenable to an easy fix.

[9] Chomsky (1957)

(23) ✗[$_{CP}$ C+T+Asp [$_{TP}$ Alex ~~T+Asp~~ [$_{AspP}$ ~~Asp~~ [$_{Evt}$ **be** [eating nattoo]]]]]]?

- **Solution from 200B:** Relativize the movement to [AUX]-bearing elements. Only overt Asp, Evt, and Voice heads bear [AUX].
- This "solution" is not really a solution. It just punts the problem up the formal pipeline. Why would null Asp, Evt, and Voice not be able to bear [AUX]?

⇒ Under GSGM, we can make reference to the morphological form, in particular whether it is overt, and thus avoid this issue entirely:

(24) F_T: COMBINE(T, [$\varnothing_{Asp}$ be eating natto]) = [T+be [$\varnothing_{Asp}$ ___ eating natto]]
where M = nonlexical verb, P = _XP

- ***Long head movement***
  Williams extends the same kind of analysis to long head movement in Bulgarian, treating the moved-over item as a clitic:

(25) Procel sum ___ knigata
read I.have the.book
'I have read the book'

(26) F_{sum}: COMBINE(have, [read the.book]) → [read have [___ the.book]]
where M = word_ , P = _XP

3.2 Absolute bounding

- Recall that we also want the movement induced by COMBINE to be clause-bounded:

(27) a. John said Mary has left.
b. *Has John said Mary ___ left?

∗ ***SEARCH***

(28) SEARCH(Value, YP) =
if Y satisfies Value:
Y
else:
SEARCH(Value, WP), where WP is the functional complement of Y

⇒ ***Only within the same functional domain***
The iterative part of SEARCH's definition restricts the search to the same functional domain, i.e. within the same clause.

- ***Example: Subject-auxiliary inversion***

(29) a. **Start search**
SEARCH(nonlexical verb, TP)

b. **Because T is not a nonlexical verb, iterate**
SEARCH(nonlexical verb, VP)

c. **Because V is not a nonlexical verb and its complement is not a functional complement**
abort

- ***Leaks at F_0***
  - The definition of SEARCH always allows checking YP for satisfaction, regardless of whether YP is a functional complement.
  - Crucially, in the case of F_0, the complement may not be a functional complement.
  - ⇒ Thus, there is a "leak" in the system, allowing crossclausal interaction between F_0 and its complement.
  - But to reiterate: F_j for $j > 0$ *cannot* search the complement of F_0.
  - Williams' discussion implies that F_0 cannot search past the topmost projection of its complement F'_n (=YP). However, I don't see how this follows from the definition in (28), because F'_{n-1} is the functional complement of F'_n, and so forth.

4 Semantics

4.1 Adverbs

⁎ ***The gist***
Adverbs are introduced in the derivation where they take scope, but they can be positioned lower by varying M and P.

- ***Example: Evidently***
  - The adverb *evidently* takes scope over tense, so presumably *evidently* > PAST, but it can appear after the highest auxiliary:

(30) {**Evidently**} Caesar had {**evidently**} been {***evidently**} giving {***evidently**} out false info.

  - In GSGM, this can be handled by setting P = $\text{aux}V^0\_$ for $F_{\text{evidently}}$:

(31) COMBINE(evidently, [Caesar [have+PAST [been ...]]])
= [Caesar [have+PAST evidently [been ...]]]

4.2 DPs

⁎ ***Co-generation***
DPs and clauses are embedded in fundamentally different ways.

  - When COMBINE targets an NP, it introduces it into the clause with a POINTER:

(32) {see, man, ...} → {[see *j*], man_j, ...}

  - The NP remains in the workspace and can be modified by F_i.
  - Determiners are introduced into the derivation at the point where the DP takes scope.
  - Finally, a pronunciation rule specifies how pointers are read out:

(33) **Read-out rule 1**
Do not read out a pointer, read out the thing pointed to, but only once, and only for the highest instance of the pointer.

- ***Example derivation (with lots of liberties taken)***[10]

(34) a. F_V : { cat_j, mouse_k, catch }
COMBINE(k, catch) = [$_{VP}$ catch k]
New workspace: { cat_j, mouse_k, [$_{VP}$ catch k] }

b. F_v : { cat_j, mouse_k, [$_{VP}$ catch k] }
COMBINE(j, VP) = [$_{VP}$ j [catch k]]
New workspace: { cat_j, mouse_k, [$_{VP}$ j [catch k]] }

c. F_{some} : { cat_j, mouse_k, [$_{VP}$ j [catch k]] }
Semantics: SOME(cat_j, [$_{VP}$ j [catch k]])
COMBINE(some, cat_j) = [$_{NP}$ some cat]$_j$
New workspace: { [$_{NP}$ some cat]$_j$, mouse_k, [$_{VP}$ j [catch k]] }

d. F_{every} : { [$_{NP}$ some cat]$_j$, mouse_k, [$_{VP}$ j [catch k]] }
Semantics: EVERY(mouse_k, [$_{VP}$ j [catch k]])
COMBINE(every, mouse_k) = [$_{NP}$ every mouse]$_k$
New workspace: { [$_{NP}$ some cat]$_j$, [$_{NP}$ every mouse]$_k$, [$_{VP}$ j [catch k]] }

e. Read out as: *Some cat caught every mouse.*

[10] Williams (2013, 2014) does not give any examples with multiple quantificational DPs, so it isn't clear to me how it handles scope ambiguity.

- ***Two constraints***

1. ***No vacuous quantification***
COMBINE(j, XP) is blocked where XP does not vary with j, though this cannot apply at F_0 (and possibly any argument-introducing F_i).

2. ***Uniqueness***
COMBINE(j,) can only be applied once at a given F_i:

(35) a. {man, think, die} $\nrightarrow$ {man_j, [j think], [j die]}

b. *Every man thinks dies.

- ***Similar to***
  - "Quantifying in" in Montague Grammar
  - Beghelli and Stowell (1997): Different quantifiers must raise to designated specifier positions in the structure
  - Sportiche (2005): Determiners are introduced outside VP
  - Fox and Johnson (2016): QR is restrictor sharing

- ***More on scope?***
There is further discussion of scope in section 10, but I don't quite follow the relevant data, as it makes claims about the relative scope of two universals:

(36) John must win every single remaining game. must $\gg$ every

(37) In order for me to win the bet, in the one-year period in question, at least one judge must vote that every single litigant has no standing to sue.

a. *must $\gg$ every $\gg$ vote

b. every $\gg$ must $\gg$ vote

c. must $\gg$ vote $\gg$ every

5 Movement

5.1 Constituent questions

⁎ *Wh*-movement is treated as quantification:

(38) Semantics: *wh*("man$_j$", "[you think that John saw j]")
/
F_{wh}:{man$_j$, [you think that John saw j]}
\
Combine (which$_{_XP}$, man$_j$) = [which man$_j$]$_j$
Combine (j, [do you think John saw j]) = [**j** [do you think John saw **j**]]
New workspace: {[which man]$_j$, [**j** [do you think John saw **j**]]}

- Here, the ban on vacuous quantification rules out merging in a pointer that is not yet in the structure:

(39) *wh*("[man]$_j$", "[Bill saw k]")
/
{[man]$_j$, [woman]$_k$, [Bill saw k]}
\
Combine (which, [man]$_j$) = [which man]$_j$
Combine (j, [Bill saw k]) = [j [Bill saw k]]

new workspace: {[j [Bill saw k]], [which man]$_j$}

5.2 LEC

⁎ ***Level Embedding Conjecture (LEC)***
As in Representation Theory, GSGM is constrained by the LEC:

(40) Level Embedding Conjecture
All clauses are developed in the workspace simultaneously, and embedding of one in another takes place at the point at which the clause to be embedded has reached the "size" in F-clock terms required by the embedding predicate.

(41) a. F_0: {John, say$_{_CP}$, he, laugh}
b. $F_{i.}$: {[John say$_{_CP}$], [he laugh]} — form both clauses
c. $F_{T.}$:{[John said$_{_CP}$]$_{TP,}$ [he laughed]$_{TP}$} — … grow them into TPs
d. F_C: {[John said$_{_CP}$]$_{CP,}$ [that [he laughed]$_{TP}$]$_{CP}$} → — …grow them into CPs
e. F_C: {[John said [that [he laughed]$_{TP}$]$_{CP}$]$_{CP}$} — embed

⇒ ***No successive cyclicity***

- Williams is explicit that the LEC does not permit successive-cyclic movement ("subjacent" movement in his terms), because there is only one cycle under the F-clock.
- Uniqueness would also block any iterative instances of the same movement type, even if embedding itself could happen iteratively.

⁎ ***Islandhood as relative***

(42) F_iP will be an island for a rule targeting SpecF_j exactly when $F_j \leq F_i$.[11]

[11] Interestingly, it went from ‘<’ in 2003 to ‘≤’ in 2013. I suspect though that this means little in practice, as one can always posit more F-structure.

- ***Example: Relativization and questions***
  - Let us assume that specificational copular constructions require narrow focus on the postcopular DP:

(43) a. John is the mayor. *predicational*
b. The major is [$_F$ John]. *specificational*

  - Relativization cannot target the postcopular position, but *wh*-movement can:

(44) a. ?I wonder who [___ is the mayor]. *predicational*
b. I wonder who [the mayor is ___]. *specificational*
c. I met the man who [___ is the mayor]. *predicational*
d. *I met the man who [the mayor is ___]. *specificational*

  - In this respect, relativization patterns with topicalization:[12]

[12] The sentences in the paper meant to illustrate this point move the wrong DP, but the claim still holds.

(45) a. John, I think [___ is the mayor]. *predicational*
b. *John, I think [the mayor is ___]. *specificational*

⇒ **Analysis:** Relativization targets a Topic, and *wh*-movement a Focus.

  - **Prediction:** If we assume Topic > Focus (following Rizzi 1997), then the LEC makes a prediction: it should be possible to relativize/topicalize out of a question, but not vice versa:

(46) a. ***Who** do you know the man [$_{RC}$ who likes ___]?
b. ?the man **who** I wonder [$_Q$ who likes ___]

(47) a. ***Who** did Sue say that [$_{TOP}$ the box, Mary gave to ___]?
b. **Bill**, I wonder [$_Q$ who likes ___]?

  - I have never been compelled by these data because of the following:[13]

[13] Baltin (1982); Culicover (1996); Keine (2016)

(48) **Topicalization does not bleed relativization**

a. He’s a man [[**to whom**]$_{REL}$ [**liberty**]$_{TOP}$ we could never grant ___$_{TOP}$ ___$_{REL}$].

b. He’s a man [[**to whom**]$_{REL}$ I believe [[**liberty**]$_{TOP}$ we could never grant ___$_{TOP}$ ___$_{REL}$]].

(49) **Topicalization does not bleed *wh*-movement**

a. I was wondering [**to what kinds of people**]$_{WH}$ [**books like these**]$_{TOP}$ you would actually have given ___$_{TOP}$ ___$_{WH}$ if you had had the chance.

b. [**To what kinds of people**]$_{WH}$ did she say [(that) [**books like these**]$_{TOP}$ you would actually have given ___$_{TOP}$ ___$_{WH}$ if you had had the chance]?

- ***What about morphosyntactic reflexes of successive cyclicity?***
  Williams (2011:ch. 7) very briefly discusses Irish lenition (*aL* vs. *go* complementizers) and French stylistic inversion.
  - **Analysis of Irish:** The matrix movement target is [Spec, F_8]; *aL*-clauses are F_7; and *go*-clauses are F_9.
  - Thus, movement out of *go*-clauses to [Spec, F_8] is blocked by the LEC ($F_9 > F_8$), but movement out of *aL*-clauses is not ($F_7 \not> F_8$).
  - **Analysis of French:** The matrix movement target is [Spec, F_8]; inversion clauses are F_7; and non-inversion clauses are F_9.

- ***Problems with this approach***
  - This analysis only accounts for half of the generalization: movement is only allowed out of certain clause types.
  - ⇒ However, it completely misses the other half of the generalization: those certain clause types only occur in the context of movement.
  - Moreover, it does not extend to reflexes where the moved element needs to be in the intermediate position, e.g. complementizer agreement in Wolof[14] or reconstruction to intermediate positions.

[14] Torrence (2012)

6 Discussion

- ***Absolute bounding***
  - Is absolute bounding redundant given the LEC?
  - That F_0–F_n interactions would be possible at all seems incompatible with the LEC.

- ***Moving non-DPs***
  - Williams argues that at least some cases of clauses moving are actually clauses base-generated and mediated by a null operator:

    (50) [That John is here]$_{\text{TOP}}$ [DP_1 [I was not aware ***(of)** ___$_1$]]

  - What about clausal extraposition?
  - What about VP and AP movement?

References

Baltin, Mark. 1982. A landing site theory of movement rules. *Linguistic Inquiry* 13:1–38.

Beghelli, Filippo, and Timothy Stowell. 1997. Distributivity and negation: The syntax of *each* and *every*. In *Ways of Scope Taking*, ed. Anna Szabolcsi, 71–107. Dordrecht: Kluwer.

Chomsky, Noam. 1957. *Syntactic Structures*. The Hague: Mouton.

Cinque, Guglielmo. 2005. Deriving Greenberg's Universal 20 and its exceptions. *Linguistic Inquiry* 36:315–332.

Culicover, Peter. 1996. On distinguishing a′-movements. *Linguistic Inquiry* 27:445–463.

Fox, Danny, and Kyle Johnson. 2016. QR is restrictor sharing. In *Proceedings of the 33rd West Coast Conference on Formal Linguistics (WCCFL 33)*, eds. Kyeong-min Kim, Pocholo Umbal, Trevor Block, Queenie Chan, Tanie Cheng, Kelli Finney, Mara Katz, Sophie Nickel-Thompson, and Lisa Shorten, 1–16. Somerville, MA: Cascadilla Press.

Halle, Morris, and Alec Marantz. 1993. Distributed Morphology and the pieces of inflection. In *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger*, eds. Kenneth Hale and Samuel Jay Keyser, 111–176. Cambridge, MA: MIT Press.

Halle, Morris, and Alec Marantz. 1994. Some key features of Distributed Morphology. In *Papers on Phonology and Morphology*, eds. Andrew Carnie and Heidi Harley, volume 21 of *MIT Working Papers in Linguistics*, 275–288. Cambridge, MA: MITWPL.

Keine, Stefan. 2016. Probes and their horizons. Ph.D. dissertation, University of Massachusetts, Amherst, MA.

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

Sportiche, Dominique. 2005. Division of labor between Merge and Move: Strict locality of selection and apparent reconstruction paradoxes. Ms. UCLA.

Torrence, Harold. 2012. The morpho-syntax of silent *wh*-expressions in Wolof. *Natural Language and Linguistic Theory* 30:1147–1184.

Williams, Edwin. 2003. *Representation theory*. Cambridge, MA: MIT Press.

Williams, Edwin. 2011. *Regimes of Derivation in Syntax and Morphology*. New York: Routledge.

Williams, Edwin. 2013. Generative semantics, generative morphosyntax. *Syntax* 16:77–108.

Williams, Edwin. 2014. Compositionality and derivation. Ms., Princeton University.