

Phases

Early minimalism, ranging from Chomsky (1989) to Chomsky’s (1995) *Minimalist Program* (MP), incorporated a *weakly derivational* approach. The computational system (narrow syntax, C_{HL} ¹) manipulates a selection of lexical items (LI) by means of a step-by-step application of the operations Merge and Move, until Spell-Out occurs. Then, PF and LF are created, the two levels of representation interfacing with the syntax-external modules A-P and C-I, respectively.

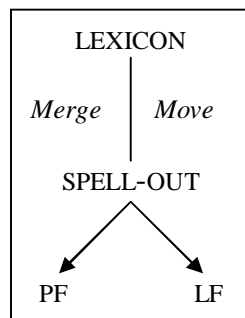


Fig. 1: The Y-Model

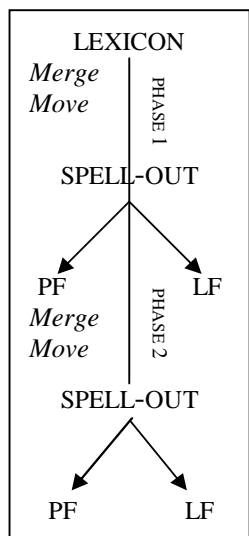


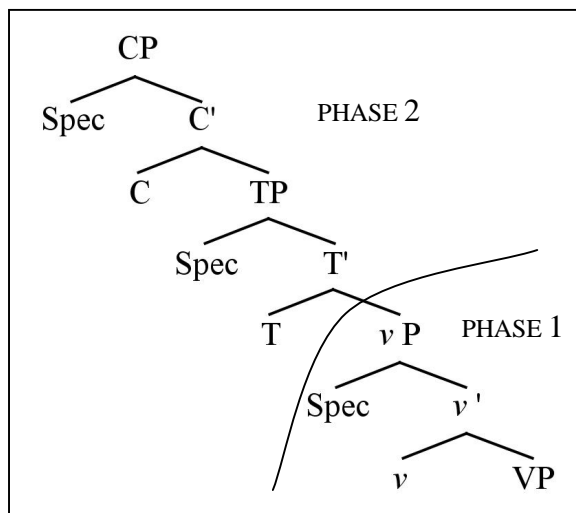
Fig. 2: Multiple Spell-Out Model

Chomsky’s (2000) *Minimalist Inquiries* (MI) sought to reduce *derivational complexity* by chopping the lexical array (LA) up into *sub-arrays*², each feeding C_{HL} to derive a particular phase – a *derivational cycle*. Phases are well-defined chunks of a derivation – vP , CP, and DP (more on the reasons for this choice below) –, each of which, upon completion, is transferred to the interfaces, and thus does no longer bother the computation with its weight. This entails a theory of *Multiple Spell-Out* (cf. Uriagereka 1999).

The syntactic objects that qualify for phases are *transitive v^*Ps* (star’s for transitive), which contain an AGENT or an EXPERIENCER as an external argument (to the exclusion of passive and unaccusative vP), and *CPs*, which are specified for Tense and Force

(Adger’s *Clause Type*)³. Crucially, *defective TPs and VPs are no phases*. The reasoning behind this assumption is for one conceptual (phases represent natural, propositional objects, saturated expressions; v^*Ps represent events, CPs full propositions), but also grammatical (pseudo-clefting is not possible with non-finite raising/ECM-complements, which are TP). CPs are *complete clausal complexes* and v^*Ps are *complete thematic complexes*.

A rather nice conceptual argument for phases concerns the uninterpretability of features: If Spell-Out has to remove uninterpretable features to avoid a crash at the interfaces, it must know which features are interpretable, and



¹ HL = Human Language.

² The LA replaced the notion of *numeration* (NUM) from minimalist frameworks previous to MI (e.g. MP). Technically speaking, a LA is a NUM if it contains more than one occurrence of one and the same lexical item (LI), in which case this item’s index is larger than ‘1’ (cf. DbP:11).

³ DPs are considered phases, too (Chomsky 2005:17f.). Whether PPs are phases or not remains to be investigated.

which are uninterpretable. However, narrow syntax is supposed to be blind as a bat, and thus, would need to be able to *look ahead* (up the tree, to LF and PF) in order to determine the interpretability of a feature. A transfer of derivational chunks to the interfaces remedies this issue, *search space* being reduced to a local domain (a phase).

Empirical support for phases comes from the *EPP-feature on T*: how does C_{HL} decide between attracting a subject DP and merging an expletive *there*? Given the economic preference of *Merge over Move* (Move is more costly than Merge since $Move \approx Copy + Merge$), an insertion of *there* should be expected in every instance, and raising should never be possible. Phases circumvent this issue since lexical sub-arrays can be defined for every cycle.

To get an idea of the technical side of this argument, first consider the following two examples, which illustrate *Merge-over-Move*. They share one and the same numeration, but one derivation yields the ungrammatical (2).

(1) Num = {there₁, T₂, seem₁, to₁, be₁, someone₁, here₁}

- (2) a. There seems to be someone here.
 b. *There seems someone to be here.

Let's take a closer look at the steps of the derivation of (2).

- A. [TP T_[EPP] to be someone here] – Merge T
 B. [TP there T_[EPP] to be someone here] – Merge there & check EPP
 D. [TP T_[EPP] [VP seems [TP there T_[EPP] to be someone here]]] – Merge T
 E. [TP there T_[EPP] [VP seems [TP ~~there~~ T_[EPP] to be someone here]]] – Move there & check EPP

Now consider the derivation of the ungrammatical (2)(1), based on the same numeration, taking another option at step B.

- A. [TP T_[EPP] to be someone here] – Merge T
 B. [TP someone T_[EPP] to be ~~someone~~ here] – Move someone & check EPP
 D. [TP T_[EPP] [VP seems [TP someone T_[EPP] to be ~~someone~~ here]]] – Merge T
 E. [TP there T_[EPP] [VP seems [TP someone T_[EPP] to be ~~someone~~ here]]] – Merge there & check EPP

The derivational step B of (2) violates *Merge-over-Move*, moving *someone* instead of merging the expletive *there* available in the Num, which is why the derivation produces an ill-formed sentence. Defining different sub-arrays for (2), provided the phasehood of vP and CP, can capture this issue *derivationally*:

(3) a. {{C, T}₃ {seem, there, T, to}₂ {be, someone, here}₁}. → There seems to be someone here. – (2)

b. $\{\{C, \text{there}, T\}_3 \{\text{seem}, T, \text{to}\}_2 \{\text{be}, \text{someone}, \text{here}\}_1\} \rightarrow$ *There seems someone to be here. – (2)

c. $\{\{C, T\}_3 \{\text{seem}, T, \text{to}\}_2 \{\text{be}, \text{someone}, \text{here}\}_1\} \rightarrow$ Someone seems to be here. – *w/o expletive*

So far, so good. However, in the examples above, appeal to *Merge-over-Move* was sufficient, without the need to invoke phases? But what of examples like the following (from Boeckx & Grohmann 2005?)

(4) There was a rumour that a man was in the room.

Here, given *one numeration* for the derivation together with the condition of *Merge-over-Move*, the embedded clause ...*that a man was in the room* could never be derived, since the availability of *there* would always *pre-empt raising*.

(5) LA = $\{C_2, \text{there}, T_2, \text{be}_2, a_2, \text{rumour}, \text{that}, \text{man}, \text{in}, \text{the}, \text{room}\}$

Here, *phasal sub-arrays actually do the trick!*

(6) LA = $\{\{C, \text{there}, T\}_4 \{\text{be}, a, \text{rumour}\}_3 \{C, \text{that}, T\}_2 \{\text{be}, a, \text{man}, \text{in}, \text{the}, \text{room}\}_1\}$

Consider the derivation.

- A. $[_{TP} T_{[EPP]} \text{be a man in the room}]$ – Merge T
- B. $[_{TP} a \text{ man was}_{[EPP]} \text{be a man in the room}]$ – Move a man & check EPP
- C. $[_{CP} \text{that a man was}_{[EPP]} \text{be a man in the room}]$ – Merge C/that
- D. $[_{TP} T_{[EPP]} [\text{be a rumour that a man was}_{[EPP]} \text{be a man in the room}]]$ – Merge T
- E. $[_{CP} [_{TP} \text{there was}_{[EPP]} \text{be a rumour that a man was}_{[EPP]} \text{be a man in the room}]]$ – Merge there & check EPP

Dividing Num up into lexical sub-arrays gets rid of the problem: if in the second sub-array $\{\dots\}_2$ no expletive is available, subject raising can take place.

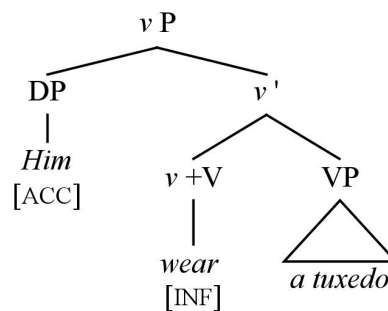
Phases consist of three parts: the phase head H (v/C), its complement ZP (the internal domain), and its *edge* YP (the specifier domain). After completion, the phase is inaccessible to further operations, as formally captured by the *Phase-Impenetrability Condition* (PIC): “In phase α with the head H, the domain of H is not accessible to operations outside α , only H and its *edge* are accessible to such operations” (Chomsky 2000:108).

To ensure classic *successive cyclic movement* (DPs moving up the tree have to make stops in the Specs of other heads), Chomsky allows a phases edge, together with the phase head, to remain accessible even after completion of a given phase (e.g. as intermediate landing site for long-distance movement – an *escape hatch*). This means that once a phase is completed, it is no longer accessible to operations – it is *impenetra-*

ble. After completion, it is Spelled Out. This is where our notoriously unexplanatory *EPP feature* comes into play, which may be more generally re-described as an *edge feature* [EF] (cf. Chomsky 2005)⁴. However, not only is T equipped with [EF] (i.e. the classic subject [EPP]), but also *v* (e.g. Icelandic Object Shift) and C (e.g. *wh*-phrases) may optionally (!) bear one.

Finally, an example from my own M.A. thesis: Progovac (2006) claims that *Adult Root Infinitives* (ARIs) might actually be a first phase (i.e. *v*P) provided with default features (subject case [ACC] and infinitival null suffix [INF]) at the PF interface. (In my view, although being rather minimal in spirit, this analysis is at least problematic in the light of left-peripheral topics in Spanish ARIs.)

(7)



However, unlike the bulk of minimalist theory, phases have spawned critical reactions within the minimalist camp. Thus, Boeckx & Grohmann (2005), for example, critically examine phase theory, essentially identifying it as old wine in new bottles (islands, bounding nodes, successive cyclicity, etc.): “[Phases] simply re-code insights from the past” (ibid.:1). Still, it should be noted that it simply is Chomsky’s phase theory that is most often exposed, for he is the leading figure in mainstream generative linguistics; but there are plenty of similar proposals relying on *phase-like concepts*, differing from Chomsky’s proposal in one or the other relevant aspect (e.g. Uriagereka’s 1999 *cascades*, Grohmann’s 2003 *prolific domains*). Furthermore, there are radically derivational approaches to syntax (cf. e.g. Epstein & Seely 2006; Chomsky’s is *weakly derivational*), which have eliminated any level of representation (LF, PF), to the effect that a derivation is dynamically accessed by the interfaces *at every derivational step* (i.e. the smallest phases ever!).

This squib was supposed to be only a technical outline of the motivation for and the implementation of phases in the minimalist framework. Many interesting aspects are left untouched, of course. For example, it is worth taking a look at defective (i.e. approximately, non-finite) TPs and intransitive *v*P, and how they differ derivationally from their phasal counterparts. In addition, Probe-Goal theory (Agree), which has figured prominently during our SCIMS, is closely associated with the introduction of phases.

⁴ In Chomsky (2000), [EDGE] was construed as a *peripheral feature* [P], in Chomsky (2001, 2004) as a *generalised EPP feature* [EPP], in analogy to the classic T-related [EPP]:

References

- BOECKX, Cedric & Kleanthes GROHMANN (2005): "Putting Phases into Perspective". Ms.
- CHOMSKY, Noam (1989): "Some Notes on Economy of Derivation and Representation". *MIT Working Papers in Linguistics* 10, 43-74.
- (1995). *The Minimalist Program*. Cambridge, MA: MIT Press.
- (2000): "Minimalist Inquiries". In: Roger MARTIN, David MICHAELS & Juan URIAGEREKA (2000). *Step by Step – Essays on Minimalist Syntax in Honor of Howard Lasnik*. Cambridge, MA: MIT Press, 89-155.
- (2001): "Derivation by Phase". in: Michael KENSTOWICZ (ed.) (2001). *Ken Hale: A Life in Language*. Cambridge, MA: MIT Press, 1-52.
- (2004): "Beyond Explanatory Adequacy". In: Adriana BELLETTI (ed.) (2004). *Structures and Beyond*. Oxford: Oxford University Press, 104-131.
- (2005): "Three Factors in Language Design". *Linguistic Inquiry* 36:1, 1-22.
- EPSTEIN, Samuel David & Daniel SEELY (2006). *Derivations in Minimalism*. Cambridge, MA: Cambridge University Press.
- GROHMANN, Kleanthes (2003). *Prolific Domains. On the Anti-Locality of Movement Dependencies*. Amsterdam: John Benjamins.
- URIAGEREKA, Juan (1999): "Multiple Spell-out". In: Samuel D. EPSTEIN and N. HORNSTEIN (eds). *Working Minimalism*. Cambridge, MA: MIT Press.