1. Introduction

Noam Chomsky is responsible for creating the field of research on grammatical theory in its present form, and he has also been material in providing frameworks for syntactic analysis, from early transformational grammar (see Chomsky (1957; 1975)) to current implementations of the minimalist program (see Chomsky (2001; 2008; 2013), Chomsky et al. (2019)). Against the background of these (and other) syntactic frameworks, a huge number of constraints on grammatical dependencies have been proposed over the years, most of them by other scholars, but quite a few of them also by Chomsky himself, from the A-over-A Principle in Chomsky (1964a) to a constraint demanding a minimization of accessible objects in workspaces in Chomsky (2019). Some of these constraints have become famous both within and outside of the field of linguistics, like the Subjacency Condition (Chomsky (1977)), the Empty Category Principle (ECP; Chomsky (1981)), and Principles A, B and C of the theory of binding (Chomsky (1981)). Other constraints initially suggested by Chomsky have a more troubled history, in the sense that they have variously been claimed to be empirically falsified, considered conceptually flawed, or simply consigned to oblivion, for whatever reason. It is a main goal of the present contribution to argue that some of the constraints arguably belonging to this latter group can, upon closer scrutiny, in fact be shown to be at the heart of current syntactic theory, and qualify as “good” (in a sense to be made precise), whereas some constraints in the former group actually qualify as “bad” constraints. More specifically, based exclusively on a consideration of constructions involving displacement (movement) of syntactic categories, I will show that the A-over-A Principle (Chomsky (1964a)), the Superiority Condition (Chomsky (1973)), the Phase Impenetrability Condition (PIC; Chomsky (2001)) and the Strict Cycle Condition (SCC; Chomsky (1973; 1995)) together form the core of a comprehensive, beautiful theory of (relativized and rigid) locality, whereas the Subjacency Condition and the ECP, while certainly improvements over Ross’s (1967) original island constraints or filter constraints as proposed in Chomsky & Lasnik (1977), have no place in current syntactic models.

2. Good Constraints

Research over the last few decades has established a number of requirements that a syntactic constraint should meet in order to qualify as a plausible candidate for inclusion in the faculty of language. For the following discussion, I will adopt the list in (1).

(1) a. Constraints are as simple and general as possible.
   b. Constraints are not complex.
   c. Constraints do not require massive search space.
   d. Constraints do not give rise to redundancies.
   e. Constraints are of type (i) or (ii).
      (i) principles of efficient computation
      (ii) interface conditions

First, (1-a) is really just a manifestation of an uncontroversial requirement for all scientific models: If there are two theories that account for the same set of data, the one is to be preferred that is superior with respect to simplicity, generality, and elegance. From this requirement, it follows that the number of Boolean operations in constraint definitions
should be minimal (cf. Grimshaw (1998)). Furthermore, syntactic constraints should not mention specific information like, e.g., particular category labels, or features characterizing arbitrary subclasses of the linguistic expressions that the constraint holds for. Evidently, construction-specific assumptions (let alone individual lexical items) should be avoided in constraints. From this perspective, a constraint like the "that-trace filter (Chomsky & Lasnik (1977)) to account for complementizer-trace effects as in (2-a), or the for-to filter (Chomsky & Lasnik (1977)) to account for the illformedness of subject-less infinitivals as in (2-b), cannot be good constraints.

(2) a. *Who_1 do you think [CP that t_1 left ] ?
   b. *We want for to win

Second, (1-b) presupposes that constraints on syntactic dependencies can be grouped into various classes determined by their complexity (see Müller & Sternefeld (2001), Graf (2013)). Local derivational constraints apply to individual syntactic operations (like the movement step). Local representational constraints apply to syntactic output representations. Global constraints apply to whole derivations of sentences; they correlate non-adjacent steps in the derivation (cf. Lakoff (1971); Chomsky’s (1981) Projection Principle belongs in this class). Transderivational constraints are even more powerful since they require the construction of a reference set of competing syntactic derivations based on an initial input from which one (or more) derivation is picked out as optimal; this group includes the Fewest Steps condition selecting the derivation with the fewest number of operations that lead to convergence (Chomsky (2001)), the Shortest Paths condition selecting the derivation with the shortest movement paths (Chomsky (1995)), and the Merge over Move condition preferring a step of basic structure-building (external Merge) to a movement step (internal Merge), in a given local domain (the phase); cf. Chomsky (2000), Frampton & Gutmann (1999), Hornstein (2009). Finally, there is a fifth type that can be referred to as translocal constraints. In contrast to transderivational constraints, a translocal constraint compares competing output representations, and picks one as optimal; a canonical constraint of this type is the Avoid Pronoun principle that favours representations with non-overt pronouns (pro, PRO) over representations with overt pronouns (Chomsky (1981), Haegeman (1995)). Given this taxonomy, a natural assumption is that only local constraints can be good constraints.

Third, (1-c) states that in order to determine whether a (local) constraint is violated or not, it should ideally not be necessary to check structures of potentially unbounded size; rather, the smaller the accessible window of a complex syntactic object is for evaluation, the better the constraint satisfies this requirement. This requirement is violated, e.g., by Fox & Pesetsky’s (2005) Cyclic Linearization constraint that demands a preservation of linear precedence relations established in a low syntactic domain (a VP, or a clause) in all higher domains of a potentially complex sentence.

Fourth, according to (1-d), redundancies should be avoided. Thus, if there are two constraints α, β with overlapping effects (such that, e.g., α and β both exclude some ungrammatical structure Γ), one of them should ideally be dispensed with.

Fifth and finally, the requirements in (1-e) have been at the core of minimalist research (see, e.g., Chomsky (2005)). Principles of efficient computation in the sense of (1-e-i) are basically local derivational economy constraints which can be motivated on grammar-external grounds. In contrast, interface conditions in the sense of (1-e-ii) are representational constraints on outputs which are not motivated syntax-externally, but result from demands imposed on syntactic representations by the phonological and semantic interfaces. Thus, according to this far-reaching requirement, constraints that rely on concepts
without independent motivation in terms of either economy or interface conditions are excluded.

Based on the requirements in (1), the next section will discuss Ross’s island constraints, the Subjacency Condition, and the ECP, concluding that none of these constraints qualifies as good (in the technical sense just laid out).

3. Constraints That Do Not Meet The Requirements

3.1. Island Conditions

Given the criteria in (1), none of the island constraints proposed in Ross (1967) (among them the Complex NP Constraint, the Sentential Subject Constraint, the Coordinate Structure Constraint, and the Left Branch Condition) qualifies as a good constraint. As a representative example, consider the Complex NP Constraint (CNPC) in (3), which blocks sentences like (4-a) in English (and many other languages), where wh-movement illegitimately takes place across both a sentence (CP, in current terms) and a noun phrase (DP, under present assumptions) with a lexical head (N), even though wh-movement from a complement CP is possible in principle (see (4-b)), as is wh-movement from an object DP (see (4-c)).

(3) **Complex NP Constraint (CNPC):**
No element contained in a sentence dominated by a noun phrase with a lexical head noun may be moved out of that noun phrase by a transformation.

(4) a. *Which book, did John hear [DP a [NP [N rumour ] [CP that you had read t1 ]]] ?
b. Which book1 do you think [CP that John has read t1 ] ?
c. Who1 did you read [DP a [NP [N book ] [PP about t1 ]]] ?

The CNPC is at variance with (1). For one thing, it is not maximally simple and general since it explicitly mentions category information (sentence:CP, noun phrase:DP, and lexical head noun:N) and thereby encodes a very specific construction. For another, the CNPC requires massive search space spanning the whole structure in (4); in fact, it is a two-node constraint that requires keeping track of a node crossed earlier. And finally, it is hard to see how (3) could be viewed as a principle of efficient computation, or as an interface constraint. Essentially the same conclusions can be drawn for the other constraints on movement dependencies in Ross (1967); but also for similar types of constraints like Chomsky’s (1964a) Wh-Island Condition (according to which movement must not take place from indirect questions) and Chomsky’s (1973) Tensed-S Condition and Specified Subject Condition, all of which describe specific constructions and thus wear their violations of (1-a) (and also of (1-e)) on their sleeves. And it goes without saying that the same kinds of consequences arise for abominations like Zaenen & Pinkham’s (1976) *Post-Sentential Subject Extraction Constraint* or Kuno’s (1973) *Clause Non-final Incomplete Constituent Constraint*. As noted by Chomsky (1982, 74-76), Ross’s island constraints and several other constraints of this general type have been immensely important for the development of the field, but only insofar as they provide a “taxonomy of properties” that makes it possible to “rethink conditions, to see whether one could deal with the [...] phenomena [...] in a unified way” (Chomsky (2002, 129-130)) and to “explain [...] such constraints in terms of deeper and more natural principles from which their effects could be deduced” (Chomsky (1986b, 71-72)), and not as candidates for belonging to the faculty of language in their own right.

3.2. The Empty Category Principle

The Empty Category Principle (ECP) was originally proposed in Chomsky (1981) (with refinements in Aoun, Hornstein & Sportiche (1981), Chomsky (1986a), Lasnik & Saito (1984;
1992), and other work). It is, without any doubt, one of the most successful and widely discussed constraints on syntactic dependencies ever.\(^1\) It is a constraint on *traces*, which was made possible by the introduction of the trace theory of movement in the previous decade (cf. Wasow (1972), Chomsky (1973, 266-269), Fiengo (1977)); see (5).

\[(5)\] **Empty Category Principle (ECP):**

Every trace \(t\) is marked \([+\gamma]\).

A trace \(t\) is marked \([+\gamma]\) iff it is properly governed.

A trace \(t\) is properly governed iff (a) or (b) holds.

\[\text{a. t is antecedent-governed.}\]

A trace \(\beta\) is antecedent-governed iff there is no barrier separating \(\beta\) from its immediate chain antecedent \(\alpha\) (and if there is no \(wh\)-phrase or overt complementizer in the \(C\) domain that intervenes between \(\alpha\) and \(\beta\)).

\[\text{b. t is lexically governed.}\]

A trace \(\beta\) is lexically governed iff it is in the \(m\)-command domain of a lexical category \(\alpha\), and no barrier separates \(\beta\) from \(\alpha\).

The ECP is a local representational constraint applying at the level of LF (after abstract operations like \(wh\)-movement of in-situ \(wh\)-phrases and deletion of that complementizers and of intermediate traces of arguments have applied). It demands that traces left by movement have to be marked \([+\gamma]\); a trace can be marked \([+\gamma]\) only if it is properly governed; and it can be properly governed either via antecedent-governance, or via lexical government. Given certain assumptions (in particular, the additional stipulation in brackets in (5-a), which is necessitated by the move from a single COMP position in Chomsky (1981) to a specifier head configuration SpecC-C in Chomsky (1986a)), the ECP can account for complementizer-trace effects with subject movement as in (2-a) (see (6-a) vs. (6-b)), which do not show up with object movement because object traces, unlike subject traces, are lexically governed.

\[(6)\] a. *Who\(_1\) do you think \([CP t'_1 [C' [C that ] ] [TP t_1 left ] ]]\) ?

b. **Who\(_1\) do you think \([CP t'_1 [C' [C ] ] [TP t_1 left ] ]]\) ?

The ECP in (5) also correctly predicts that the complementizer-trace effect disappears if the complementizer intervenes at a later stage of the derivation, and the original subject trace can be antecedent-governed by an intermediate trace, as in (7-a); this latter intermediate trace cannot be marked \([+\gamma]\), but, not being a final member of a \(wh\)-movement chain, it can be deleted before the ECP is checked at LF. Similarly, assuming that adjunct traces can only be \([\gamma]\)-marked at LF (whereas argument traces are \([\gamma]\)-marked as soon as they arise in the derivation), the non-occurrence of a complementizer-trace effect in (7-b) can be accounted for by postulating complementizer deletion before LF. (Still, in other environments where adjunct movement has to cross a barrier, such movement is correctly excluded.)

\[(7)\] a. **Who\(_1\) do you believe \([CP that Mary thinks [CP t'_1 [C' [C that ] ] [TP t_1 left ] ]]\) ?

b. **Why\(_1\) do you think \([CP t'_1 [C' [C that ] ] [TP Mary left t_1 ] ]]\) ?

The ECP also covers many so-called superiority effects, as in (8-b) vs. (8-a), assuming that there is LF \(wh\)-movement of the in-situ \(wh\)-phrase in multiple questions in English (which gives rise to a subject LF trace \(t_1\) in (8-b) that is not properly governed, in contrast to the

\(^1\) An extended google scholar search in early 2020 yields almost 7000 hits for articles or books containing “ECP” and “Chomsky”, whereas there are hardly more than 1000 hits for work containing “A-over-A” and “Chomsky”.

4
object LF trace $t_2$ in (8-a)).

(8)  
   a.  I wonder [\text{CP who}_1 \text{ t}_1 \text{ bought what}_2 ]  
   b. *I wonder [\text{CP what}_2 \text{ who}_1 \text{ bought t}_2 ]

In addition, subsequent research has identified many more ECP effects of diverse provenance, and in many different languages, exhibiting asymmetries between subject and object movement on the one hand, and between argument and adjunct movement on the other. However, despite its considerable empirical coverage, the ECP is not a good constraint. First, as should be clear from (5), the ECP has a highly complex inner structure, involving a huge number of auxiliary concepts and a disjunction at its core (antecedent-government or lexical government); so it certainly does not look simple and general. This conclusion is reinforced by the fact that the constraint talks about a specific type of empty category (traces), rather than empty categories in general (including pro, PRO). And second, the ECP cannot naturally be conceived of either as a principle contributing to efficient computation, or a condition imposed by the LF interface (the fact that it is formulated as an LF constraint notwithstanding). A systematic account of ECP effects in terms of constraints that meet the requirements in (1) is still outstanding.\(^2\)

3.3. The Subjacency Condition

Another constraint that has been hugely successful over the years is the Subjacency Condition (see Chomsky (1977; 1986a)). It blocks movement operations which cross two bounding nodes (DP and TP in English, in slightly anachronistic terminology).

(9)  
   \text{Subjacency Condition:}  
   \text{In a structure } \alpha \ldots [\beta \ldots [\gamma \ldots \delta \ldots ] \ldots ] \ldots , \text{ movement of } \delta \text{ to } \alpha \text{ cannot apply if } \beta \text{ and } \gamma \text{ are bounding nodes.}

The Subjacency Condition is of particular interest in the present context since it derives effects that had earlier been attributed to other, less general constraints, like the CNPC or the Wh-Island Condition. Thus, (4-a) can be excluded by the Subjacency Condition: After an initial movement step to the embedded SpecC position, the wh-phrase has to cross both the object DP and the matrix TP on its way to the final matrix SpecC position.\(^3\) Furthermore, wh-islands as in (10) in English can be derived if one assumes that the embedded wh-phrase blocks the use of an intermediate SpecC position by the matrix wh-phrase, which must therefore cross two TPs in one fell swoop.

(10) *How\(_1\) does [\text{TP}_3 \text{ she know } [\text{CP [DP}_2 \text{ which car }] [\text{TP}_4 \text{ Mary fixed t}_2 \text{ t}_1 ]]] ?

Based on the observation that Italian behaves like English with respect to CNPC environments but is somewhat more liberal with respect to wh-islands, Rizzi (1982) suggested that bounding nodes are subject to \text{parametrization}, with DP and CP (rather than TP) the relevant categories in Italian. This may have been the first instance of genuine parametrization of a constraint on syntactic dependencies, and this strategy was generally viewed as convincing at the time (see, e.g., Chomsky (1982, 45)). However, there are several problems

\(^2\) A general \text{Anti-Locality Constraint} (see Bošković (1994; 2014), Grohmann (2000; 2003), Grohmann \& Nevins (2004), Abels (2003; 2012), Erolwine (2016), Pesetsky (2016)) can block subject movement as too short where object movement in the same context is sufficiently non-local, and might thus ultimately be a candidate for this role.

\(^3\) As noted by Riemsdijk \& Williams (1986), this necessitates some additional reanalysis operation affecting the DP in (4-c), so as to ensure that the Subjacency Condition is not violated here.
with this view. For instance, the movement types underlying the conflicting evidence from Italian and English are different (relativization vs. wh-movement); Italian wh-movement constructions that closely resemble constructions like those in (10) are also not well formed (as noted already in Rizzi (1982)). Furthermore, some English extractions across wh-islands are often not judged to be ill formed after all, e.g., if the embedded clause is non-finite; see Grimshaw (1986) and Chomsky (1986a). In addition to this, there are empirical problems raised by the conception of the Subjacency Condition as a two-node constraint. The prediction here is that any movement type that has its ultimate landing site between the two bounding nodes, in a position that is not SpecC, should ceteris paribus be able to circumvent CNPC or wh-island effects; this simply does not seem to be the case (Müller (2011)).

Perhaps most importantly in the present context, from the point of view of (1), the Subjacency Condition does not emerge as a good constraint. It is not maximally simple and general since it involves construction-specific assumptions (with respect to the parametrized bounding nodes). Being a two-node condition, it requires massive search space (and keeping track of earlier bounding nodes crossed in the course of a single movement step by some sort of counter). It does not appear to be an interface condition (in fact, based on evidence from extraction from ECM constructions where CP deletion counter-bleeds wh-movement via SpecC, Chomsky (1981) argues that the Subjacency Condition must be a derivational constraint, not a representational one); and in its reliance on counting bounding nodes, it would seem to also fail the efficient computation criterion clearly. In view of the many empirical and conceptual problems raised by the Subjacency Condition, Riemsdijk (1978) and Koster (1978; 1986) have argued for simpler, single-node, category-neutral constraints that can be viewed as precursors of the PIC; see below.

4. Constraints That Meet The Requirements

In contrast to the constraints discussed in the previous section, there are two pairs of constraints introduced by Noam Chomsky which are at the very heart of current approaches to locality in syntactic dependencies: on the one hand, the A-over-A Principle and the Superiority Condition (relativized locality), and on the other hand, the Phase Impenetrability Condition and the Strict Cycle Condition (rigid locality). I address the two concepts in turn.

4.1. Relativized Locality Constraints

4.1.1. The A-over-A Principle

The A-over-A Principle is arguably the first substantial locality constraint on grammatical dependencies that has ever been proposed; see Chomsky (1964a,b); the formulation in (11) essentially follows Chomsky (1973, 235) (also cf. Ross (1967, 10), Gazdar (1982, 176-177), Riemsdijk & Williams (1986, 20) for slightly different formulations).

(11) A-over-A Principle:

If a transformation applies to a structure or the form $[\alpha \ldots [A \ldots [A \ldots] \ldots] \ldots]$, where $\alpha$ is a cyclic node, then it must be so interpreted as to apply to the higher phrase of the type $A$.

The standard interpretation (and also the original one adopted in Chomsky (1964b)) is that “$A$” in (11) is a variable over category features, i.e., labels. The constraint then states that movement of some category $\gamma$ to some landing site $\Delta$ is impossible if there another category $\beta$ with the same category feature that dominates $\gamma$ (in this case, $\beta$ is closer to $\Delta$ than $\gamma$ because the path from $\beta$ to $\Delta$ is necessarily shorter than the path from $\gamma$ to $\Delta$). The A-over-A Principle is a local derivational constraint. It predicts that, within a given cyclic domain,
a DP cannot be extracted from another DP; a VP cannot be extracted from a higher VP, and so on. As such, the constraint makes a number of non-trivial correct predictions; however, as pointed out by Ross (1967), in the (label-based) interpretation standardly adopted, the A-over-A Principle is both too strong and too weak. The second problem may make the A-over-A Principle look less plausible but is not fatal (after all, the data that the A-over-A principle cannot exclude even though one might intuitively think that it should – like, e.g., cases of illicit AP extraction from DP, as in *How intelligent do you have a sister?* – can always be excluded by some other constraint(s) in the grammar). The first problem is potentially severe, though. Examples like (4-c) show that a DP can in principle be moved out of another DP. Examples like those in (12-ab) (from English and German, respectively) indicate that a VP can undergo topicalization across another VP, and that a CP can be topicalized over another CP that initially dominates it.

(12)  

- a. \[[\text{VP}_2 \text{ Give a book to Mary }] \text{ I don’t } [\text{VP}_1 \text{ think } [\text{CP } \text{ that she will } t_2 ]]\]
- b. \[[\text{CP}_2 \text{ Dass er nett ist } ] \text{ hat Fritz behauptet } [\text{CP}_1 \text{ dass Maria gesagt hat } t_2 ]\]

The wellformedness of these kinds of examples strongly suggests that the A-over-A Principle cannot be maintained in the interpretation given to “A” so far, viz., as a category label. However, following Ross (1967), the vast majority of the field seems to have reached the much more far-reaching conclusion that the A-over-A Principle should be abandoned in toto. In line with this, most textbooks on syntactic theory do not even mention this constraint anymore. Where it shows up, it is typically characterized as having been falsified by Ross (see, e.g., Riemsdijk & Williams, 1986). Very few textbooks introduce the constraint from a more neutral point of view (see Freidin for one exception), and even fewer have something positive to say about it (but cf. Hornstein et al., 2005, who mention the A-over-A Principle in passing).4

However, the obvious alternative to abandoning the constraint is of course to revise the assumption that “A” in (11) stands for a category label. As a matter of fact, this is exactly what Chomsky (1973, 235) does: He points out that it is possible to assume that the A-over-A Principle in (11) “does not establish an absolute prohibition against transformations that extract a phrase of type A from a more inclusive phrase of type A. Rather, it states that if a transformational rule is nonspecific with respect to the configuration defined, it will be interpreted in such a way as to satisfy the condition.” As Chomsky also notes, “alternatively, one might interpret the A-over-A constraint as legislating against any rule that extracts a phrase of type A from a more inclusive phrase A,” which is the standard understanding facing empirical counter-evidence. Importantly, Chomsky’s (1973) actual conclusion already is that “the former interpretation […] is perhaps more natural”, and that he “will adopt it tentatively here, for this and other conditions to be discussed.” What is more, this version of the A-over-A Principle is also adopted by Bresnan (1976): The item that can be subjected to a given transformation should be as inclusive as possible, independently of category labels (whereas an even more inclusive item that cannot be subjected to the same transformation because it does not fit its structural description can be ignored, even if it shares the same category label).5 Thus, under the revised understanding of “A” in (11) as the feature that is involved in the operation, the A-over-A Principle predicts that a wh-phrase cannot be

---

4 In view of Ross’s counter-arguments, Postal (2012) takes this constraint to be a “failed principle”, a “bankrupt idea”, and criticizes Chomsky for not abandoning the A-over-A Principle after Ross (1967). As we will see momentarily, both Postal’s assessment of the status of the A-over-A Principle and his criticism of Chomsky based on it are misguided.

5 Here is Bresnan’s formulation:
moved out of a wh-phrase (but can move out of a non-wh phrase), a topic-marked phrase cannot move out of another topic-marked phrase, and so on. Let us now adopt a more contemporary approach according to which all syntactic operations are driven by designated features (cf. Chomsky (1995; 2001), Pesetsky & Torrego (2006), Urk (2015), Collins & Stabler (2016) and Georgi (2017), among many others). There are structure-building features (rendered here as \( *[F] \)) that trigger movement and basic structure-building (Merge), and there are probe features (rendered as \( *[F+] \)) that trigger Agree (cf. Assmann et al. (2015)). As far as movement is concerned, the A-over-A Principle can then be given the reformulation in (13), which is completely faithful to the reinterpretation of (11) along the lines of Chomsky (1973) and Bresnan (1976).

(13) \textit{A-over-A Principle} (updated version):

In a structure \( \alpha [F+] \ldots [ \ldots \beta [F] \ldots \gamma [F] \ldots ] \ldots \), \( \gamma \) cannot move to \([F] \) if \( \beta \) dominates \( \gamma \).

Incidentally, this version of the A-over-A Principle corresponds to a relativized locality constraint that has widely been proposed in the more recent literature (cf. Koizumi (1995), Fukui (1997), Kitahara (1997), Müller (1998), Sauerland (1999), Fitzpatrick (2002), Vicente (2007), Heck (2008), and Pesetsky (2013)), typically without reference to either Chomsky (1973) or Bresnan (1976). (13) provides a straightforward explanation for an otherwise mysterious restriction on the movement of remnant categories (i.e., categories from which extraction has taken place): Remnant movement of a certain type is impossible if the movement operation that has created the remnant is of the same type (cf. Takano (1994), Müller (2014)). This effect is shown in (14-a) for DP scrambling from VP co-occurring with remnant VP scrambling in German (where VP topicalization would be perfect in the same context), in (14-b) for DP scrambling from CP co-occurring with remnant CP scrambling in Japanese, and in (14-c) for wh-movement from a wh-phrase co-occurring with remnant wh-movement of that latter, more inclusive wh-phrase in English (which leads to stronger illformedness than one would expect if only a typical wh-island effect were involved; see above).

(14) a. \[ \text{dass } [\text{ VP}_2, t_1 \text{ zu lesen }] \] \[ [\text{ DP}_1, \text{ das Buch }] \text{ keiner } t_2 \text{ versucht hat that to read the book, no-one tried has} \]
b. \[ *[\text{ CP}_2, \text{ Mary-ga } t_1 \text{ yonda-to }] \] \[ [\text{ DP}_1, \text{ sono hon-o }] \text{ John-ga } t_2 \text{ itta (koto) Mary-nom read-COMP that book-nom John-nom said fact} \]
c. \[ *[\text{ DP}_2, \text{ Which picture of } t_1 \] \text{ do you wonder [CP who1 she likes } t_2 ] \]

Given that scrambling and wh-movement involve designated features on the attracting head and the moved item (like \([*\Sigma ]/[\Sigma ]\) and \([*\text{wh }]/[\text{wh }]\), respectively), the A-over-A Principle requires \( \text{XP}_2 \) to move first in (14-abc), so that subsequent movement of \( \text{XP}_1 \) from the \( \text{XP}_2 \) in the derived position to a lower position will have to violate a number of other constraints (in particular, the Condition on Extraction Domain and the Strict Cycle Condition; see below).

(i) \textit{A-over-A Principle} (Bresnan’s revision):

No transformation \( T \) can apply to a structure \( \phi \) under a proper analysis \( \pi \) unless \( \pi \) is a maximal proper analysis of \( \phi \) for \( T \).

Simplifying a bit, this means that a transformation can only apply to a given structure if the size of the target predicate (i.e., the item that is supposed to undergo movement in the case of movement transformations) is maximal vis-à-vis the demands imposed by the structural condition of the transformation.

\* However, see also Chomsky et al. (2019), Safir (2019), and further literature cited in these articles for the opposite view, essentially the “Affect-\( \alpha \)” view of standard government-and-binding theory.
Finally, note that this version of the A-over-A Principle meets all the requirements in (1). It is simple, general, and not complex. It is compatible with a larger search space, but it does not need it to be properly evaluated in any given derivation. And since it is essentially a minimality requirement that brings about a selection of the closest matching item, it can plausibly be viewed as a constraint contributing to efficient computation.

4.1.2. The Superiority Condition

The original formulation of the Superiority Condition is given in Chomsky (1973, 246).

(15) **Superiority Condition:**

No rule can involve, X, Y in the structure

\[ \ldots X \ldots [\alpha \ldots Z \ldots \text{-WYZ} \ldots ] \ldots \]

where the rule applies ambiguously to Z and Y and Z is superior to Y.

In current terms, a category is superior to another one if it asymmetrically c-commands it. Furthermore, the condition that a rule applies ambiguously to two items captures exactly the same scenario that we have seen with the revised A-over-A Principle: Both items (Z and Y in (15)) can in principle be affected by the (movement) operation. (15) then demands that the higher, c-commanded one is to be affected by the transformation (or to be affected first if the transformation can apply more than once). Like the A-over-A Principle, the Superiority Condition can be given a fully faithful update, as in (16).

(16) **Superiority Condition** (updated version):

In a structure \( \alpha_1 [\bullet F] \ldots [ \ldots \beta [F] \ldots \gamma [F] \ldots ] \ldots \), \( \gamma \) cannot move to \( [\bullet F] \) if \( \beta \) c-commands \( \gamma \).

The Superiority Condition straightforwardly accounts for wh-movement asymmetries in multiple questions, as in (8) in English, without recourse to either the ECP or abstract LF movement: A subject wh-phrase c-commands an object wh-phrase and thus blocks its movement. Furthermore, unlike the ECP, the Superiority Condition accounts for similar intervention effects with two wh-objects, as in (17) (see Pesetsky (1982, 602ff)).

(17) a. Whom\(_1\) did John persuade t\(_1\) [CP to visit whom\(_2\) ] ?

b. *Whom\(_2\) did John persuade whom\(_1\) [CP to visit t\(_2\) ] ?

The constraint can also cover wh-island effects, as in (10), where wh-movement takes place from a wh-island. It has nothing to say about topicalization from wh-islands, given that different movement-related features are involved ([top] vs. [wh]). However, as noted by Fanselow (1987), the relevant examples are in fact well formed in German (unlike cases of wh-movement from wh-clauses); so this is a welcome consequence.

Like the A-over-A Principle, the Superiority Condition qualifies as a good constraint. First, it is simple and general. It is worth pointing out that, although it is usually treated as a constraint purely on wh-movement, there is nothing in the formulation of the Superiority Condition that would support the idea that it is confined to wh-environments.\(^7\) Second, it is a local derivational constraint, and hence not complex. Third, it is compatible with, but does not presuppose, a large search space. And fourth, it implements a minimality requirement that lends itself to a classification as a principle of efficient computation.

\(^7\) Indeed, explicitly general minimality constraints that are virtually indistinguishable from (16) have been proposed subsequently, by Fanselow (1990; 1991) and by Ferguson (1993) and Ferguson & Groat (1994). Also note in this context that Rizzi’s (1990) Relativized Minimality is very different both conceptually and in its empirical consequences; see Müller (2011, 61-64).
Evidently, the A-over-A Principle in (13) and the Superiority Condition in (16) are two sides of the same coin; both constraints demand movement of the closest item, and the only difference is that closeness is defined in terms of domination in the first case, and in terms of c-command in the second. It thus makes sense to subsume the two constraints under a single, generalized Minimal Link Condition (MLC; see Chomsky (1995; 2001; 2008), and Takano (1994), Kitahara (1997), Müller (1998), Fitzpatrick (2002), Rackowski & Richards (2005), and Medeiros (2008), among many others).

4.2. Rigid Locality Constraints

The two constraints discussed in the previous section are relativized locality constraints since movement options for some item γ are not restricted to a fixed domain; rather, they are determined by the presence or absence of an intervening item β that is closer to the landing site. In contrast, rigid locality constraints on syntactic dependencies identify intrinsic, invariant properties of domains that make them (in-) accessible for syntactic operations.

4.2.1. The Strict Cycle Condition

The Strict Cycle Condition in (18) has been proposed for syntactic dependencies in Chomsky (1973, 243).

(18) **Strict Cycle Condition (SCC):**

No rule can apply to a domain dominated by a cyclic node A in such a way as to affect solely a proper subdomain of A dominated by a node B which is also a cyclic node.

The Strict Cycle Condition is arguably the single most important constraint in every derivational theory of grammar; see Chomsky (1995; 2001; 2008), Jacobson & Neubauer (1974), Pullum & Soames (1979), Collins (1997), Kitahara (1997), Bošković & Lasnik (1999), Freidin (1999), Müller (2011), Abels (2012), Collins & Stabler (2016), and Gallego (2020), among many others, for discussion and alternative formulations. This constraint ensures that once the derivation has created a certain structure, it is impossible to exclusively modify an embedded part of this structure by carrying out counter-cyclic operations. The degree of strictness of the constraint depends on how “cyclic node” is defined. According to the most restrictive, and most interesting, concept, every projection is a cyclic node. On this view, every legitimate operation must involve the current root. Among many other things, the Strict Cycle Condition is indispensable in a derivational account of freezing effects (Wexler & Culicover (1980), Browning (1991), Lohndal (2010)), as in (19-a) (with DP raising to subject in the passive interacting with wh-movement from DP in English) and (19-b) (with VP topicalization interacting with wh-movement from VP in German).

(19) a. *Who_{1} was [DP_{2} a picture of t_{1} ] painted t_{2} by Mary ?
   b. *Was_{1} _denkst du [CP [VP_{2} t_{1} gelesen ] hat keiner ] t_{2} ] ?

Given that extraction from XP is possible only if XP is a complement (cf. the Condition on Extraction Domain; see Huang (1982), Chomsky (1986a), Cinque (1990)), the illformedness of (19-ab) can be derived if movement of DP_{2} and VP_{2} precedes extraction of DP_{1} (because DP_{2}/VP_{2} occupies a specifier when DP_{1} extraction takes place); but the reverse, counter-

---

8 Versions of the constraint include the Extension Condition in Chomsky (1995), and the No Tampering Condition in Chomsky (2008).
cyclic sequence of movement operations where DP\textsubscript{1} extraction takes place when DP\textsubscript{2}/VP\textsubscript{2} is still in situ, in a complement position, must also be excluded, and this is accomplished by the Strict Cycle Condition. Similarly, if it is assumed that the embedded wh-phrase in SpecC is responsible for the wh-island effect in a sentence like (10), the Strict Cycle Condition is needed to ensure that it is already present when movement of the other wh-phrase to the matrix domain takes place.

More generally, as observed by McCawley (1984; 1998), the Strict Cycle Condition predicts orders among syntactic operations. Given a very narrow concept of cyclic domains (the minimal projection), the constraint can thus resolve what would otherwise be indeterminacies in rule application. As a case in point, consider the dual role of (transitive) v heads. Under standard assumptions, v introduces the external argument (i.e., it has a structure-building feature [•D•] triggering Merge), and it can assign accusative case as an instance of Agree (i.e., there is a valued probe feature [∗acc∗] in addition). It is a priori unclear which operation has to take place first (with radically different consequences potentially arising; see Müller (2004)); but the Strict Cycle Condition working on minimal cyclic domains resolves the order in favour of Agree preceding Merge.

Sometimes a weaker version of the Strict Cycle Condition is adopted. For instance, Safir (2019, 292) proposes a Peak Novelty Condition that permits operations which are not massively counter-cyclic (i.e., reasonably close to the current root domain). Now and then, the concept of Featural Cyclicity has been pursued (cf. Richards (2001) and Preminger (2018), among others); here the only requirement is that active features that can trigger operations must do so as soon as possible; i.e., they cannot wait and become embedded by further structure-building. From a historical perspective, Featural Cyclicity is similar to the Cyclic Principle of classic transformational grammar (cf., e.g., Perlmutt & Soames (1979)): When two operations can be carried out, where one applies to the cyclic domain D\textsubscript{x} and the other applies to the cyclic domain D\textsubscript{x−1} included in D\textsubscript{x}, then the latter is applied first. Note that the Cyclic Principle and the Strict Cycle Condition are far from the same thing: A rule applying in a higher cycle may change the context for the application of a rule in a lower cycle, such that, e.g., the second rule can now apply (as part of the higher cycle). This interaction is excluded by the Strict Cycle Condition, but not by the Cyclic Principle.

The different predictions of the Strict Cycle Condition on the one hand and Featural Cyclicity and the Cyclic Principle on the other are striking. Numerous counter-cyclic operations that have been proposed are incompatible with the former constraint but compatible with the latter ones, among them late adjunction of adjuncts (Lebeaux (1988), Epstein et al. (1998) vs. Chomsky (2004)); wholesale later merger (Takahashi & Hulsey (2009), Urk (2015)); tucking in (Richards (2001)); head movement as adjunction (cf. Chomsky (1995), Matrushevsky (2006), Georgi & Müller (2010) for alternatives); and feature inheritance (Chomsky (2008), Richards (2007)).

Like the A-over-A Principle and the Superiority Condition, the Strict Cycle Condition qualifies as a good constraint according to (1). It is simple and not restricted to particular categories; it is not complex; it brings about a massive reduction of accessible space for items that can trigger operations; and by doing this and ensuring maximal stability of linguistic objects created in the course of the derivation, it looks like a good candidate for a principle of efficient computation.

4.2.2. The Phase Impenetrability Condition

Another fundamental constraint imposing rigid locality on operations is the Phase Impenetrability Condition (PIC) introduced in Chomsky (2000, 108) and Chomsky (2001, 13), and adopted in much subsequent minimalist work. The more restrictive version of the two PICs
Chomsky considers is given in (20).9

(20) **Phase Impenetrability Condition** (PIC):

The domain of a head X of a phase XP is not accessible to operations outside XP; only X and its edge are accessible to such operations.

The edge of a head X is the residue outside of X'; it comprises specifiers of X (and adjuncts to XP). Chomsky (2000; 2001; 2008) proposes that phases are the CP and vP, which share the property of being “propositional categories.” It follows from the PIC that all material that needs to be accessible outside the current phase must occur in the edge domain of the phase or on the phase head. For non-local movement operations ending beyond the current phase, this requires an intermediate step to a specifier of the phase head; for non-local agreement operations, the PIC may suggest a concept like cyclic Agree (Legate (2005)) or piggybacking of agreement on movement (Polinsky & Potsdam (2001)).10

Crucially, the PIC can act as a locality constraint on movement and have substantial effects here, but only if it is accompanied by a theory of what can, and what cannot, undergo movement to the phase edge. Pertinent suggestions include specific restrictions on the insertion of edge features as triggers for intermediate movement steps (see Müller (2011)) and, more recently, an Anti-Locality Constraint that prohibits items from reaching the phase edge that are too close to it (see the references in footnote 2).

Like the Strict Cycle Condition, the PIC qualifies as a good constraint in the sense of (1). It is a local constraint, hence, not complex. It clearly leads to a massive reduction of search space. It was explicitly introduced by Chomsky (2000; 2001) as a principle of efficient computation. And it is also simple and general. In particular, it can be observed that even though it singles out CPs and vPs as phases, this does not have to be stipulated but follows because these two categories form a natural class defined by the independently motivated feature “propositional”.11

As with the two relativized locality constraints, the two rigid locality constraints highlighted in this section form two sides of the same coin. The Strict Cycle Condition and the Phase Impenetrability Condition both reduce search space, but whereas the Strict Cycle Condition imposes restrictions on the position of an item that triggers an operation ([*F*], [*F*]), the PIC imposes restrictions on the position of an item that is sought by the trigger ([F]). The small size of the accessible window in syntactic derivations resulting from these two constraints, and the constant change of this window in the course of the derivation, are illustrated in (21) (where phases are underlined).

(21) **Search space reduction:**

\[
\begin{align*}
\text{SCC} & : \\
\text{PIC} & \\underline{\text{YP...Y}} \underline{\text{XP...[X:X}}} \underline{\text{WP...W}} \underline{\text{UP...U...]]]]}}
\end{align*}
\]

9 Koster’s (1978) **Bounding Condition** and Riemsdijk’s (1978) **Head Constraint** can be viewed as predecessors of the PIC.

10 Alternatively, Chomsky et al. (2019) suggest that the complement of a phase head might remain accessible throughout; it cannot be modified, but goal features in this domain can be accessed for Agree relations.

11 Basically the same conclusion holds if all XPs are viewed as phases (Müller (2011)); arguably, some additional effort is required from this perspective to motivate proposals where only CP is a phase (Keine (2016)); where DP is a phase in addition to CP and vP (Matushansky (2005), Kramer (2007)); where TP can be a phase (Richards (2011)); or where phase status is more flexible (Bobaljik & Wurmbrand (2003), Marušić (2005), den Dikken (2007), Gallego (2007)).
To conclude, Chomsky’s A-over-A Principle, Superiority Condition, Strict Cycle Condition and Phase Impenetrability Condition are four constraints on grammatical dependencies that have widely been adopted in recent work on syntax, and that can be classified as good constraints according to natural criteria related to simplicity, generality, complexity, domain dependence, and efficient computation. Notice, though, that I have so far been silent on the criterion of redundancy avoidance (see (1-d)). And indeed, as noted by Chomsky (2001, 47, fn. 52), “the effect on the MLC is limited under the PIC, which bars ‘deep search’ by the probe.” Thus, the relativized locality constraints can only become active in the relatively small portions of structure permitted by the PIC; this may eventually suggest an option of further generalization.

References


Chomsky, Noam (2019): Lectures. UCLA & MIT.


Inquiry 15, 235–289.
Müller, Gereon (2011): Constraints on Displacement. A Phase-Based Approach. Vol. 7 of Language Faculty and Beyond, Benjamins, Amsterdam.
Rackowski, Andrea & Norvin Richards (2005): Phase Edge and Extraction, Linguistic Inquiry