# Principles of Universal Grammar
# and the Minimalist Program

David Adger and Ian Roberts[1]
April 2021

## 1. What are principles of Universal Grammar?

A principle of a theory is simply a statement that specifies the relationships that obtain between the concepts of that theory; it is an attempt to say something about the natural object that the theory is a theory of. Principles of Universal Grammar are, then, theoretical statements about the nature of Universal Grammar. Universal Grammar itself is a cognitive system, (partly) responsible for the structure and content of attained grammars. It is (close to) identical across all human beings, the initial cognitive state that holds before an individual is exposed to linguistic experiences. It is an organization of the human mind, specific to our capacity to acquire and use language. In essence, Universal Grammar delimits what a possible humanly attainable grammar is.

The grammars that are acquired by individuals (known as I-languages) are cognitive systems, storing information, determined by both Universal Grammar and experience. Since I-languages store information, they must interact with other systems that put that information to use. At least two such systems are crucial: motor systems that are involved in producing and comprehending utterances (irrespective of whether the medium is signed or spoken) and systems that connect language to thought and are involved in planning, social cognition, conceptual structure, reasoning, etc. These two systems are usually called the Articulatory-Perceptual, and Conceptual-Intentional interfaces.

Putting all this together, principles of Universal Grammar make theoretical claims about the content of Universal Grammar, its structure, and its interfaces.

In early versions of transformational generative grammar (Chomsky 1957, 1965), I-languages, then just called grammars, are constituted by rules, which are learned by exposure to linguistic data. Principles of Universal Grammar in this framework are assumed to be constraints on the rules of the particular grammar that an individual acquires. Universal Grammar provides a format for rule systems, and restricts how rules operate. There were thought to be two syntactic rule systems: Phrase Structure rules, which build structures, and transformational rules, which manipulate structures. The apparent complexity of the syntactic rule systems for particular I-languages, it was assumed, makes such systems unlearnable without some set of constraints, specific to language, that guide the learning process; one such constraint was the evaluation metric, by which acquirers were able to select the optimal rule system from among those which could generate the sentences they were exposed to.

During the 1970s and the 1980s, the role of rules in syntax was radically reduced, with the principles of Universal Grammar taking on more of the explanatory burden. Phrase Structure rules, which had provided a way of generating the base structures of transformational grammar, were replaced by the general schema of X-bar theory. Ross's (1967) discovery of constraints on the functioning of particular transformational rules was followed by the factoring out of commonalities in these constraints into subjacency, a major principle of

[1] Thanks to Coppe van Urk for comments.

Universal Grammar. This line of research, starting from Chomsky (1973), led to the reduction of the transformational component to a general rule, Move Alpha, by ca1980. In theories of the time, the operation of this single transformational rule is constrained by principles such as subjacency. Conditions on representations further restrict its output, such as the Condition on Extraction Domains and the Empty Category Principle.

This research program and the theories that emerged from it were motivated by two different desiderata that impact on the tasks of linguists. On the one hand there is the task of how best to describe the particular grammars of very syntactically distinct languages across a wide empirical domain; on the other is how to explain the very fact that these grammars are acquired, and the uniformity of their acquisition.

The view in transformational generative grammar in the 1980s was that general mechanisms of learning were not up to the task of explaining how grammatical systems of huge complexity were acquired. The empirical findings of researchers in this tradition were thought to require a fairly rich set of abstract principles that constrained the development of a grammar in an individual. These principles of Universal Grammar had to be restrictive enough to account for acquisition, but loose enough to allow description of the variety of languages in the world. Chomsky (1981) suggested a means of doing this: principles of Universal Grammar allow specific points of variation within them (is Case assigned to the left, or to the right (Koopman 1984)? Is the bounding node for Subjacency S or S' (TP or CP) (Rizzi 1982)? Etc.).

The picture at the end of the 1980s was, then, this: there are a number of Principles of Universal Grammar, deductively locked together, which specify the content and structure of the initial state of language-ready human beings, a state that develops over time and experience to become an I-language. I-languages are not characterised by rules, defining constructions and their interactions, rather they are a specification of which variants of which principles are at play.

The principles of Universal Grammar in this model make reference to language-specific concepts like NP, pronoun, Case, empty category, bounding nodes, theta-roles, etc. These concepts have either a very indirect relationship to concepts relevant to other cognitive modules, or perhaps no relationship at all. The structure of the principles is also highly sui generis: the core ideas of c-command, government, movement, binding, etc. are not obviously derivable from independently known mathematical, physical, biological or psychological relationships. The system, though empirically successful, has the consequence that principles of Universal Grammar, and hence the object that they are a theory of, is complex and highly domain-specific.

The advent of the Minimalist Program in the late 1980s and early 1990s (Chomsky 1991 Chomsky 1993) was an attempt to subject this picture to further questioning. The impetus for this questioning, however, had early roots.

In *Aspects of the Theory of Syntax* (Chomsky 1965), Chomsky, discussing the idea that knowledge of language in an individual derives from very general cognitive skills, such as association and inductive generalization, applied to linguistic experience, remarks:

"However, there is surely no reason today for taking seriously a position that attributes a complex human achievement entirely to months (or at most years) of experience, rather than to millions of years of evolution or to principles of neural organization that may be even more deeply grounded in physical law -- a position that would, furthermore, yield the conclusion that man is, apparently, unique among animals in the way in which he acquires knowledge. Such a position is particularly implausible with regard to language, an aspect of the child's world that is a human creation and would naturally be expected to reflect intrinsic human capacity in its internal organization." (Chomsky 1965: 59)

This quote effectively lays out two of the research programs associated with generative grammar. One, which led to the Principles and Parameters models is: what is the biological content of Universal Grammar, given by human evolution? The other, which revolves around the questions that drive the Minimalist research program, is: how "perfect" is human language? How much of human language can be attributed to principles operative outside language, and possibly outside the individual. As pointed out by Chomsky (2007: 1), the first question, which is epistemological, can be recast as the second, which is metaphysical.

A later impetus for the Minimalist program, highlighted in Chomsky 2000, is that human language appears to have been a fairly recent evolutionary development, having emerged in, at most, the past 200,000 years (Tallerman and Gibson 2011). If the content and structure of the principles of Universal Grammar are as complex and sui generis as they appear, how did they come to evolve in such a short space of evolutionary time? If Universal Grammar is less complex and less specific, and if the apparent principles of Universal Grammar can be derived from more general principles, then this question becomes easier to answer. Universal Grammar could be the result of a small evolutionary change in the genome, a change whose phenotypical effects would be considerable.

Simplifying Universal Grammar in the way demanded by the Minimalist program may lead to (at least temporary) loss of empirical coverage. Here we may be faced with an example of what is known in philosophy of science as "Kuhn-loss", which can be defined as follows: "a later period of science may find itself without an explanation for a phenomenon that in an earlier period was held to be successfully explained" (Bird 2018; see Kuhn (1962/1970a, 99–100)). We will consider a possible case of this, in relation to the Empty Category Principle of GB theory, in Section 2.2.

The Minimalist program, as developed over the last few decades, interrogates the principles of Universal Grammar by asking two questions, both of which attempt to reduce the specificity of Universal Grammar without losing the descriptive and explanatory properties of its principles. The first is how much of the *content* of the principles of Universal Grammar can be derived from the interaction of the human language faculty with the rest of human cognition? The second is to what extent Universal Grammar is optimally *structured* for the role that it plays in its interaction with those other cognitive systems?

## 2. How does a Minimalist Architecture impact on this?

The minimalist analysis of the principles of Universal Grammar is, to a great extent, the whole program of Minimalist Syntax, which we cannot do justice to in this short piece. Instead, we take two case studies, one focussing on the nature of structure building and the other on constraints on that process, and show how one might proceed to rethink principles of UG in these terms. We draw on particular suggestions in the literature, but the intention is not so much to argue for these as to show how this kind of thinking looks. Such work attempts to solve problems with multiple unknowns. We have only vague ideas about what count as the right notions of minimality, simplicity, and optimization. The correct principles of Universal Grammar are, without doubt, only understood to some approximation. The structure of the interfaces is largely guesswork. Because of this, minimalist analysis is bound to be tendentious, and dynamic, but it is, we think, capable of much insight and of opening up new questions.

2.1 Case Study: phrase structure, movement and locality.

Generative grammar has always taken there to be an irreducibly computational core to syntax, seeing syntax as a device for pairing form and meaning over an unbounded range using finite resources, and drawing from ideas in recursive function (computability) theory to do so (Chomsky 1967, p405-408). The minimalist program seeks to understand whether this computational core is in some sense minimal, and whether it interacts with the systems that put it to use in an optimal way. More concretely, can the principles of Universal Grammar be reduced, without loss of descriptive or explanatory power, to these putative properties of the system?

Focussing first on the nature of the computational system itself, we begin by asking what counts as minimal for a computational system.

Classical recursive function theory (like the equivalent theory of Turing Machines) is a way to define what a computational system is in general. Computation is seen as the composition of elementary functions, so simple that they cannot be decomposed further (e.g. Kleene 1952). The three elementary functions of the theory (i) map any input to zero, (ii) map any input to that input, and (iii) map an input to a minimal change from the input, e.g., its successor. In addition to these elementary functions, there are three modes of composing functions recursively: (i) substitution (which allows you to substitute a function for the output of that function); (ii) primitive recursion (which defines a new function in terms of itself); and (iii) minimization (which calculates the composition of functions up to the minimal point of success and returns a result). These ways of combining the elementary functions have certain properties which one can think of as computationally economical: they reuse the results of previous computations, they minimize the amount of information that is stored within the computation, they return the result that is available after the shortest computation, etc.

If syntax is a computational system then the first question to be answered is: what is the elementary function in syntax? The second is: is the way it works also characterized by computationally economical properties? The hunch in Minimalist approaches to syntax is that a number of the properties of Universal Grammar can be rethought in these terms.

For example, within GB theory, X-bar theory (which is a collection of principles that regulate the organization of phrase structure) is responsible for building syntactic structure as a projection of lexical properties. The principles of X-bar Theory require that when a lexical item X combines with a complex phrase Y to create a larger structure, [X Y], the categorial properties of the lexical item X are projected to that larger structure, with a specification that the projection admits further combinations (the notion of a bar level projection). Further, the sister of the lexical item X is marked with the information that it is a maximal projection (YP), so we have [$_{X'}$ X YP]. Other principles require that when two complex phrases combine, one of these (the phrase marked as a bar-level phrase) will be stipulated as projecting its categorial properties to the larger item, giving [$_{XP}$ ZP X'].

The Bare Phrase Structure theory of Chomsky 1995a proposes reducing the various stipulations that comprise these principles of X-bar Theory to a single operation, Merge, plus some principles of computational economy. Merge is a function that takes two arguments and creates a larger unit from them. Chomsky proposes that that larger unit should be thought of as a set, since it specifies no ordering information, and sets are simple mathematical structures that add no further information when they are formed, beyond the very fact of the grouping of their members. Merge, then, takes X and Y as arguments, to output a set: Merge(X,Y) = {X, Y}.

The role of Merge in syntax is similar to the role of an elementary function (like the zero function, or the identity function) in recursive function theory, in that it is intended to be so elementary that it cannot be broken down further. Since syntax clearly requires grouping, Merge groups two distinct units into a larger unit, but adds no further information than that. We can compose the output of Merge with Merge itself (cf. the substitution pattern in recursive function theory), allowing the output of Merge to act as its input, and this will then give Merge an unbounded range. Merge is intended to be as minimal a computational function as is possible that will generate unbounded complex syntactic structure.

Since Merge adds no information beyond grouping, the question is how the projection properties of the principles of X-bar Theory are dealt with. In Bare Phrase Structure theory, the claim is that these are given by the lexical item, plus a principle of how to project the information in a lexical item through the structure. Later approaches (Chomsky 2013) supersede this, deriving the fact that it is the lexical item X in an [X YP] structure that projects in terms of the minimization of computation. It is instructive to look in some depth at how this works.

Assume that X Merges with YP, forming {X, YP}, where YP is a complex phrase (a result of a previous Merge), while X is simplex, a lexical item. The label (that is category and other relevant information) of the simplex object is immediately determined by the fact that it is an argument of Merge, whereas the label of the complex object requires further computation (essentially looking inside YP to determine a simplex object within it). The reason that the label of the simplex element projects is, then, not a stipulated principle of X-bar theory, but is rather derived by appeal to a non-language-specific property of computational systems: minimize computation. This approach assumes that each output of Merge has to be labelled, and that the label is determined by minimal computation. The former assumption might be motivated by how syntax is optimized for semantic interpretation in that there must be some mapping from syntactic categories to their meanings, while the latter is motivated by non-language specific considerations.

Turning to cases where two phrases are combined, X-bar Theory stipulates the labelling as [$_{XP}$ YP X'] (that is, the bar-level projects to the phrasal level). Within Bare Phrase Structure, we have instead {{Y, ZP}, {X, WP}}, where each substructure is equal in complexity. Following Moro (1997, 2000), Chomsky 2013 takes such structures to be unlabelled, which means, assuming that semantic interpretation requires a label, that these structures would be ruled out. This is a case where the new theory has a different outcome to the old one. Of course, the question then is how to understand the phenomena that this aspect of X-bar Theory had been used to analyse (Specifiers). We return to this below, after discussing a further innovation connected to Merge. The final logical possibility, Merge of two lexical items, we leave aside here (see Chomsky 2013 for discussion).

The classical Principles and Parameters model treats movement and the building of phrase structure as two distinct parts of Universal Grammar, governed by distinct principles (X-bar Theory and Move Alpha). This is a residue of the earlier architecture where these were two rule systems: a phrase structure rule system followed by a separate transformational rule system. Chomsky (2004) suggests that this was an error, and argues that structure building and structure transformation can be elegantly collapsed into the single computational operation Merge.

Assume a structure {X, {Y, Z}}, then

(1) Merge(Y, {X, {Y, Z}}) = {Y, {X, {Y, Z}}}

There is no extension in how Merge is defined here. The innovation is simply the recognition that the domain of Merge includes sub-parts of material already constructed by Merge (similar to primitive recursion in recursive function theory).

If Merge applies to two units that are distinct from each other, it is called External Merge, and if one element is part of the other, it is called Internal Merge, but crucially this is the same operation. In this case we have a unification of distinct aspects of the theory: it had been assumed that Universal Grammar distinguishes structure building from structure changing, but the new approach claims that these are the same thing.

Structures derived using Internal Merge involve two instances of the same element in the structure (e.g. the two instances of Y in {Y, {X, {Y, Z}}}). Chomsky (1993, 1995b) calls these "copies," but there is no special process that makes copies, there is simply Merge. The two copies are similar in one fashion (they are identical in constitution) and distinct in another (they appear in different places in structure). Empirical phenomena suggest that, if the Ys are related by Internal Merge, the interface systems treat them as a single element (they are pronounced only once, and they are interpreted only once), though numerous questions arise in both cases.

Moro and Chomsky appeal to the properties of copies in addressing the question of what label ensues when two complex structures are Merged (as in the case of {{Y, ZP}, {X, WP}}, discussed above). The idea is that such structures cannot be labelled by simply inspecting the constituents in the set, but that in cases where one or the other of these undergoes further Internal Merge, the label of its copy is not seen by the computational mechanisms that calculate labels: these see only the highest copy.

However, this is insufficient to solve the problem. Obviously, if {X, WP} in {{Y, ZP}, {X, WP}} Internally Merges, it will Merge with a complex structure, recreating the problem.

The solution Chomsky proposes to this is that labelling will succeed if the heads of both complex constituents have the same label.

To see how this works, assume that {X, WP} has undergone Internal Merge to some larger structure headed by the lexical item U, so we have:

(2) Merge({X, WP}, {U, {{Y, ZP}, {X, WP}}}) = {{X, WP}, {U, {{Y, ZP}, {X, WP}}}}

The two instances of {X, WP} are identical and the not unreasonable assumption is that they therefore are interpreted and labelled only once. To determine that there are indeed two copies, the computation that labels structures needs to apply to the structure containing both.

There are two obvious possibilities for how the labelling computation operates, assuming it operates in serial mode. If it applies bottom up, starting at {{Y, ZP}, {X, WP}}, it will be unable to label this structure. It then applies to the structure headed by U, labelling it with whatever the label of U is (call this Label(U)). But now we have again two complex structures. The idea is that the computation looks at X, and if Label(X)=Label(U), it succeeds. However, on a bottom up approach, the computation now needs to backtrack to {{Y, ZP}, {X, WP}}, discount {X, WP}, as it is a copy, and then label the structure with Label(Y).

The alternative is that the labelling computation starts at the top of the structure, in which case it cannot compute the whole structure by looking at its head (as there is none), so it looks at both Label(X) and Label (U) and, if they are the same, labels the whole structure with that label. It then continues down the structure, labelling {U, {{Y, ZP}, {X, WP}}} with Label(U), and on encountering {{Y, ZP}, {X, WP}}, it recognizes that it has encountered the higher copy of {X, WP}, and so discounts the lower copy, resulting in labelling that constituent via Label(Y). The top down approach doesn't require the same backtracking as the bottom up approach, and from that perspective might be considered more minimal.

For the top-down approach to work, however, something must provide a 'start symbol' for the labelling computation. This start symbol defines what is called a Phase for the computation. The computation internal to the Phase must have a memory of some sort, if it is to determine which elements are copies (Chomsky 2008).

If the computation responsible for labelling is bounded by Phases in this way, then some sort of signal to the computational operations is necessary. To provide a 'start symbol' for the Phase, a lexical item (head) is needed, since heads require no further computation to determine relevant properties. This suggests that the computational system of human language has at least one (ideally one) phase head, easily recognizable to computational operations. Once a phase head is Merged, the labelling computation is triggered (and plausibly other computations necessary for other cognitive systems to use the results of the syntax).

It is possible that the Phase for the labelling computation is irrelevant for other computations, such as Merge. The minimal hypothesis, however, is that all computational operations are bounded by Phases, and moreover, that the Phases for all operations are the same. A further question is what the domain of the computation is when the phase head is Merged: is it the constituent containing the phase head, or its complement? Again, minimal computation would suggest the former, since the latter will require the operations to ignore part of what is in their domain.

Adopting this, we can reduce some aspects of the principles of Universal Grammar that had been appealed to ensure the locality of Move Alpha to phases.

One aspect of the Universal Grammar principle of subjacency is the idea that movement cannot cross finite clause boundaries which are not furnished with an 'escape-hatch.' Together with other stipulations, this provides one part of a theory of the locality of movement. Although a consensus analysis of all aspects of the locality of movement is still not available within minimalist syntax, the idea that Internal Merge cannot cross finite CPs is firmly established (given empirical evidence like that of McCloskey 2002): apparent long distance movement is successive cyclic.

Identifying finite C as a phase head is one part of capturing this aspect of Subjacency. The idea is that once all the necessary computations have been carried out on a phase, the information in that phase is then no longer available to further computation (this is the Phase Impenetrability Condition (PIC) of Chomsky 2008). The PIC remains a stipulation, though it does function to minimize computation: it delineates a window within which all the relevant computational operations should terminate.

As it stands, however, this set of ideas is insufficient for capturing the cyclicity of long distance movement: The PIC as stated entails that when a finite CP is Merged to another element (such as a V or N), nothing from that CP is available to the higher Phase. This will of course rule out successive cyclic movement. The solution that has been proposed (which is a version of the older 'escape hatch' idea) is that the 'edge' of the Phase is, in fact, accessible to higher computation. One way of thinking about this is that the label of the lower Phase must be available at the higher Phase, or the selectional relationship between, say, V and CP cannot be ensured (by whatever mechanisms ensure it). Since the label is determined by a computation that looks at both the head of CP and the head of its specifier, both the specifier of CP and C must be accessible to the higher phase. This means that CP will be able to Externally Merge with V and that both CP and its specifier will be able to Internally Merge with higher structure, allowing successive cyclic derivations.

Of course, one might ask why language is designed in that way. Why can CPs Merge with Vs at all in human languages, given that the narrowing of the PIC to something smaller than CP seems to be an imperfection from the point of view of minimizing computational load? Presumably the solution to this comes down to something imposed by the interface with the conceptual intentional system, though it remains mysterious.

This analysis does not fully derive successive cyclic movement from language-external concepts. There is a residue of a principle of UG (a language specific principle identifying lexical items bearing the syntactic category C as phase heads, as part of the general theory of syntactic categories, which itself appears to be language-specific), and the PIC still requires that the 'edge' of CP remain available to computational operations in the higher phase. However it does go some way to providing a rationale, in terms of language external computational factors, for what was earlier stipulated as part of a principle of UG. There are of course further stipulations to be made. It is often argued that there is also a phase head at the verb phrase level, so that finite clauses are biphasal. There is empirical evidence for this in some languages (e.g. van Urk 2018), but it does in principle double the stipulations that need to be made about phase heads, unless the existence of the second phase head can be shown to derive from a deeper principle and/or the behaviour of the two phase heads is identical.

Recapping, the effects of the principles of Universal Grammar that constitute X-Bar Theory, and those that license and to some extent constrain movement, are reduced to a more minimal set of principles whose operation is constrained by considerations that are strictly outside of the human language faculty, and plausibly external to the individual: minimization of computation. The new system looks as follows:

(3) a. Merge (encompassing X-bar principles of combination and Move Alpha), defined as an elementary function whose unbounded range is given by its recursive combination.
b. Label (encompassing X-bar principles of projection), defined as an elementary function, bounded by Phases.
c. Phases (encompassing the successive cyclic nature of Move Alpha), defined as limited memory spaces for computation.

2.2 Case Study: the Empty Category Principle, subject-object asymmetries and labelling

In this section we illustrate the move from GB-style UG principles to the sparser but conceptually more elegant and satisfactory minimalist approach by looking at a subset of the phenomena accounted for by the Empty Category Principle (ECP). The ECP was a central principle of UG that was the object of a great deal of research in the 1980s. Originally proposed in Chomsky (1981: 250), we can formulate it as follows:

(4) Traces must be properly governed.

Clearly, (4) relies on the notion of government. This notion, imported from traditional grammar in seminal work by Jean-Roger Vergnaud (see Vergnaud 1977/2008) as part of the theory of abstract Case (Rouveret & Vergnaud 1980, Chomsky 1981: 162f), was formalised in a variety of ways and successively refined, particularly in relation to the ECP. The core notion of government can be formulated as minimal c-command by a head, as in (5):

(5) $\alpha$ governs $\beta$ iff
(i) $\alpha$ is a head,
(ii) $\alpha$ c-commands $\beta$ and
(iii) there is no $\gamma$, $\gamma$ a head, such that $\gamma$ c-commands $\beta$ and $\alpha$ c-commands $\gamma$.

(Beginning with Aoun & Sportiche 1983, some versions of government relied on m-command, command by a head of all elements dominated by exactly the same set of XPs, in order to allow a head to govern its specifier, e.g. for the assignment of Nominative Case by T/Infl).

It is clear from (5) that the core case of government is the head-complement relation (and this is where the connection with traditional grammar is clearest, since traditional grammars would speak of a given Latin Preposition, for example, as governing a particular case on its complement). The relevant configuration is (6), where $\alpha$ is a head, and $\beta$ can be anything as long as it is in the local relation to $\alpha$ specified by (5):

(6) $[_{\alpha P} \ldots \alpha \ldots [\ \ldots \beta \ldots [_{\gamma P} \ldots \gamma \ldots ]]]$

But $\alpha$ can also govern into a lower Specifier, as may be required in Exceptional Case-marking configurations, where $\alpha$ is a verb of the relevant class and $\gamma$ is non-finite T/Infl (minimal m-command would not permit this, as $\gamma$ m-commands $\beta$ here):

(7) $[_{\alpha P} \ldots \alpha\ [_{\gamma P}\ \beta\ [_{\gamma'} \ldots \gamma \ldots ]]$

Furthermore, if c-command is irreflexive (which it is almost always assumed to be), $\alpha$ governs $\gamma$ in (7).

The proper-government relation was intended to be a proper subset of the government relation (hence the term). One definition was (8) (this definition is close, but not identical, to that of Chomsky 1981: 250; we have modified Chomsky's definition for expository purposes):

(8) α properly governs β iff
(i) α is a **lexical** head,
(ii) α c-commands β and
(iii) there is no γ, γ a head, such that γ c-commands β and α c-commands γ.

The restriction in (8), as compared to (5), is that (8) refers to a proper subset of the set of heads, namely lexical heads.

The definition in (8) accounted for subject-object asymmetries, situations where objects seemed to have syntactic privileges not accorded to subjects. The asymmetry could be derived from the fact that subjects are typically in the domain of functional heads, being in SpecTP, therefore governed by C (or, on the m-command approach, by T), while objects are in VP and hence governed by the lexical category V. Objects are therefore properly governed, while subjects generally are not.

A range of subject-object asymmetries in various languages was elegantly accounted for by the ECP, with proper government defined as along the lines of (8). Foremost among these, arguably, were English complementiser-trace effects (first observed by Perlmutter 1971, and playing an important role in theory of filters of Chomsky & Lasnik 1977). The core phenomenon is illustrated in (9):

(9) a. Who did you say [$_{CP}$ (that) John saw t ] ?
b. *Who did you say [$_{CP}$ that t saw John ] ?
c. Who did you say [ t saw John ] ?

In (9a), the trace of wh-movement in direct-object position is properly governed by the lexical verb. In (9b) on the other hand, the trace in subject position is not properly governed: neither C nor T can properly govern it since they are functional heads, and *say* in the main clause is prevented from properly governing the lower SpecTP because the complementiser *that* in C is a closer c-commander (it would be γ by the definition in (8)). Since the trace is not properly governed, the sentence is ungrammatical. Here we can observe that the ECP is a condition on representations: traces have to appear in a particular configuration in order to be well-formed, their derivational history is irrelevant.

Various proposals were made in order to account for the well-formedness of (9c). By the logic of the ECP, the trace in the subject position here must be governed by a lexical head. A natural proposal is that C (and so CP) is not present, allowing *say* to properly govern the trace, there being no closer c-commanding head in that case. A further idea is that the null C, as opposed to *that*, has some special property making it able to properly govern; this idea is developed in particular in Rizzi (1990: 51f). This idea was motivated by the *que/qui* alternation in French. The analogous paradigm to (9) in French is (10):

(10) a. Qui as-tu dit [$_{CP}$ que Jean a vu t ] ?
b. *Qui as-tu dit [$_{CP}$ qu' t vu Jean ] ?
c. Qui as-tu dit [$_{CP}$ qui t vu Jean ] ?

(10a) and (10b) are exactly parallel to their English counterparts in (9a,b) and can clearly be accounted for in exactly the same way. In (10c), however, the trace in subject position must be properly governed by the special complementiser *qui*. It appears, then, that certain complementisers have special proper-government privileges. Rizzi (1990: 51f) extended this idea to the null complementiser of English, accounting for (9c) in a way parallel to (10c).

Still a further idea is that the trace in SpecCP, required for successive-cyclic movement (i.e. ultimately by subjacency; see the previous section), could properly govern in virtue of being the antecedent of the trace in subject position in a configuration like (11) (Chomsky 1981: 250-1):

(11) Who did [$_{TP}$ you say [$_{CP}$ t [$_{C}$ ø ] [$_{TP}$ t saw John ]]] ?

Antecedent government is thus a further case of proper government. The ECP requires both lexical-head government and antecedent government (although later versions of the ECP were disjunctive, requiring either lexical-head government or antecedent government; see Chomsky 1986: 17, 22; Rizzi 1990: 32 for a conjunctive definition of the ECP, and the discussion in Lasnik & Saito 1992: 156-8). We can note in passing that antecedent government departs significantly from the intuitive idea of government with its roots in traditional grammar. We will not linger over these variants of the definition of proper government here.

Starting from Kayne (1981), the empirical coverage of the ECP was extended to further subject-object asymmetries, for example those seen with the French empty negative quantifier in (12) (Kayne 1981: 95):

(12) a. Jean n'a pas trouvé [ *e* de livres ].
John Neg=has Neg found of books
"John hasn't found (any) books."

b. *[ e De livres ] n' ont pas été trouvés (par Jean).
of books Neg=have Neg been found by John

Here the nominal containing the empty quantifier *e* (occupying a position in the nominal in complementary distribution with other quantifiers such as *beaucoup* 'a lot', *peu* 'little/few', etc.) shows a subject-object asymmetry which is accounted for by the ECP, in that, by the definition in (8), *e* is properly governed in (12a) but not in (12b).

A further case was pointed out by Longobardi (1994). Bare NPs in Italian (and elsewhere in Romance, although not in French) show a subject-object asymmetry in their distribution:

(13) a. *Latte è buono.
Milk is good.

b. Il latte è buono.
The milk is good (generic/existential).

c. Bevo latte.
I.drink milk.

If the bare nominal *latte* has the structure *[$_{DP}$ [$_{D}$ e ] [$_{NP}$ latte ]]*, then *e* is properly governed in (13c) but not in (13a). In (13b) an "expletive determiner" *il* appears, and so there is no occurrence of *e*; the ECP forces the appearance of this determiner here.

Belletti & Rizzi (1981) observed similar subject-object asymmetries in the cliticisation of the Italian partitive-genitive clitic *ne*:

(14) a. Tre settimane passano rapidamente.
Three weeks pass.3pl quickly
"Three weeks pass quickly."

b. Gianni ne passerà [ tre *e* ] a Milano.
John of-them spend.Fut.3sg three in Milan
"John will spend three in Milan."

c. *[ Tre *e* ] ne passano rapidamente.
Three of-them pass.3pl quickly

In (14b), the trace of *ne* is properly governed by V; in (14c) it isn't properly governed. Similarly, *ne*-cliticisation from the single argument of the unaccusative V in (15a) is grammatical since the trace is properly governed by V, but not from the single argument of the unergative in (15b):

(15) a. Ne sono cadute molte.
Of-them are fallen.f.pl many.f.pl
"Many of them have fallen."

b. *Ne hanno telefonato molte.
Of-them have phoned many.f.pl.

Baker (1985) observed that Noun-Incorporation in Mohawk and elsewhere shows a similar distribution to *ne*-cliticisation, as the unaccusative-unergative contrast in (16) shows:

(16) a. Wa'- ka- wir- ʌ'-ne'. [Mohawk]
FACT-NsS-baby-fall-PUNC
"The baby fell."

b. *Wa'- t- ka- wir- ahsʌ'tho-'.
FACT-DUP-NsS-baby-cry- PUNC
"The baby cried." (Baker 1996:293-4)

So we see that the ECP, with proper government defined as in (8), can elegantly account for a range of phenomena, notably subject-object asymmetries of the kind illustrated in (9), (10) and (12-16). (A much wider range of phenomena is accounted for by antecedent government; see Chomsky 1981: 250f, 300f, Chomsky 1986: 16-28, Kayne 1981, Huang 1982, Jaeggli 1980: 252f, Lasnik & Saito 1984, 1992, Rizzi 1990, but these range well beyond the specific cases of subject-object asymmetries we are concerned with here; it is worth mentioning, though, that Rizzi 1990 proposed Relativised Minimality as a general theory of antecedent government, and that Relativised Minimality has outlived the ECP as an independent explanatory locality principle). The abandonment of the notions of government and proper government, along with the ECP, in minimalism appears to be a case of Kuhn loss.

In this connection, two observations can be made. First, in (12), (13) and (16) at least, we can see that the object is able to lack something that the subject must have, essentially a determiner, quantifier or agreement of some kind. Second, the definition of proper head government (as opposed to antecedent government, see above) relies on the core insight that objects are complements, while subjects are not. We now tentatively suggest that this core insight can be exploited in terms of the labelling theory of Chomsky (2013, 2015) in such a way as to allow us to retain an account of the asymmetries in (12-16) (but not the Comp-trace effects in (9) and (10), as we shall see).

As we saw in the previous section, the Labelling Algorithm (LA) proposed by Chomsky (2013, 2015) has three subcases (where α arises from Merge(X, Y), and so has no label in virtue of Merge alone):

(17) a. [$_{\alpha}$ X YP ] -- X is a head and YP is not (i.e. X is minimal and Y is not);
b. [$_{\alpha}$ XP YP ] -- neither XP nor YP are heads (i.e. minimal);
c. [$_{\alpha}$ X Y ] -- both X and Y are heads (minimal).

In (17a), X is the head and provides the label for α. Here we retain the standard head-complement asymmetry, based on (non-)minimality. In (17b), X and Y must share a feature, or one of them must move leaving the other to label α. For present purposes, we leave (17c) aside.

Subject-object asymmetries pattern with the difference between (17a) and (17b): objects correspond to YP in (17a); subjects to XP in (17b). Therefore subjects contribute to labelling in the configuration in (17b), while objects, in the configuration in (17a), do not. We propose that it is for this reason that subjects are required to be featurally, and therefore in many cases, structurally, richer than objects; the relevant feature may well be the Person feature, which, following Longobardi (2008), is associated with D (although other features may be relevant in other languages; see Richards 2014). The ungrammatical examples in (12b), (13a) and (16b) then result from the fact that the elements in subject position lack the features required for them to participate in the labelling operation. The DP in (12b) lacks a lexical head, the head of the DP in (13a) is empty, and the trace of the subject in (16b), as a copy, lacks features entirely, whether it is VP internal or in the specifier of TP.

Concerning *ne*-cliticisation in (15) and (16), something more must be said. First, we should observe that a *ne*-less subject containing a numeral or quantifier can appear in subject position:

(18) Tre passano rapidamente.
Three pass.3pl quickly
"Three pass quickly".

Chomsky (1981: 301) suggests that the subject here is *[$_{NP}$ tre PRO ]*. We could try to update that idea by saying that the subject here has "pronominal content" in having a Person feature able to participate in labelling. However, this proposal is inconsistent with the evidence that the finite T alone is able to label TP in examples like (15a) (assuming there is no "expletive *pro*"); in fact, Chomsky (2015: 9) suggests that such a "strong" T, rich in φ-features, characterises null-subject languages like Italian and hence subjects are not required to move to SpecTP in these languages. Nonetheless, we can postulate that a nominal permitting *ne*-extraction lacks the crucial feature which blocks subextraction (perhaps the high A-position Specifier proposed by Cinque 2013: 86) and that this position, or the feature creating it, must be present in a preverbal subject. This implies that, although T is strong enough to label TP on its own, if a DP occupies its Spec position, that DP cannot be inert for labelling. In this way, we guarantee that *ne*-cliticisation is impossible from preverbal subjects. Finally, we account for the ungrammaticality of *ne*-cliticisation from "freely-inverted" postverbal subject as in (15b) following Belletti (2004: 21): the subject raises to a Focus position in the low left periphery and the remnant IP, containing the trace of the subject, raises past it, hence *ne* fails to c-command its trace. In this way, the subject-object asymmetries in *ne*-cliticisation originally observed by Belletti & Rizzi (1981) and discussed and analysed in terms of the ECP in Chomsky (1981: 300f) can be fully accounted for.

On the other hand, the original Comp-trace effects seen in (9) and (10) are difficult to account for in terms of the difference between (17a) and (17b). Here an account along the lines of those put forward by Bošković (2016) and Douglas (2017) seems to be preferable. Labelling may play a role, though: as Bošković (2015) observes, where wh-movement is triggered by the inability of the wh-phrase and the head it is the specifier of to label α in (17b), wh-movement must take place before α is labelled. In that case, α cannot count for the computation of the Anti-locality Condition in (19):

(19) Anti-locality condition:
Movement of A targeting B must cross a projection distinct from B ***(where unlabelled projections are not distinct from labelled projections***).
(Bošković 2016:3, emphasis added)

The Comp-trace configuration is given, at the relevant stage of the derivation, in (20):

(20) [$_{CP}$ that [$_{\alpha}$ who left ]]

Here, *who* must move before α is labelled. Hence, movement to SpecCP fails to cross a projection distinct from C, in violation of (19), since unlabelled projections are not distinct from labelled ones. On the other hand, one-step movement beyond CP violates the Phase Impenetrability Condition, since it crosses the phase-boundary CP from a non-edge position (see previous section). The window for well-formed movement defined by the interaction of locality and anti-locality constraints is closed. The contrast between (20) and cases where there is no overt complementiser may involve a smaller, non-phasal domain, as both Bošković and Douglas propose in different ways.

We could perhaps extend this approach, and postulate that vP is the head-labelling domain; here labelling works as in (17a). TP, on the other hand, is the Spec-head labelling domain; labelling works as in (17b) here. Therefore, DPs occurring in SpecTP need more features to participate in labelling. This derives many of the subject-object asymmetries formerly handled by the ECP, as we have seen. Furthermore, it correlates with the idea that there is frequently more inflection in TP than in vP: subject agreement is cross-linguistically more frequent than object agreement, null subjects are cross-linguistically more frequent than null objects, etc. Intriguingly, the phase boundary in the highest vP (or VoiceP) projection would then mark the transition point between the two kinds of labelling; this may vary parametrically, a point whose exploration would take us too far afield here.

Why should verbal categories be distinct from T (and other possible "inflectional" categories such as Mood, Aspect, etc.)? Perhaps these categories are "strong" enough to provide their own label because they are "more lexical" than T-categories in that they contribute θ-roles/events. Because of this, they have fewer purely formal features than T-categories.

Leaving these final more speculative points aside, we have seen here how many of the subject-object asymmetries formerly accounted for by the ECP can receive a novel and interesting treatment in terms of labelling, and this treatment has some intriguing further theoretical implications. As far as these asymmetries are concerned then, if the proposals here are correct, there has been no Kuhn-loss in this domain; in fact we see that the new scientific framework, the Minimalist program, opens up new questions.

## 4. Conclusion

In the foregoing we have tried to outline the main respects in which the Minimalist program has innovated, while still retaining the principal goals of generative theory. UG has the same general function of defining what a humanly attainable I-language can be, although it is now considerably less rich compared to earlier theories, notably GB theory. We have shown that the principles of X-bar theory and of Movement, as well as at least part of the principles of locality that enforced successive cyclic movement, can be understood not as aspects of human genetic endowment, but rather as reflections of language external computational constraints interacting with a minimal UG. Although there has undoubtedly been some Kuhn-loss in the shift of paradigms from GB to Minimalism, our case studies of the subject-object asymmetries and the ECP have further shown that it is possible to reanalyse some of the phenomena formerly captured by the ECP in a novel way in terms of labelling. Arguably, here we are witnessing theoretical progress.

## References

Aoun, Josef, and Dominique Sportiche. On the formal theory of government. The Linguistic Review, 2. 211-236.

Baker, Mark. 1985. Incorporation: A Theory of Grammatical Function Changing. Ph.D. thesis, Massachusetts Institute of Technology.

Baker, Mark. 1996 The polysynthesis parameter. Oxford: Oxford University Press.

Belletti, Adriana, and Luigi Rizzi. The syntax of "ne": some theoretical implications. The Linguistic Review 1.2 (1981): 117-154.

Belletti, Adriana. 2004. Aspects of the low IP area. Luigi Rizzi, ed. *The structure of CP and IP. The cartography of syntactic structures*. 16-51. Oxford: Oxford University Press.

Bird, Alexander. 2018. "Thomas Kuhn", The Stanford Encyclopedia of Philosophy (Winter 2018 Edition), Edward N. Zalta (ed.), URL <https://plato.stanford.edu/archives/win2018/entries/thomas-kuhn/>. Accessed 1/3/21.

Bošković, Željko. 2016. On the timing of labeling: Deducing Comp-trace effects, the Subject Condition, the Adjunct Condition, and tucking in from labelling. The Linguistic Review. 33: 17-66.

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

Chomsky, Noam. 1965. Aspects of the theory of syntax. Cambridge, Massachusetts: M.I.T.

Chomsky, Noam. 1967. The formal nature of language. In Eric Lenneberg, Biological Foundations of Language, 397–442, New York, N.Y.: Wiley and Sons.

Chomsky, Noam. 1973. Conditions on transformations. In Stephen Anderson and Paul Kiparsky, eds., A Festschrift for Morris Halle, 232–286, New York: Holt, Rinehart, and Winston.

Chomsky, Noam. 1981. Lectures on Government and Binding. Dordrecht: Foris.

Chomsky, Noam. 1986. Barriers. Cambridge, MA: MIT Press.

Chomsky, Noam. 1991. Some notes on the economy of derivation and representation. In R. Freidin, ed., Principles and Parameters in Comparative Grammar, Cambridge, MA: MIT Press.

Chomsky, Noam. 1993. A minimalist program for linguistic theory. In Kenneth Hale and Samuel Keyser, eds., The view from Building 20, 1–52, Cambridge, MA: MIT Press.

Chomsky, Noam. 1995a. Bare phrase structure. In Gert Webelhuth, ed., Government and Binding theory and the Minimalist Program, 383–439, Cambridge, MA: Basil Blackwell.

Chomsky, Noam. 1995b. The minimalist program. Cambridge, Massachusetts: MIT Press

Chomsky, Noam. 2000. Minimalist inquiries: the framework. In Roger Martin, David Michaels, and Juan Uriagereka (eds), *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, 89–155. Cambridge, Mass.: MIT Press.

Chomsky, Noam. 2004. Beyond explanatory adequacy. In Adriana Belletti, ed., Structures and Beyond, 104–131, New York: Oxford University Press.

Chomsky, Noam. 2007. Approaching UG from Below. In Uli Sauerland and Hans-Martin Gärtner, eds., Interfaces + Recursion = Language? Berlin: Mouton de Gruyter.

Chomsky, Noam. 2008. On phases. In Robert Freidin, Carlos P. Otero, and Maria Luisa Zubizarreta, eds., Foundational Issues in Linguistic Theory, 133–166, Cambridge, MA: MIT Press.

Chomsky, Noam. 2013. Problems of projection. Lingua 130:33 – 49.

Chomsky, Noam. 2015. Problems of projection: Extensions. In E. Di Domenico, C. Hamann & S. Matteini *Structures, strategies and beyond: Studies in honour of Adriana Belletti* 1-16. Amsterdam: John Benjamins

Chomsky, Noam & Howard Lasnik. 1977. Filters and Control. Linguistic Inquiry 8. 425-504

Douglas, Jamie. 2017. Unifying the that-trace and anti-that-trace effects. Glossa 2:60.

Huang, C.T. James. 1982. Logical relations in Chinese and the theory of grammar. Ph.D. thesis, Massachusetts Institute of Technology.

Jaeggli, Osvaldo. 1980. On some phonologically-null elements in syntax. Ph.D. thesis, Massachusetts Institute of Technology.

Kayne, Richard S. 1981. ECP Extensions. Linguistic Inquiry 12. 93-133.

Kleene, Stephen. 1952. Introduction to Metamathematics. Amsterdam: North-Holland Publishing Company.

Koopman, Hilda. 1984. The Syntax of Verbs. Dordrecht, The Netherlands: Foris Publications.

Kuhn, Thomas. 1962/1970a, *The Structure of Scientific Revolutions*, Chicago: University of Chicago Press (1970, 2nd edition, with postscript)

Lasnik, Howard and Saito, Mamoru. 1984. On the Nature of Proper Government. Linguistic Inquiry. 235-289.

Lasnik, Howard & Mamoru Saito. 1992. Move α. Conditions on its Application and Output. Cambridge MA, MIT Press.

Longobardi, Giuseppe. 1994. Reference and proper names. *Linguistic Inquiry* 25: 609–65.

Longobardi, Giuseppe. 2008. Reference to individuals, person, and the variety of mapping parameters. In A. Klinger and H. Mueller eds. *Essays on nominal determination: From morphology to discourse management*. 189-211. Amsterdam: John Benjamins.

McCloskey, James. 2002. Resumption, successive cyclicity, and the locality of operations. In Samuel David Epstein and T. Daniel Seely, eds., Derivation and Explanation in the Minimalist Program, 184–226, Malden, Massachusetts: Blackwell Publishing.

Moro, Andrea. 1997. The Raising of Predicates. Cambridge: Cambridge University Press.

Moro, Andrea. 2000. Dynamic Antisymmetry. Cambridge, MA.: MIT Press.

Perlmutter, David M. 1971. *Deep and Surface Constraints in Syntax*. New York: Holt, Rinehart & Winston.

Richards, Marc D. 2014. Defective Agree, Case alternations and the prominence of Person. In Ina Bornkessel-Schlesewsky, André L. Malchukov, & Marc D. Richards (eds), Scales and Hierarchies: A Cross-Disciplinary Perspective, 173–96. Berlin: de Gruyter.

Rizzi, Luigi. 1982. Issues in Italian syntax . Dordrecht, Holland: Foris Publications.

Rizzi, Luigi. 1990. Relativized Minimality. Cambridge, MA: MIT Press.

Ross, John. 1967. Constraints on variables in syntax. Ph.D. thesis, Massachusetts Institute of Technology.

Rouveret, Alain & Vergnaud, J.-R. 1980. Specifying Reference to the Subject: French Causatives and Conditions on Representations. *Linguistic Inquiry* 11: 97-202.

Tallerman, Maggie and Gibson, Kathleen R. 2011. Introduction: the evolution of language. In The Oxford handbook of language evolution, 1–33, Oxford: OUP.

Van Urk, Coppe. 2018. Pronoun copying in Dinka Bor and the Copy Theory of Movement. Natural Language and Linguistic Theory 36: 937–990.

Vergnaud, J.-R. 1977/2008. Letter to Noam Chomsky and Howard Lasnik on "Filters and Control", April 17th 1977. In Robert Freidin, Carlos Otero, & Maria-Luisa Zubizarreta (eds), *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*, 3–17. Cambridge, Mass.: MIT Press.