The Minimal Revolution: Chomsky's MPLT

Katarzyna Miechowicz-Mathiasen

Uniwersytet im. Adama Mickiewicza, Poznań

0. Abstract

The earliest work on minimalism presenting its most general ideas and assumptions is Chomsky's (1993) paper entitled "A Minimalist Program for Linguistic Theory" (MPLT). As Chomsky puts it, the paper is just a sketch of the program. Careful analyses of certain problematic areas faced by the previous forms of the generative framework the Government and Binding Theory (in *Lectures on Government and Binding,* Chomsky 1981) and the Principles and Parameters Theory (in *Principles and Parameters in Comparative Grammar* Chomsky 1991) contributed to the formation of the Minimalist Program. Chomsky points out the most important changes introduced to the generative approach via the Minimalist Program as well as the problematic aspects haunting the theory. The goal of this paper is firstly to present the ideas as they were introduced and developed by Chomsky and others, and, secondly, to provide analyses of their application in a description of concrete linguistic phenomena. Chomsky concludes his MPLT paper naming the leading assumptions constituting the Minimalist Program, quoting Chomsky (1993, in Chomsky 1995: 212):

i.   A linguistic expression (SD) is a pair (π, λ) generated by an optimal derivation satisfying interface conditions.

ii. The interface levels are the only levels of linguistic representation.

iii. All conditions express properties of the interface levels, reflecting interpretive requirements.

iv. UG provides a unique computational system, with derivations driven by

   morphological properties to which syntactic variation of languages is restricted.

v. Economy can be given a fairly narrow interpretation in terms of FI, length of

   derivation, length of links, and Greed.

All of the above ideas entail reanalysis of a substantial amount of data as well as reformulation of certain rules and conditions. They entail reduction of the levels of syntactic representation provided by the Extended Standard Theory (D-Structure, S-Structure, Logical Form and Phonetic Form) leaving only two of them, namely, the interface levels of Logical Form (LF) and Phonetic Form (PF). Derivations are supposed to satisfy conditions applying only on the aforementioned interface levels, hence conditions previously holding at D-Structure and S-Structure must be reformulated in terms of minimalist assumptions and proved to apply on the remaining interface levels, at LF to be precise. The conditions are restricted to interpretation, namely, only legitimate objects bearing interpretable features are allowed at each of the interface levels. The convergence principle requiring legitimate objects on the interface levels is referred to as Full Interpretation. Any object entering LF or PF equipped with a feature that is uninterpretable on any of the respective levels will cause a crash of a given derivation. Derivations are said to be driven by the need of satisfaction of morphological features, hence any performed movement must satisfy a morphological feature of the moved element (Greed; Chomsky 1993: 266). Economy principles concern both representations and derivations. With respect to representations we apply the aforementioned principle of Full Interpretation (FI), with respect to derivations we will discuss principles such as Shortest Move, Fewest Steps, Procrastinate and Greed. All of these call for a thorough investigation and research. Chomsky's presentation of the problems along with some of his proposed possible solutions in MPLT paper leave a lot of material for further analyses and justification and it is the goal of this paper to address the problems as fully as possible. The

problems will be presented in the following fashion: firstly, we discuss the minimization of the levels of syntactic representation along with the consequences this step entails; secondly, we are going to present the way in which phrase structure within the minimalist framework is built up and show the workings of the computational system both in the overt and covert component; thirdly, we are going to examine the reasons behind movement in syntax and introduce the concept of feature checking (Checking Theory) in minimalism; lastly, we are going to put forward a detailed discussion on economy principles in the Minimalist Program and provide reanalysis of the well-known problematic structures that can now be accounted for by the economy conditions.

1. Levels of syntactic representation.

Universal Grammar (UG) is to provide specification of the levels of syntactic representation. It is also a task of UG to determine all possible symbolic representations and derivations. Within the previous frameworks the so-called Extended Standard Theory, part of UG, assumed the existence of four levels of representation, namely: D(eep)-Structure, S(urface)-Structure, Logical Form (LF) and Phonetic Form (PF). The first three levels constitute the syntactic component and the level of PF constitutes the phonological component, the representation of sound. UG provides language with the computational system C_{HL} whose task is to take items from the lexicon and form derivations. Derivations result in formation of linguistic expressions, the so-called structural descriptions (SD), which are sequences of representations: one at each linguistic level; hence, every linguistic expression within the EST would be a sequence of four representations, one at each of the following: D-Structure, S-Structure, LF and PF. The model of the EST linguistic levels has the following form:
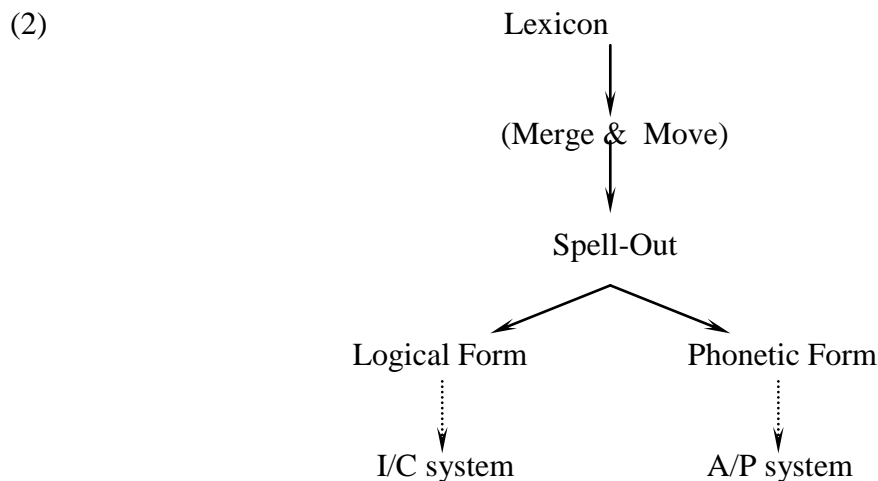
(1)
$$\text{Lexicon}$$
$$\downarrow$$
$$\text{D-Structure}$$
$$\downarrow$$
$$\text{S-Structure}$$

Logical Form   Phonetic Form
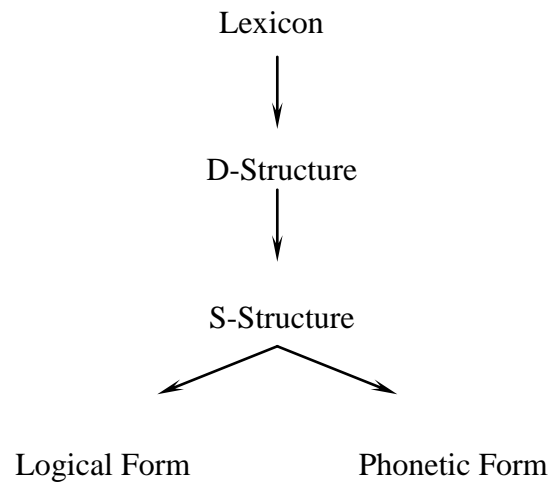
1.1. Virtual conceptual necessity: LF and PF

Language faculty is assumed to be embedded in performance systems: the so-called articulatory/perceptual (A/P) system and conceptual/intentional (C/I) system. These systems allow linguistic expressions to be articulated, interpreted, etc. As Chomsky suggests, we can take a linguistic expression SD to be a set of instructions for those systems containing indispensable information for them to function properly. It has already been established that SDs are sequences of representations, one at each postulated linguistic level. The aforementioned performance systems map into two linguistic levels, those of PF and LF respectively and constitute a part of the so-called virtual conceptual necessity, i.e. the simplest language design would opt for having only the conceptually necessary *interface levels* that would furthermore be the only linguistic levels. Taking this as a relevant assumption, we reached points (a) and (b) quoted in the introduction. Accordingly, each linguistic expression constitutes a pair (π, λ) (π at PF and λ at LF), where π is an abstract representation of sound and λ an abstract representation of meaning, both meeting the conditions holding at the interface. Taking the above idea as the core of the Minimalist Program, Chomsky dispenses with two levels of syntactic representation, namely: D-Structure and S-Structure. He claims that while PF and LF have external motivation and are thus seemingly ineliminable, D-

Structure and S-Structure have only theory-internal motivation and, hence, can be made redundant as superfluous. Naturally, such a step calls for empirical justification and this is going to be provided in the forthcoming sections. The Minimalist Program version of the language model looks then as follows:

(2)   Lexicon

                                       ↓

                            (Merge & Move)

                                       ↓

                              Spell-Out

                        ↙   ↘

            Logical Form   Phonetic Form

                   ⋮   ⋮

              I/C system   A/P system

1.2. Eliminating D-Structure

If we recall the EST model, D-Structure is placed between the Lexicon and S-Structure. It is the internal interface level between the Lexicon and the computational system. The standard assumption was that lexical items were drawn from the Lexicon via an operation that can be called *Satisfy* and mapped onto D-Structure in the form compatible with the X-bar format. The motivation for D-Structure's existence lies in certain UG principles that are supposed to hold at this level, namely the well known Projection Principle and θ-Criterion. If we decide to get rid of the D-Structure level, we must prove that both principles mentioned above are applicable elsewhere, in our case, probably at the level of LF.

The D-Structure's *"all-at-once"* operation *Satisfy* draws from the lexicon selecting an array of lexical items. Chomsky stresses the difference between an array and a set, as it is the former and not the latter that is selected from the lexicon. The difference is crucial, i.e. an array can result in various linguistic expressions depending on the arrangement

of the partaking items, whereas a set already constitutes an arrangement. This array is then presented in a form accessible to the computational system which Chomsky (1993) takes to be a format compatible with X-bar Theory. Bearing this in mind, we move on to the subject of Projection Principle (defined under (3)) that is a D-Structure phenomenon. In short, the principle states that representations at each syntactic level are projected from the lexicon in such a way that subcategorisation and θ-marking properties are maintained throughout the derivation, that is, in other words, every node that is present at one level of syntactic representation, must be present at all other levels.

(3) *Projection Principle*

    Lexical information is syntactically represented.

Thus, if our all-at-once operation *Satisfy* presents a given array of lexical items in the X-bar format, hence introduces nodes into the derivation, it follows from the Projection Principle that the whole structure, also containing empty positions generated for the elements to be moved later in the derivation should be built immediately at D-Structure and X-bar Theory should be made inaccessible throughout the derivation.

(4) a. S-Structure: *[IP We[VP wonder [CP whether [IP this book [VP was stolen t]]]]]*
    b. D-Structure: *[IP We[VP wonder [CP whether [IP e [VP was stolen this book]]]]]*

Such a state of the matters faces empirical problems and is highly undesirable in the Minimalist approach. Chomsky resigns from the operation *Satisfy* and moves towards a theory of Generalized Transformations whose workings along with operations Merge and Move are to replace the previous problematic *Satisfy* and are applicable throughout the derivation having access to Lexicon by Spell-Out (the details of the Minimalist operations are going to be given in section 2). It appears that we do not need D-Structure to have access to X-bar Theory as it is accessible at all levels of syntactic representation. In view of these facts

we can dispense with the Projection Principle holding at D-Structure and, thus, with this aspect of D-Structure as well.

In the Government and Binding Theory (GB) and Principles and Parameters (P&P) frameworks the θ-Criterion was assumed to hold at D-Structure. This fact was also to account for the postulate of D-Structure. The θ-Criterion is a principle regulating the assignment of the so-called thematic roles to the arguments of predicates. The θ-role assignment applies on one-to-one basis, namely, a predicate can assign one and only one thematic role to a given argument, and a given argument can bear one and only one thematic role[1], moreover it is necessary for arguments to be assigned thematic roles, and it is necessary for predicates to assign the roles they have at their disposal. Chomsky points to the problematic issues if the θ-Criterion is to hold exclusively at D-Structure. He gives an analysis of complex adjectival constructions such as the following:

(5) a. *John is easy to please*

b. *John is easy [CP Op [IP PRO to please t]]*

c. *It is easy to please John*

The representation under (5b) is the assumed S-Structure, the problem being, however, that *John* finds itself in a position where no θ-role assignment can take place, *John* gets its θ-role from *please* in the same position where *John* finds itself in (5c). The trace within IP is the trace of the empty operator, not *John*. (5b) seems to be problematic not only with respect to θ-Criterion; *John* seems to have moved from a Case-position to another Case-position, both A-

---

[1] It has been suggested in the literature that it is not always the case that arguments bear just one thematic role. In sentences such as (1) below NP *the house* seems to be assigned two θ-roles, one from the verb and the other from the adjective:

(1) John painted the house red.

The Uniqueness Condition imposed by the θ-Criterion appears to be violated, unless we propose some *pro* element that could be related to the NP in question and bear the second θ-role.

positions, via an intermediate A'-position forming an improper chain. Moreover, if the trace within the IP is left by the empty operator, it is a variable[2]. Variables are said to behave like R-expressions with respect to Binding Theory, meaning: variables must not be bound. The positioning of the variable and *John* should lead to the Principle C violation, as *John* c-commands the variable it is co-indexed with. The construction (5b) violates the θ-Criterion, Relativized Minimality (Rizzi 1990) and principle C of BT, but still is undoubtedly grammatical. We are led to believe that, at least with respect to the θ-Criterion there must be some other chance, perhaps somewhere else, for the θ-role to be assigned. This is a most desirable conclusion for the Minimalist Program followers. As the remnants of syntactic levels of representation is here restricted to Spell-Out (the Minimalist counterpart of S-Structure) and LF, it is most likely that the level where the NP *John* will eventually be assigned its thematic role is the interface level of LF. In the introduction to *Minimal Ideas* (Abraham et al. 1996), it is noted as well that the θ-Criterion, being a condition on interpretation, should apparently apply at LF on conceptual grounds. Dispensing with θ-Criterion at D-Structure, we can again suggest dispensing with D-Structure itself.

Both the Projection Principle and the θ-Criterion constituted the theory-internal (and only) motivation for the existence of D-Structure. If we can prove that these conditions are applicable at some other linguistic level, most desirably LF in Minimalist terms, we can just as well get rid of D-Structure. Chomsky, however, tries to avoid strong statements and concludes saying that "the empirical consequences of the D-Structure conditions remain to be faced" (Chomsky 1993, in Chomsky 1995: 188), thus leaving the question open to further research and discussion. Chomsky's tentative conclusion seems justified especially that he

---

[2] Wh-words and quantified NPs are not arguments, yet they find themselves in positions typical of arguments. Ouhalla (1999) suggests that QPs after Quantifier Raising at LF leave a variable behind and, hence, become potential arguments. The problem for the θ-Criterion applicable at DS is then straightforward: at DS all the QPs are still in their base positions, hence are not arguments and cannot bear a θ-role. It is more plausible then for the θ-Criterion to apply later in the derivation, most desirably at LF.

continues to claim that thematic roles are assigned locally within VP (situated closely to the predicate), very early on in the derivation[3]. This state of the matters has a flavour of D-Structure and VP seems to be the residue of D-Structure within the Minimalist Program, hence not fully dispensed with leaving a question of its existence open.

1.3. Eliminating S-Structure

The level of S-Structure, analogously to D-Structure, cannot boast its existence within the Minimalist framework. In the Minimalist Program S-Structure is replaced by Spell-Out - an optional rule that can apply at any point in the derivation. The motivation for the S-Structure level was, just like in the case of D-Structure, only theory-internal, namely, restricted to the conditions that were assumed to apply on this particular level of representation: Case Theory and Binding Theory. The conditions holding at S-Structure found support in two kinds of evidence, i.e., to quote from Chomsky (1995: 191):

(6) a. *Languages differ with respect to where Spell-Out applies in the course of the derivation to LF.*

    b. *In just about every module of grammar, there is extensive evidence that the conditions apply at S-Structure.*

Chomsky proves the above evidence insufficient and turns it against S-Structure, that is, it is enough to prove conditions holding at S-Structure satisfiable at the interface levels (LF & PF) and the elimination of this level becomes unproblematic. Bearing this in mind, he suggests that the point in the derivation where Spell-Out applies must be determined by either PF or LF, Spell-Out being a conceptually necessary operation, i.e. one that results in two indispensable representations: that of sound and that of meaning. Moreover, a strong

---

[3] First we apply Merge to use up the contents of the initial numeration (NI), only then are we allowed to perform movement.

Minimalist assumption is that differences between languages amount to morphological variation reflected at PF, whereas at LF languages are said to be alike.

It is worth while analysing Chomsky's arguments against S-Structure, excluding application of the conditions at this level if only there is a possibility of satisfying them elsewhere:

(7) a. *The condition in question **can** apply at LF alone.*

   b. *Furthermore, the condition sometimes **must** apply at LF.*

   c. *Furthermore, the condition must **not** apply at S-Structure.*

Apparently, even the weakest argument (a) is enough to deny motivation for S-Structure's existence. Let us now turn to the evidence for S-Structure as given under (6) and show on the basis of empirical data and the above assumptions in (7) how the level of S-Structure can be dispensed with.

1.3.1. Spell-Out application

In the Minimalist framework languages are said to differ with respect to the moment of Spell-Out application. Considering the position of the verb, Spell-Out representations from French-type languages and English-type languages provide us with interesting empirical data. It has been vastly accepted in the generative approach that in French the verb moves overtly and that in English such movement takes place in covert syntax. Consider the following examples (taken from Ouhalla, 1999)

(8) a. *John often kisses Mary.*

   b. *[$_{IP}$ John [$_{I'}$ t$_I$ [$_{VP}$ often [$_{VP}$ [$_V$ kiss [I]] Mary…*

The French parallel structure is ungrammatical:

(9) a. * *Jean souvent embrasse Marie.*

      Jean   often   kisses   Marie

  b. *[IP Jean [I' tI [VP souvent [VP [V embrasser [I]] Marie…*

What is grammatical in French, however, is ungrammatical in English:

(10) a. *Jean embrasse souvent Marie.*

      John   kisses   often   Marie

   b. *[IP Jean [I' [V embrasser] I [VP souvent [VP tembrasser Marie…*

(11) a. * *John kisses often Mary.*

   b. *[IP John [I' [V kiss] I [VP often [VP tkiss Mary…*

Translating this into the Minimalist Program terms, we assume that the main verb in French-type languages performs movement before Spell-Out and its English counterpart performs this movement after that point. Going further, we can account for the differences on the basis of strength of features that are said to drive movement in the Minimalist approach. If features are strong, their checking will force overt movement, if they are weak, the movement will take place covertly. Operations applied after Spell-Out are said to be less costly, why then different languages use different options? Why is this the case that French chooses a more costly option? As languages are said to be alike at LF in the approach presented here, different options used by French and English are probably determined by the PF component. A French sentence with a verb that did not undergo movement to check its agreement features is considered an illicit PF object, causing the derivation to crash. In English this movement is allowed to take place later (Procrastinate), hence it can wait till LF. All this is attributed to the strength of features of Agr (agreement phrase); while it is said to have strong features in

11

French, it bears weak features in English[4]. So, even though French uses a seemingly more costly option, it is not the case: what we compare are convergent derivations, and the one without overt V-raising in French is not admissible. A similar problem arises with wh-phrases and their positions across languages. In English wh-phrases are fronted when there is only one in a sentence, if there are more, only one is fronted and the rest remains in situ; Turkish, Japanese and Chinese do not front any of the wh-phrases (examples taken from Huang, in Webelhuth 1994), while languages like Polish allow fronting of all the wh-phrases.

(12) a. *Kogo   widział Janek?*
      whom   saw   Johnny
    'Who did Johnny see?'

  b. *Kto co   komu   kupił?*
     who what   for whom   bought
    'Who bought what for whom?'

(13) a. *Who did John see t ?*

  b. *What does John think Mary bought t?*

(14) *Zhangsan   yiwei Lisi   mai-le   shenme?*
   Zhangsan thinks Lisi   bought   what
   'What does Zhangsan think Lisi bought?'

The answers to the questions will be similar to those about verbs, namely, covert and overt movement of phrases and strength of the [+Q] feature carried by the wh-phrase. We will elaborate on such matters and the conditions holding at LF in the section on economy principles.

1.3.2. Binding Theory Conditions and Copy Theory of Movement

---

[4] The agreement and Case features are to be found within Agreement, Tense and Verb phrases, hence we have triple placement of Φ-features within the structure. An NP Subject, for instance, to yield its overt form has to enter two kinds of structural relations: a Case relation with [T AgrS] V], and an agreement relation with AgrS (again the same complex [T AgrS] V]).

Binding Theory (BT) has been widely assumed to apply at S-Structure in the previous frameworks. As quoted above, there is a good deal of evidence for such a state of the affairs. Eliminating the S-Structure level means finding empirical justification for application of this module at the interface level of LF. Let us first consider the data concerning the Condition C effects of Binding Theory; the relevant examples are under (15) (Chomsky, 1993):

(15) a.   *you said <u>he</u> liked [the pictures that <u>John</u> took]*

  b.   *[how many pictures that <u>John</u> took] did you say <u>he</u> liked t*

  c.   *who [t said <u>he</u> liked [$_\alpha$ how many pictures that <u>John</u> took]*

These examples were regarded as evidence for the application BT mainly at S-Structure, now Chomsky uses them to prove just the opposite. In (15a) *John* cannot be an antecedent of *he,* the latter c-commands *John* and the coindexation of the two elements would lead to the Condition C violation. We do not encounter such problems in our example (15b) where *John* and *he* can be coreferential. Here the S-Structure configuration is such that *John* is removed from the c-command domain of *he*. In (15c) *John* again finds itself in a configuration that would give rise to the Condition C violation if it constituted an antecedent of *he*. This example was used as a crucial argument against BT at LF (Chomsky, 1981); if LF movement displaces entire constituents to [Spec, CP], *John* should be placed out of the c-command domain of *he* again:

(16) *[[how many pictures that John took] who [t said he liked t']*

LF wh-movement should bleed Principle C, but it does not. The conclusion drawn in the GB theory was that BT applied at S-Structure and LF displacement was too late. Chomsky, however, changes his standpoint and suggests that Quantifier Raising only moves *how many* at LF and adjoins it to *who*, and the relevant structure would look as the one under (17):

(17)   *[[how many]$_j$ who$_i$] [t$_i$ said he liked [[t$_j$ pictures] that John took]]*

The above LF representation apparently shows the same relations holding between *he* and *John* as in (15a) and (15c). Taking also another Chomsky's suggestion into account, namely: the level at which *he* c-commands *John* will be the level at which Condition C applies, we now see that S-Structure loses its exclusiveness in the case of Condition C application and, at the same time, loses its credibility. If we now recall arguments given in (7), our conclusion can be qualified as the first type of arguments against S-Structure, the one allowing the condition in question to apply at LF.

Another set of data suggested in Chomsky's MPLT concerns the Condition A of BT. These data turn out to be even more compelling, consider the following examples:

(18)   a. *John wondered which picture of himself Bill saw.*

b. *John wondered [which picture of himself] Bill saw {[which picture of himself]}*

The structure under (18b) is an LF representation of (18a). Chomsky treats movement as copying and deletion, namely, every performed movement is said to leave a copy of the moved element in its base position. All copies are kept till LF for interpretation purposes, at PF, however, the situation is different: here, only one copy of a given element is allowed, the rest must delete before entering PF and the one that is left is spelled out. Exchanging traces for copies we also deny existence of all principles applicable to traces, thus a principle of crucial importance like ECP (Empty Category Principle; demanding all traces to be properly governed). ECP gave neat explanations to many facts, now these facts will have to be accounted for by some other mechanisms or phenomena minimalist in spirit. Chomsky also dispensed with the notion of government which was necessary for ECP[5]. Copies being

---

[5] ECP accounted for the widely known *that-trace effect* with respect to the subject/object asymmetries. The following examples illustrate the phenomena (taken form Ouhalla 1999):

different creatures than traces will obey different principles, especially that the latter undergo

deletion (only one copy is spelled-out). Multiple copies in (18b), a result of reconstruction,

make interpretation a bit more complicated, that is to say, we have two possibilities of

analysing the wh-phrase and, depending on the choice, we will have two candidates for

binding the reflexive pronoun: *John* and *Bill*. Were the Condition A of BT to apply solely at

S-Structure (without LF reconstruction), we would have no problems deciding on the

antecedent of the reflexive. Wh-phrases undergo further movement at LF, just like in (19),

and again we have two options:

(19)   a.   *John wondered [[which picture of himself] [$_{wh}$ t]] [Bill saw*

[[which picture of himself] [$_{wh}$ t]]]

b.   *John wondered [which][$_{wh}$ t picture of himself] [Bill saw*

[which][$_{wh}$ t picture of himself]]]

Let us now interpret the representations in (19) as operator-variable constructions, following

Chomsky (1993):

(20)   a.   *John wondered [which x, x a picture of himself] [Bill saw x]*

b.   *John wondered [which x] Bill saw [x picture of himself]*

---

(1)   a. *Which car did you say (that) John would fix t?*
     b. *Who did you say (*that) t would fix the car?*

Example (1a) shows that the object of the verb *fix* can be extracted from the embedded clause without any
problems. Even though the antecedent of the object is too far away to antecedent-govern the object trace, the
trace is still properly governed (head governed) by the verb. In the case of the subject, the problem is that the
subject trace would not be properly governed at all. The antecedent of the subject is too far and there is no such
option as the object has, i.e. there is no lexical head to properly (head) govern the subject trace. However, if the
complementizer *that* would not be there, the structure would be saved and the trace governed properly, CP not
constituting a barrier for outside government then.
Let us consider yet another example with a strong ECP violation; here we face an extraction of An adjunct out of
an island:

(2) **How do you wonder [$_{CP}$ whether[$_{IP}$ John fixed the car t]]*

The only chance for an adjunct to be properly governed is to be antecedent governed, nevertheless, due to the
presence of an IP preceded by a CP (constituting a barrier), this option is made unavailable.
The cases above used to be accounted for by the ECP, in the MPLT, however, they are left out without
explanation.

Movement is copying and deletion, we have dealt with the copying part, now the time has come to face the deletion problems. Depending on which copy we decide to delete, we will yield an interpretation favouring either *John* or *Bill*. It is only thanks to LF reconstruction that both interpretations are available.

Chomsky presents yet another analysis of the above examples in his MPLT. Here, we witness another movement: the reflexive pronoun moving out of the wh-phrase. Deletion of copies is now constrained by the θ-Criterion. If the θ-Criterion is to be defined over chains (i.e. a chain must bear a θ-role), deleting copies constituting part of some chain, might result in depriving these chains of their θ-positions. Consider the relevant examples:

(21)   a. *John [self -wondered [which picture of t_{self}][NP saw [$_{TR}$ which picture of himself]]*

   b. *John wondered [which picture of himself [NP self -saw [$_{TR}$ which picture of t_{self}]]*

In (21a) we are allowed to delete the lower copy and in (21b) the higher copy, otherwise the θ-criterion is violated.

Idiomatic expressions appear to be an interesting issue with respect to the above analysis. Let us replace the verb *see* in (18) by a verb *take*, for instance. The resulting structure shows even more ambiguity, that is, the expression *take a picture* can be interpreted either literally, or idiomatically. Chomsky shows that it is only thanks to reconstruction at LF that both interpretations are available. However, it has been also suggested that it is the application of the θ-Criterion that additionally contributes to this result. Let us present both analyses:

(22) a. *John wondered [which picture of himself] [Bill took t]*

   b. *John wondered [which picture of himself] [Bill took [which picture of himself]]*

The representation in (23), just like in previous examples, shows the structure after the wh-phrases have performed movement at LF:

(23) a. *John wondered [which picture of himself][$_{wh}$ t] [Bill took*

   [which picture of himself] [$_{wh}$ t]

   b. *John wondered [which][$_{wh}$ t picture of himself] [Bill took*

   [which][$_{wh}$ t picture of himself]]]

It is enough to look closely at the operator-variable construction given under (24) to see that only in the case (24b), and not in the case (24a), is the idiomatic reading possible. Moreover, the idiomatic interpretation is available only with one of the possible antecedents for the reflexive pronoun, namely *Bill*, and not *John*. Chomsky claims that idiomatic interpretation is only available with parts of the idiomatic expression being in a local relation at LF. Under reconstruction this is achievable: *take* and *picture* are in a local relation with respect to each other, and the closest possible antecedent for *himself* is *Bill* ; consider (24):

(24) a. *John wondered [which x, x a picture of himself] [Bill took x]* - literal reading

   b. *John wondered [which x] Bill took [x picture of himself]*   - idiomatic reading

If it were not for reconstruction, then, the idiomatic reading would be unavailable, as the only antecedent for the reflexive at S-Structure is *John*. This appears to be a strong argument against BT applying at S-Structure and against S-Structure in general.

Now let us have a look at the analysis based on the θ-Criterion:

(25) a. *John [himself$_i$ [wondered [which picture of t$_i$ [Bill took which picture of himself]]]*

   b. *John wondered [which picture of himself [Bill himself$_i$ took which picture of t$_i$]]*

The θ-Criterion requires deleting the lower copy in (25a), hence only *John* is available as an antecedent for *himself*. In this case, however, the idiomatic reading is inaccessible. If we want the idiomatic interpretation, we need a structure like (25b). As we see, both analyses, even though based on completely different requirements, bring identical results. What is more, they prove S-Structure redundant. Reconstruction being unavailable at S-Structure shows that in certain constructions LF is the only level where Condition A can apply. This is an argument of the type (7b) and (7c) of Chomsky, enough to eliminate S-Structure. S-Structure is in no way exclusive with respect to BT application, and, as this used to be an argument motivating its existence, it appears to be an exceptionally weak one in view of the discussed facts.

1.3.3. Case Theory (Case Filter)

Case Filter has been for years an unquestionable S-Structure phenomenon and, thus, a strong argument for its existence. Just like in the case of Binding Theory, it must be proved that it is not necessary for Case to be assigned at S-Structure, and it can just as well be assigned elsewhere. Many a time, it actually must be assigned elsewhere, as the S-Structure configuration would in no way allow for Case assignment, consider:

(26) a. *There is a book on the table.*

   b. *[$_{IP}$ There [$_{I'}$ 3 sg [$_{VP}$ is [$_{NP}$ **a book**][$_{PP}$ on the table]].*

Case assignment in the GB and P&P approaches could only take place in two configurations: under government, under agreement (in a spec-head configuration involving a functional head). There were three possibilities of structural Case assignment, these include: Nominative Case assignment (under agreement), Accusative Case assignment (verb to object under government) and Exceptional Case Marking (verb of the matrix clause assigning Accusative under government to the subject of the embedded clause), consider the examples:

(27) a. *[IP [NP John] [I' -ed [VP kick [NP the ball]]]]*

           NOM   ACC

(28) a. *Bill considers [SC John / him /(*he) intelligent]*   (ECM)

        ACC

In the above example (26) none of the presented configurations is available at S-Structure. The sentence should be rendered ungrammatical as our NP *a book* does not seem to get Case. This clearly is not the case. Chomsky proposes a different analysis of Case phenomena and proves Case Filter a condition on LF representations. He dispenses with the notion of head-government and proposes a more unified approach to Case-assignment, namely all Case features are checked in spec-head configuration. Moreover, we are not talking about Case-assignment any more, in the Minimalist Program we are dealing with a phenomenon referred to as Case-checking, Case being a formal feature that along with other features undergoes checking (we will elaborate on the details of Case-checking in the section on Checking Theory). Our NP *a book* in (26) will check its Case feature covertly against T(ense) bearing Nominative Case feature, all this in a spec-head configuration. The same phenomenon accounts for ECM constructions just like the one above. In earlier frameworks NPs *John/him* have been assumed to occupy an Object position and assigned Accusative Case. In the Minimalist framework these NPs are said to perform movement to the specifier of AgrOP where they check their Accusative Case feature against the verbal head in Agr complex. The Minimalist version of the above examples (27a) and (28a) is showed under (27b) and (28b) respectively:

(27) b. *[AgrSP John [TP kicked [AgrOP the ball [VP tJohn [V' tkick [NP tball]]]]]]*

(28) b. *[AgrSP Bill [TP considers [AgrOP John/him/*he [VP tBill [V' tconsider]]] [AgrAP tJohn/him [AP*

    intelligent]]

There is no need to postulate a special level for Case Filter to apply if it can easily apply at LF. The above data give just an overview of the proposal as the details are still to come, nevertheless, the evidence seems compelling.

We assume now that we have successfully dispensed with two problematic levels, i.e. D-Structure and S-Structure. Both levels had only theory-internal motivation and it was the goal of this section to prove it insufficient and dispensable. As the previously D-Structure and S-Structure phenomena find their rescue at the LF interface level, we are led to believe that both levels are redundant. This conclusion has been proved and the goal allegedly attained.

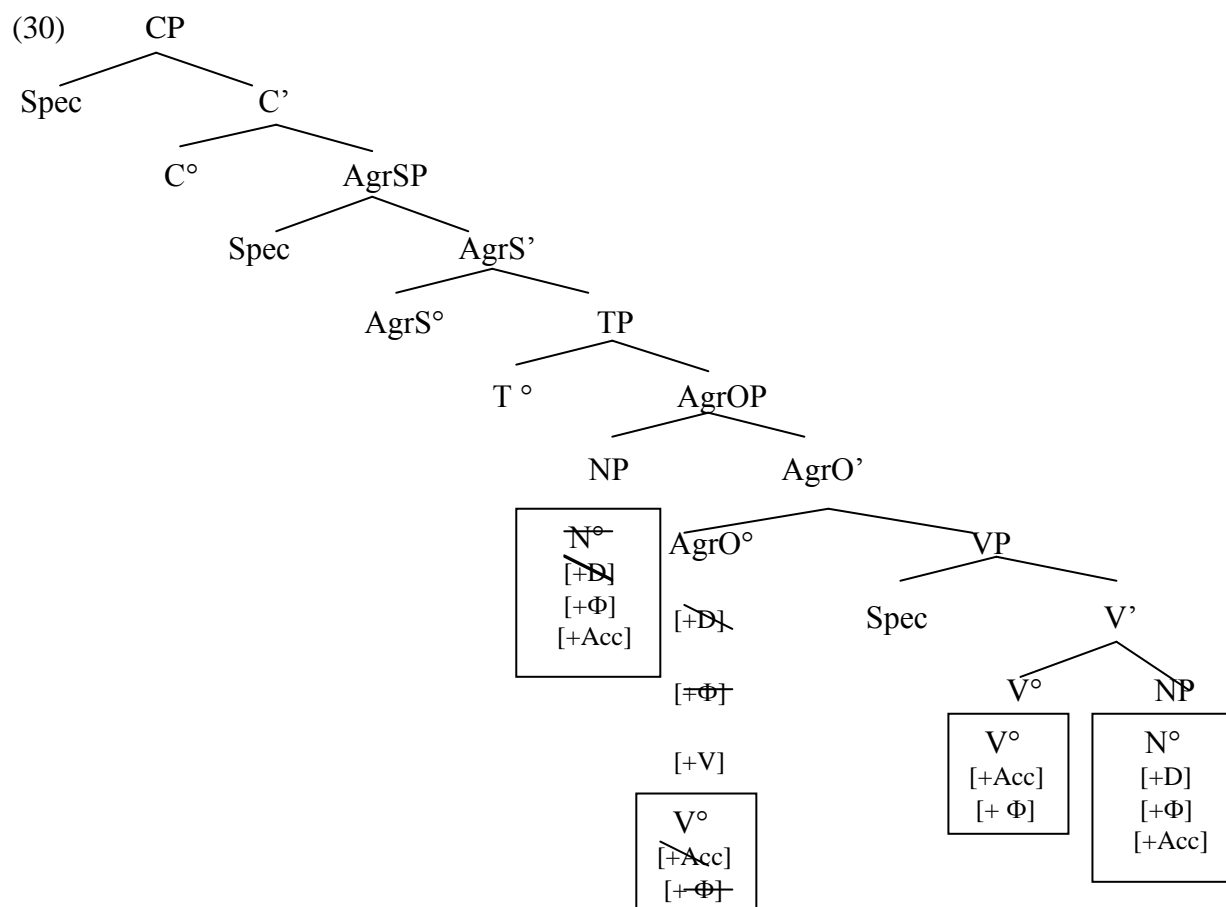2. X-bar Theory, phrase structure and the computational system

"A language consists of a lexicon and a computational system […] The computational system takes representations of a given form and modifies them. Accordingly, UG must provide means to present an array of items from the lexicon in a form accessible to the computational system. We may take this form to be some version of X-bar Theory. The concepts of X-bar Theory are therefore fundamental. In a minimalist theory, the crucial properties and relations will be stated in the simple and elementary terms of X-bar Theory." (Chomsky 1993)

2.1. Basic clause structure and the concept of domains

Clause structure within the Minimalist Program obeys X-bar theoretic relations, hence we continue with the well-known concepts of specifier, head and complement. The basic X-bar structure looks as follows:

(29)
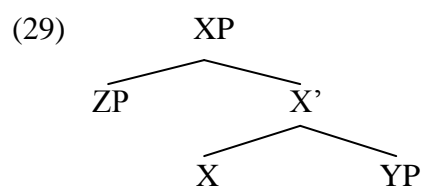
Here XP is a maximal projection, X' an intermediate projection, ZP constitutes a specifier, X is a head and YP is a complement. Chomsky enumerates two most important local relations, namely: spec-head relation (ZP to X) and head-complement relation (X to YP). The spec-head relation , as already mentioned in the preceding section, is important with respect to Case and Agreement features checking, the head-complement relation, on the other hand, plays an important role in θ-role assignment (Chomsky does not elaborate on where exactly this assignment should take place; it should be by now applicable at LF, nevertheless, arguments find themselves in most convenient positioning for θ-role assignment before Spell-Out, that is in most local relation to the predicate assuming PISH, making VP the residue of DS). Chomsky (1993: 173) collapses Pollock's (1989) and Belletti's (1990) Split Infl hypotheses and takes the basic clause structure to look as given below:

(30)

CP
├── Spec
└── C'
    ├── C°
    └── AgrSP
        ├── Spec
        └── AgrS'
            ├── AgrS°
            └── TP
                ├── T °
                └── AgrOP
                    ├── NP
                    │   └── [N° / [+D] / [+Φ] / [+Acc]]
                    └── AgrO'
                        ├── AgrO°
                        │   [+D]
                        │   [+Φ]
                        │   [+V]
                        │   └── [V° / [+Acc] / [+Φ]]
                        └── VP
                            ├── Spec
                            └── V'
                                ├── V°
                                │   [V° / [+Acc] / [+ Φ]]
                                └── NP
                                    └── [N° / [+D] / [+Φ] / [+Acc]]

21

The above structure can be still supplemented by Neg(ation) Phrase and a specifier position of T(ense). Specifier of TP constitutes an optional position for subject Case-checking in Transitive Expletive Constructions in Scandinavian languages exhibiting overt verb movement.

Chomsky argues that it is necessary to postulate agreement phrases: AgrSP and AgrOP, for subject and object respectively. Agreement phrases are collections of the so-called Φ-features (i.e. features specifying gender, number and person) and are necessary participants in Case-checking strategy assumed in the Minimalist Program, meaning: they provide slots for the spec-head relation in which Case-checking is allowed to take place. Subject Case-checking takes place in AgrSP, subject moves from within VP (from its base position assuming PISH) and checks its Nominative Case-feature against the head T in AgrS; object Case-checking works in a similar fashion, that is to say, object moves from within VP and substitutes into the specifier position of AgrOP, the verb moves then and head-adjoins to AgrO (as shown in (19); Agr bears two types of features, thus features find themselves in three different places: in N°, V° and AgrO°). Minimizing the possibilities of Case-checking to just one configuration seems successful and a most desirable result in the Minimalist Program. The elements undergoing movement are constrained by the so-called Shortest Move Condition, movement for Case reasons is to obey the condition as well. Shortest Move requires that the categories move to the first available position of the needed type (similarly to Rizzi's (1990) Relativized Minimality). Let us look at the structure under (20) showing the mechanics of Case-checking:
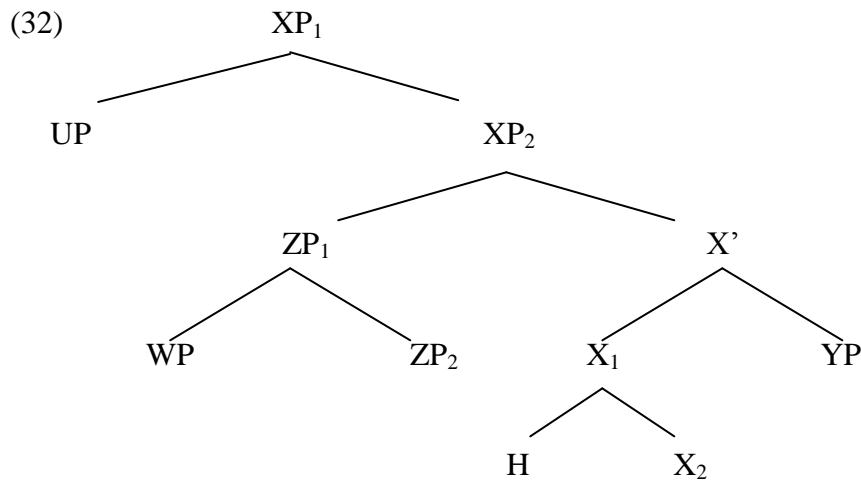
(31)
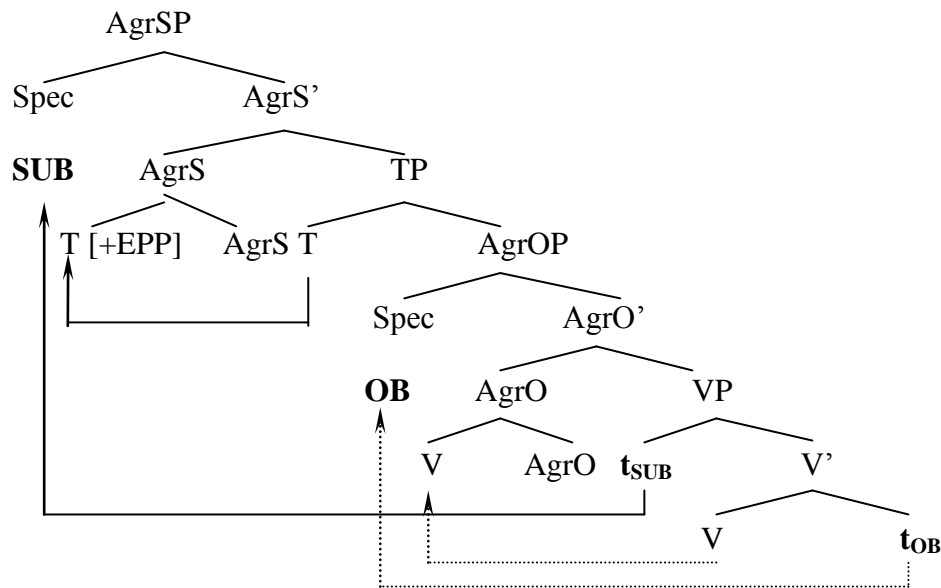
There is evidence that subject Case is actually checked against the verbal complex [V AgrO[T]AgrS], this would require verb movement to T before T performs its movement to AgrS. The only chance for the object to have its Case checked is to move first to [Spec, AgrOP] (over the subject) and check its Case against the complex [V AgrO], then subject moves to [Spec, AgrSP] (over the object, object unable to move now having all its features checked) and check its Case against the complex [V AgrO]T]AgrS][6]. The crossing movements of subject and object over each other raised questions, namely: how does the system differentiate between the two NPs? Why is it allowed for the NPs to skip a closer position, a case of superraising, hence a violation o the Shortest Move in Minimalist terms? The answers become straightforward after investigating Chomsky's concept of domains and Equidistance to come.

Chomsky takes the concepts behind X-bar Theory to be fundamental; he assumes binary branching to be the only option and takes segment/category distinction to hold in overt

---

[6] Chomsky notes that if we actually allowed the subject to move from the VP internal position to the specifier of AgrOP it would block any kind of Case assignment to the object and the object would be "frozen in place". In footnote 22 (Chomsky 1993), however, he notes that in such a case our subject would still have to raise to [Spec, AgrSP] in a Nominative-Accusative language. This suggests that languages preferring and apparently using such an option could be accounted for in this framework via the same logic. Such languages are the ergative ones in which the ergative arguments exhibit the syntactic behaviour of direct objects in Nominative languages.

syntax in the cases of adjunction that is said to have a *"structure-preserving" character*. Examine the structure below for relevant relations:

(32)

```
                    XP₁
                  /    \
               UP       XP₂
                       /    \
                   ZP₁       X'
                  /   \     /   \
               WP    ZP₂  X₁    YP
                         /  \
                        H    X₂
```

The maximal projection XP in the above structure has a specifier ZP, a head X and a complement YP. We have three instances of adjunction here: UP to XP (maximal projection to a maximal projection), changing XP into a two-segment category $\langle XP_1, XP_2 \rangle$; similarly WP to ZP resulting again in a two-segment category $\langle ZP_1, ZP_2 \rangle$, making the internal structure of the specifier more complicated; H to X (head-to-head adjunction) turning X into $\langle X_1, X_2 \rangle$. According to Chomsky, in the Minimalist framework the basic elements constituting representations are chains, and these are chains that enter into relations and have domains defined over them (either chains (α, t) or trivial chains (α)). Before we go further and present definitions of domains, let us introduce two relevant concepts, namely: *dominate* and *contain*:

(33)   *A category α dominates β if every segment of α dominates β.*

(34)   *A category α contains β some segment of α dominates β.*

In the above structure XP only contains UP, but does not dominate it, the same holds for ZP with respect to WP and X with respect to H; XP dominates ZP, WP, X, H and YP. These

relations are crucial when defining domains. Following Chomsky (1993, in Chomsky 1995:

178), we give definitions of domains and necessary related concepts:

(35) a. *Max (α) is the smallest maximal projection dominating α.*

    b. *The <u>domain</u> of a head α is the set of nodes contained in Max(α) that are distinct*

       from and do not contain α.

    c. *The <u>complement domain</u> of a head α is a subset of the domain reflexively dominated*

       by the complement of the construction.

    d. *The domain of a head α minus its complement domain will be called the <u>residue</u> of α.*

    e. *The <u>minimal domain</u> is the smallest subset K of the domain of α such that for any γ*

       belonging to the domain, some β belonging to K dominates γ.

    f. *The <u>minimal complement domain</u> of α is called the internal domain and the <u>minimal</u>*

       <u>residue</u> of α is called its checking domain.

As already mentioned, Chomsky defines domains over chains. The chains are of two forms:

two-membered, nontrivial, chains (α, t) or one-membered, trivial, chains of the form (α). An

important matter is that domains are defined once and for all, hence it is suggested that they

should rather be understood "derivationally, not representationally: defined for α as part of

the process of introducing α into the derivation. If α is a trivial chain, then Min(S(α)) is

defined when α is lexically inserted; if α is a nontrivial chain ($\beta_1,\ldots,\beta_n$), then Min(S(α)) is

defined when α is formed by raising β_1" (Chomsky 1993, in Chomsky 1995: 179). The idea

of domains, especially checking domain, is essential in feature-checking. The checking

domain provides necessary spec-head relation in which feature-checking is only allowed. Let

us now come back to our former example concerning Case feature-checking by both subject

and object and the problematic, but necessary, movement of the object over the subject and vice versa:

(36)
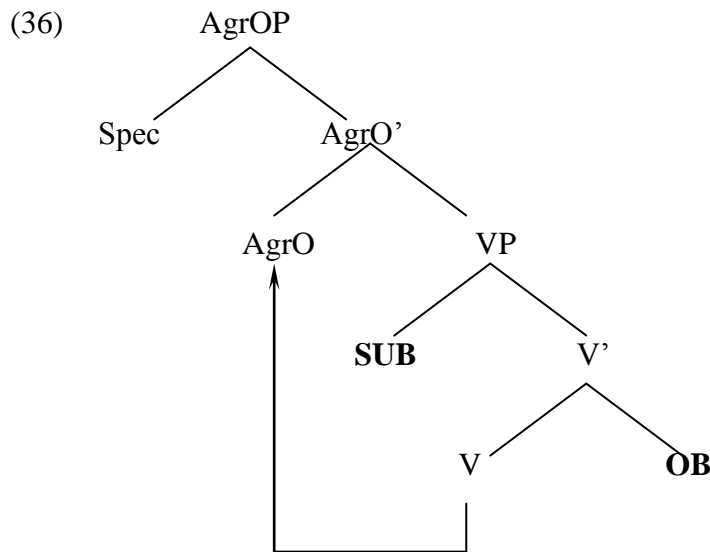
The movement of the verb to AgrO, where it head adjoins, extends the verbal domain. Now, we are talking about the domain of the chain CH=(V, t_V). The domain of that chain includes both subject (the competing category) and [Spec, AgrOP] (the target position), and, thus, both subject and object are said to be equidistant from the target position [Spec,AgrOP], hence not really skipping any positions and not at all violating Shortest Move that way. He grounds his definition of Equidistance on the concept of the minimal domain:

(37) *If α, β are in the same minimal domain, they are equidistant from γ.*

This means that both NPs are eligible for movement to that position. Chomsky, however, does not give any answer concerning these facts. It seems plausible with overt object movement, but not really satisfactory if object does not move overtly for Case-feature checking, when it can wait till LF (Procrastinate), which is apparently the case in English. He further notes that this analysis allows object raising only if the verb performs its movement,

which is confirmed empirically in the overt counterpart, namely in Icelandic object shift is possible only with the presence of overt verb movement[7].

2.2. The computational system

Language is said to consist of a lexicon and a computational system, both conceptually necessary. Computational system is to build representations out of the items it draws from the lexicon. Lexical items are equipped with three types of features: semantic, phonological and syntactic. The linguistic levels of representation available in the Minimalist Program are manifestations of these features: Logical Form, Phonetic Form and Spell-Out, respectively. The computational system forms derivations satisfying demands of each of the levels, that is to say, forming representations that converge at each of the levels consisting of legitimate objects. We are going to present the mechanics behind the computational system C_{HL} in the sections below. First, we are going to present the fashion in which phrase structure is built analysing relevant operations, then we are going to discuss the workings of the computational system in both overt and covert component that exhibits certain asymmetries.
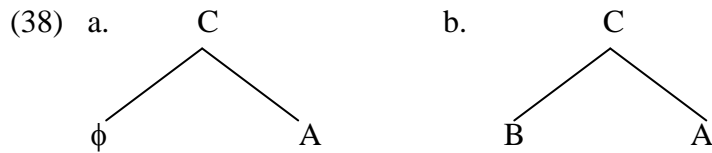
2.2.1. Generalized Transformations: Merge, Move and Form Chain and the Strict Cycle Condition

In the Minimalist Program Chomsky comes back to his previous GB assumptions on building phrase structure. Since the very much undesirable all-at-once operation *Satisfy* has been dispensed with, he proposes an alternative, a theory of Generalized Transformations. Generalized Transformations build structure in a bottom-up fashion, as opposed to earlier all-at-once top-bottom manner. Chomsky, apparently, restricts the mechanics to a single Generalized Transformation (GT) that is a binary substitution operation. This translates into a

---

[7] This refers to Holmberg's Generalisation stating that Object Shift is possible only if the main verb raises out of VP.

following situation: an operation GT takes two independent phrase markers A and B and substitutes one into a designated empty position of the other extending the target and eventually forming a new phrase marker C satisfying X-bar Theory:

(38) a.

```
        C                      b.        C
       / \                              / \
      /   \                            /   \
     φ     A                          B     A
```

Substitution is a binary operation as it is insertion of a new item into the structure, this operation is referred to as Merge in the Minimalist Program; there is another operation, a singulary one: it takes an item from within the phrase marker to substitute it into the empty position, this operation is called Move. Every application of GT results in a new phrase marker and takes place in the so-called successive-cyclic manner. The computational system keeps building the structure via application of GTs respecting the so-called Strict Cycle (or Extension) Condition (Chomsky, 1993: 190):

(39) *Insertion and movement are allowed only if they extend the targeted phrase marker.*

There is yet another operation proposed in the Minimalist framework, namely Form Chain. Up till now the basic transformational operation was Move α. This operation, however, is subject to restrictions such as economy principles. In our present discussion only two economy conditions on derivations within the Minimalist framework are relevant. These two are seemingly contradictory, namely: Shortest Move and Fewest Steps. The problem is as follows: keeping to Shortest Move where we want to make shortest steps (very much like in the case of Relativized Minimality), we need to make many steps while performing a long-distance movement; keeping to Fewest Steps, on the other hand, we do not want to produce many intermediate copies, skipping available positions on the way, rather performing movement in one fell swoop. It seems rather difficult to satisfy both, Chomsky, however,

finds a solution. He proposes that we should resign from Move α as a basic transformational operation, taking Form Chain[8] instead that forms structures in one step, consider (40):

(40) a. *e seems [e to be likely [John to win]]*

Form Chain:

b. *John seems [t' to be likely [t to win]]*

c. *CH = (John, t', t)*

---

[8] Examining the well known cases of superraising, Head Movement Constraint and wh-island condition violations, we see that in Minimalist terms these are violations of Shortest Move. Having Form Chain as an option now, we might escape some of these problems. It must be noted as well that Form Chain does not eliminate Move α from the theory. Poole (1996, in Abraham et al. 1996*)*, compares applications of Move α and Form Chain and suggests that only the latter is subject to economy conditions, while the former can be an instantiation of a cost-free movement if its application does not result in chain-formation. He gives the following example, where (a) has been formed by Form Chain and (b) by two applications of Move α:

(2) *Why do you think Bill hit Barney?*
(3) a. *[$_{CP}$ Why$_i$ do [$_{IP}$ you think [$_{CP}$ [$_{IP}$ Bill hit Barney t$_i$]]]*   (Form Chain)
    b. *[$_{CP}$ Why$_i$ do [$_{IP}$ you think [$_{CP}$ t'$_i$ [$_{IP}$ Bill hit Barney t$_i$]]]*   (Move α)

Both operations form chains, the difference being that the first one performs Move α twice and the second one only once. Here both Move α and Form Chain are subject to economy conditions because they both form chains. It seems, however, that Form Chain would always be more economical than Move α in such cases. The representation in (a) violates the so-called Minimal Link Condition, and is, thus, ungrammatical. Both derivations are said to converge, though. What we compare here are convergent most economical derivations, hence, (a) would be an appropriate candidate to choose as the most economical one, the point is that (b) does not violate any principles of grammar, and, even though demands more effort it makes it more eligible.

        Poole (1996) raises the subject of optionality of movement. He takes two convergent derivations one where movement of an element takes place and the other where there is no such movement and the element remains *in situ*. As both derivations are considered equally economical, it is suggested that the one applying Move α must be costless. According to Poole, Move α is cost-free only when it does not form chain. This means that the moved element and the trace that it leaves are not connected, moreover, it must be the case then, that the moved element will have to be deleted and the trace left *in situ* will undergo LF reconstruction for the sake of interpretation requirements (Full Interpretation). Relevant examples are to be found in the Icelandic Stylistic Fronting constructions that do not differ from constructions applying canonical word order.

2.2.2. Counter-cyclic movement at LF

The optional operation Spell-Out can take place at any point in the derivation. Only a single phrase marker is allowed in the PF component, no matter how big the structure is, we always end up with a single phrase marker, just like combining phrase markers A and B results in a phrase marker C, not A+B. The computation carries on after Spell-Out, the splitting point that leads to the interface levels: PF and LF. Interestingly enough, in the covert component the Extension Condition does not hold. We do not have any further access to the lexicon in covert syntax, hence no new items will enter the derivation, the structure is already fully-fledged. This issue, however, turned out to be problematic, suggesting that the computational system is not uniform .

There is a problematic question of LF reconstruction that is required by the Principle C of BT. Under the Copy Theory of Movement where we no longer work on traces, but rather on copies of the moved elements, we apply two operations: copying and deletion. If only one copy can be pronounced, we have to delete all the remaining copies before Spell-Out. However, later on, for the sake of satisfying Full Interpretation we have to reconstruct the deleted copies. This has a flavour of counter-cyclic movement.

Another problematic issue concerns adjunction. Adjuncts are not subject to the Extension Condition and are said to be introduced non-cyclically. This, however, poses serious problems. First we Merge all elements, only then are we allowed to move them. So, when and how do we introduce adjuncts? Merge should always extend the target an adjuncts are said not to, how is it possible then to have them at all? Chomsky says that they are introduced non-cyclically, so when exactly does this happen, in-between merging and moving? The only possible way would be to introduce them at LF, where the Extension Condition does not hold, however, they are spelled-out, hence should appear in the derivation before Spell-Out.

The computational system C_{HL} makes use of Generalized Transformations to build phrase structure. GTs (Merge and Move) are subject to the Extension Condition, they always extend the target; movement is said to apply successive-cyclically.

3. Checking Theory and movement

In the MPLT paper Chomsky proposes a Strong Lexicalist Hypothesis approach. He claims that both inflectional and derivational morphology is assigned still within lexicon and lexical items are then drawn from the lexicon bearing all assigned features and seeking to check them in syntax (we used to deal with case assignment earlier, and now we are discussing Case-checking instead). Within the Minimalist Program lexical items are drawn from the lexicon equipped with three sets of features:

- PF features (relevant information for articulatory/perceptual system);

- LF features (relevant information for conceptual/intentional system);

- formal features (FF) (intrinsic/[+interpretable] vs optional/[-interpretable] features; strong vs weak features).

Some of these features reach interface levels and must survive there for interpretation, these are intrinsic/interpretable/categorial features (e.g. substantive like N for nouns or V for verbs, gender for nouns, etc.), other will have to undergo checking and, depending on whether they are strong or weak, they will have to be checked off in overt or covert syntax, respectively. Feature-checking within the framework we are presenting here constitutes a driving force for movement, the only reason why movement should take place. Thus now, thinking about any moved category, the immediately arising question is: what feature triggers this movement undergoing checking at the same time?

3.1. Feature-checking as a driving force for movement

According to Chomsky elements move to satisfy morphological requirements. An element bearing a strong morphological feature that must be visible at PF requires checking before Spell-Out, otherwise the derivation crashes. Weak features are said to be invisible at PF, hence can undergo checking at LF, i.e. covertly after Spell-Out. Languages differ with respect to morphology, meaning, weak and strong features will have different allocation across languages. Let us take, for instance, a [+Nominative] feature of a subject NP in the following English sentence:

(41) *[$_\alpha$ A strange man] seems t$_\alpha$ to be in the garden*

The feature forces overt movement, hence must be strong. The situation is different with [+Accusative] feature of the object which, being weak, undergoes checking in covert syntax. Considering, however, constructions with expletives, we come across yet another phenomenon:

(42) a. *there is [$_\alpha$ a strange man] in the garden*

       b. *there seems to [$_\alpha$ a strange man] [that it is raining]*

In (42a) α is not in a position where it can have its Case checked. If a lexical item bearing a Case feature to check does not find itself in a Case-checking configuration ([Spec, AgrSP] for the subject and [Spec, AgrOP] for the object), it must find itself in such a configuration somewhere else, in Minimalist terms this must take place at LF. Chomsky suggests movement of α adjoining it to LF affix *there* where it can check off its [+Nominative] Case feature in a checking configuration with the matrix inflection (we face a serious problem of the Nominative feature once being strong and some other time weak, this is undesirable). In (42b), on the other hand, the phrase α has its Case feature checked within the PP and is frozen

in place. Assuming that categories move only to satisfy their own features (Greed), α not only would not be interested in movement, but also would not be allowed to move. As *there* is an affix, and affixes must be dependent on other categories for interpretation, *there* is dependent on some NP that will attach to it at LF. If the *there*-associated NP is hindered from performing that movement, freestanding *there* will be left without semantic interpretation. The derivation will converge anyway, as according to Chomsky "derivations are driven by the narrow mechanical requirement of feature checking only, not by "search for intelligibility" or the like". As all movement is considered to be purely self-serving, Chomsky proves his thesis by the following example:

(43) * *seems to [α a strange man] [that it is raining outside]*

The matrix inflection has a [+Nominative] feature to assign, nonetheless, phrase α is not allowed to move as it has its all features checked in the position it occupies at Spell-Out. What Chomsky is driving at, is proving that the condition on movement being an exclusively self-serving operation cannot be overridden even if it were to result in a convergent derivation.

4. Economy conditions

4.1. Conditions on representations: Full Interpretation - a convergence condition

Representations on the two remaining interface levels within the Minimalist Program: LF and PF, are subject to the so-called Full Interpretation condition. The idea behind it is as follows: *every symbol must receive an "external" interpretation by language-independent rules.* A symbol will not be able to receive such interpretation if it, hence its features, will not be readable (interpretable) at the interface. If a symbol can be interpreted either on LF or PF with their respective requirements (C/I and A/P systems), we assume that symbol to be a legitimate

LF/PF object. An illegitimate object on PF would be some object violating morphological requirements, e.g. an NP with its Case unchecked, or an object bearing contradictory PF features, such as [± voiced] at the same time, which is unpronounceable, incompatible with universal phonetics. Legitimate LF objects are chains, either a trivial, one-membered chain, or a nontrivial two-membered chain, hence every object that is not a chain constitutes an illegitimate LF object. A derivation *crashes* at PF if there is some illegitimate object present, the same applies to LF, however it does not have to necessarily be the case that a derivation crashing at LF must also crash at PF and vice versa. We say that a derivation *converges* at PF/LF if there are only legitimate objects present at the interface.

4.2. Conditions on derivations

Conditions on derivations are mainly concerned with necessity of the undertaken steps, that is, steps are taken only if they are necessary for convergence of the derivation. The less costly the derivational steps, the better. Convergent derivations in the Minimalist Program are compared for cost, and the most economical convergent derivation is the only possible choice. Conditions on derivations actually amount to conditions on operations Move α and Form Chain. Merge is not subject to economy conditions as it is an absolutely necessary, irreducible operation - it builds phrase markers that are indispensable to talk about syntax at all. As opposed to Merge, Move and Form Chain are optional, hence subject to economy conditions.

4.2.1. Greed

We start with the already mentioned Greed as most importantly it is the reason behind all movement. Chomsky claims that elements move only to satisfy their own features - self-serving property of Greed. Greed is a Last Resort principle, it applies only when it is absolutely necessary, when features of a given category cannot be satisfied in any other,

perhaps less costly, way. Just like in the example (34) broached earlier, this principle cannot be overridden even if its violation would result in a convergent representation.

There are constructions, however, that raise questions about the self-serving property of Greed. Let us take the so-called ECM constructions (also previously discussed in the section on Case Theory)[9]:

(44)   *Bill considers <u>John</u> to be intelligent.*

(45)   *Bill considers <u>him</u> to be intelligent.*

Our NPs *John* / *him* both are base-generated to the left of the predicative adjective *intelligent*, they both move to its AgrP where they check their agreement features. The surface order, however, shows that both NPs performed movement to the specifier position of the AgrSP of the verb *be*. As they are unable to check any of their features there (the verb of the embedded clause being non-tensed is devoid of [+Nominative] that the NPs could possibly be interested in) and, moreover, they both bear Accusative which they most probably check overtly in AgrOP of the verb *consider*, the movement of the NPs to this position seems altruistic, hence, violating Greed. Altruistic movement is banned in MPLT, nevertheless, the structures are unquestionably grammatical and the economy condition overridden.

A similar problem is posed by constructions where we have multiple wh-questions. A wh-operator bears a strong [+Q] feature that must be satisfied overtly, the force behind movement boiling down to morphology. In the case of multiple wh-questions we have more wh-operators, one of which moves to the front and the rest stays in situ. It is not at all obvious why only one of the wh-operators should move and the rest should stay in their base positions. If they all bear the same [+Q] feature, why only one is allowed to move, and what distinguishes the one that moves from all the others, and why the ones staying in situ do not

---

[9] Discussed extensively in Lasnik (1995).

cause the crash of the derivation not having their features checked overtly? Moreover, Chomsky claims that the wh-words staying in situ never , i.e. not even at LF, move to [Spec, CP], even though they are interpreted there (Lasnik 1999). If Greed were a principle behind self-serving movement then we would expect all the wh-operators to move.

Finally, in the constructions where elements seemed to have moved twice, it appears that apparently one of these movements is totally altruistic, hence violating Greed; let us consider Chomsky's example:

(46) *John seems [t' to be likely [t to win]]*

Our NP *John* moves twice, but it checks its Case feature only after moving to the matrix subject position. In the position where it leaves its intermediate trace it does not seem to check any of its features, against Greed. Apparently, it is more likely checking the EPP feature of the embedded clause satisfying the features of the target and not its own.

4.2.2. Procrastinate

We have introduced the idea of a most economical convergent derivation: such a derivation undertakes movement only if it is absolutely necessary for convergence, and even though derivations try to reach the PF component as fast as possible, it is much cheaper to perform movement in the covert component. This means that, if it is only possible, derivations will deliberately put movement off till LF and satisfy their features there. What forces a derivation to apply a more costly overt movement is the PF component, i.e. unchecked morphological features are illegitimate objects at PF interface and lead to the crash of the derivation.

4.2.3. Shortest Move vs Fewest Steps

The economy principles Shortest Move and Fewest Steps were already roughly discussed in the section on Generalized Transformations referring to Form Chain. For all the enumerated phenomena Shortest Move condition seems to play a crucial role, i.e. violations are usually due to movement that skips a possible feature checking position (the only trigger for movement). In previous frameworks this would be referred to as a violation of Rizzi's Relativized Minimality. We have already discussed the object shift in English that was considered to be an example of Shortest Move/Relativized Minimality violation. Chomsky escaped the problem by introducing the notion of Equidistance. The Fewest Steps principle, on the other hand, wants a derivation to converge with fewest possible operations performed, hence it will choose between convergent derivations on the basis of the number of operations applied on their way to convergence, and thus a derivation converging with three operations is more economical than the one needing four such operations.

The remaining problem was how to combine the two economy principles, as they seem to be against each other and Chomsky rescued the discussion by introducing Form Chain as an option (discussed earlier).

5. Conclusion

A Minimalist Program for Linguistic Theory where Chomsky presents his new framework is, as he puts it himself, only a sketch that still leaves a lot of material for further research and investigation. Some of the attractive ideas like eliminating D-Structure, the Extension Condition with respect to adjuncts, the nature of features or economy principles, are at times rather vague.

The argument behind eliminating the D-Structure level wasa based on the so-called virtual conceptual necessity, namely, D-Structure is said to be conceptually irrelevant

for both the A/P (articulatory/perceptual) and the C/I (conceptual/intentional) systems. Nevertheless, we do not find any satisfactory answer to the question of why exactly we are forced to introduce the elements entering the VP-Shell first into the derivation and theta role assignment still takes place very early in the derivation within the VP. As noted, these facts have a strong taste of D-Structure and it seems that we have disposed of the name, but not all of the phenomena applying there, nor the timing as it appears the same.

The idea behind the Extension Condition sounds attractive, but the pending question is: why is it not applicable to adjuncts? Adjuncts are said not to extend the target, nonetheless, it sounds logical that all lexical items introduced into the derivation obey that rule. The framework, so far, leaves us without answers.

Chomsky claims that the whole force behind movement lies in satisfaction of morphological features, however, he does not seem to specify the nature of these features in the present approach[10]. Can one feature be checked twice?

Eventually, many problems posed against the economy principle Greed sound alarming, i.e. Greed in its nature and formulated as it is still leaves a lot to be desired for it does not appear at all that elements move only to satisfy their own features.

The framework presented here under the name *A minimalist Program for Linguistic Theory* seems undoubtedly an attractive one and goes to explain a number of phenomena that occurred problematic for the previous frameworks. However, as a new approach it already faces serious problems that were not difficult to account for in the GB and P&P frameworks. The program is full of unanswered questions and under closer inspection some aspects are not so distant from the previous phases of generativism.

---

[10] These matters are devoted lots of time and space in his future articles.

REFERENCES

Abraham, et al. 1996. *Minimal Ideas*. Amsterdam/Philadelphia: John Benjamins Publishing

    Company.

Belletti, A. 1990. *Generalized Verb Movement: Aspects of Verb in Syntax.* Turin: Rosenberg

    and Sellier.

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

Chomsky, N. 1993. "A Minimalist Program for Linguistic Theory". In: Hale and Keyser *The*

    view from Building 20. Cambridge, Mass.: MIT Press.

Chomsky, N. 1995. *The Minimalist Program.* Cambridge, Mass.: MIT Press.

Hornstein, N. 1994. *Logical Form: From GB to Minimalism.* Oxford and Cambridge, MA:

    Blackwell.

Lasnik, H. 1999. *Minimalist Ananlysis.* Oxford: Blackwell.

Ouhalla, J. 1999. *Introducing Transformational Grammar: From Principles and Parameters*

    to Minimalism. London/Sydney/Auckland: Arnold.

Pollock, J-Y. 1989. "Verb Movement, Universal Grammar and the structure of IP. *Linguistic*

    Inquiry 20, 365-424.

Rizzi, L. 1990. *Relativized Minimality*. Cambridge, Mass.: MIT Press.

Webelhuth, G. (ed.). 1995. *Government and Binding Theory and the Minimalist Program.*

    Oxford and Cambridge, MA: Blackwell.