On Minimalist Requirements in Syntax*

Hong Bae Lee
(Sogang University)

Lee, Hong Bae. 2003. **On Minimalist Requirements in Syntax.** *Korean Journal of English Language and Linguistics* 3-2, 255-280. The present paper will argue what can be considered to be principled elements of the initial state S_0 of the Faculty of Language, which are called the Interface Condition (IC), and how far we can take the strongest minimalist thesis (SMT), which aims to offer principled explanation of language in terms of IC and the principle of efficient computation, to linguistic analysis. We will discuss implications of label-free phrase structures, required by the strong version of the Inclusiveness Condition, and possibilities of crash-free syntax, required by the condition of efficient computation. I will point out problems of Chomsky's assumption that an externally Merged expletive *there* is a head, which, as a probe, undergoes agreement with the goal T. I will present several advantages we obtain if we maintain A and A' distinction, and assume that *wh*-movement to the outer [SPEC, *v*] is an A'-movement like *wh*-movement to [SPEC, C].

Key Words: label-free phrase structure, minimalist requirement, Inclusiveness Condition, crash-free syntax

1. Introduction

Since the term "minimalism", I understand, was first introduced in Chomsky (1993), the spirit of minimalism for linguistic analysis has not undergone any change, but it seems that working hypotheses for implementing minimalism in

---

*I would like to express my gratitude to the three anonymous reviewers, who have greatly helped to improve the contend of the paper as well as its typological and grammatical errors. Of course, all remaining errors are mine.

This work was supported by the Brain Korea 21 Project in 2003 and a research grant of Sogang University.

syntactic analysis has undergone several dramatic changes. Consider some of the concepts that were considered to be minimalist hypotheses, conditions and principles, but no longer believed to be the ones: the principle of Procrastinate, the notion of feature strength, checking and checking relations, Agr-node, Move-F, etc. If we mean by minimalism "as few hypotheses as possible and as simple operations as possible", every theory of language can be said to be a kind of minimalist approach to language analysis. We know, however, that "minimalism" in linguistics does not simply mean "as few and simple as possible"; it has a deeper meaning accumulated through a long history of generative tradition.

Since the three "levels of adequacy" were first presented in Chomsky (1964), the goal of generative grammar has been how we write a grammar that meets the highest level of adequacy (i.e., explanatory adequacy). As Chomsky (1995, 2000) says, we have always tried to solve the tension between descriptive adequacy for L and explanatory adequacy for UG. It seems that sometimes we need rich descriptive mechanism for the description of particular languages, but the problem of language acquisition requires a theory of language be as simple as possible. In that sense, minimalism is a natural consequence of a theory of language that aims to reach explanatory adequacy. To put it another way, if a child acquires a language using only the linguistic data that it is exposed to, logic goes like this: a theory of language must employ only properties of human language: that is, "conceptually necessary" properties of human language. Then, a question naturally arises as to what are exactly conceptually necessary properties of language.

A plausible answer to the question above will be discussed in section 2. The remainder of the paper is organized as follows. In section 3, we will deal with the Inclusiveness Condition together with Collins's (2002) label-free phrase structure theory. In section 4, the possibility will be discussed whether a crash-free syntax in which every derivation is "convergent" can be formulated for the economy of optimal computation. In the section to follow, we will examine the kinds of structural relations that are allowed in minimalist syntax, and Chomsky's (2001a and b) argument in which the externally-merged expletive *there* should be analyzed as a probe, taking the lower finite T as its goal. Section 5 deals with interactions between uninterpretable features, Agree and Move, together with some examples showing that they do not always occur together. I propose that we revive the distinction between A- and A'-movement as we did in the GB-theory. Section 7 concludes the paper.

2. FL and Initial State S^0

Chomsky's biolinguistic approach to language L takes the object of inquiry to be a subcomponent of the brain that is specifically dedicated to L, called "the faculty of language" (FL). FL has an intial state S_0 embedded in human gene as a product of evolution. The initial state determines possible human languages; a particular language, which is one of attainable states of FL, is a further specification of S_0 by fixing values of parameters, which are given without values in S_0. We know that we have to be exposed to a certain language to acquire that language. This exposure to linguistic environment is assumed to play a major role in determining values of parameters. In that sense, the acquisition of a particular language crucially depends on analyzing primary linguistic data (PLD) of the language. Chomsky (2001b), however, claims that the initial conditions for language acquisition include more than PLD and S_0. Since humans are also an organic system, they have properties that are generally shared by all organic systems. Therefore, it is argued that language acquisition is also affected by general properties of organic systems. Then, the question arises as to exactly what aspects of the general properties are related to human language acquisition.

We can infer some general properties of organic systems from Chomsky's (2001b) discussion on this subject. First, from the statement that ". . . natural selection [in evolution] can only function within a "channel" of options afforded by natural law", we can infer that languages cannot be different without bounds but within the limits of parametric variation afforded by the initial state S_0 of FL. Second, from the statement that ". . . the brain is as far forward as possible on the body axis" as minimization of "wire length". . . in microchip design, we can infer that language obeys economy conditions, including principles of efficient computation. Third, from the statement that "nature is perfect" in the sense that nature has "a sense for beauty", language contains no imperfections. With these properties in mind, then we can seek an explanation of L going beyond explanatory adequacy, "asking not only *what* the properties of language are, but *why* they are that way" (Chomsky 2001b: 2), whereas our usual research focusing on the problems of descriptive and explanatory adequacy is restricted to the factors related to PLD and S_0. The transcendence of these limits, Chomsky (2001b: 2) argues, is made possible by the crystallization of the Principles-and-Parameters theory. And with this theory we are able to find a way to overcome the tension between descriptive and explanatory adequacy for the first time.

Let's consider reconstruction problems as an example of the tension,

(1) *Which claim that John$_i$ is a traitor did he$_i$ believe *t*?

To account for the Condition C violation of (1), the GB-theory has employed the reconstruction operation, which undoes the effects of (raising) movement. Reconstruction is a "lowering" operation, while all other movement operations are a "raising" operation. Therefore, we have a tension between descriptive adequacy for accounting for the

Condition C violation in English sentences like (1) and explanatory adequacy in which the lowering operation has no place. Chomsky (1993), however, adopts a copy theory of movement. According to the copy theory of movement, the output of movement is not (1) but (2), which undergoes a later transformation to form one of the two structures in (3) for interpretation.

(2) which claim that $John_i$ is a traitor did he_i believe which claim that $John_i$ is a traitor

(3) a. [Which claim that $John_i$ is a traitor]$_x$ did he_i believe x
b. $Which_x$ did he_i believe x [claim that $John_i$ is a traitor]

The preference principle (Chomsky 1993: 209), which requires the restriction to be minimal in the operator position, prefers (3b) to (3a). Note that there is no lowering in (2), and the Condition C violation is a natural consequence that follows from the structure in (3b). Then, we may say that the developments of generative grammar since its inception in the 1950s have been a history of struggle to solve the tension between descriptive and explanatory adequacy.

Chomsky (2001b) divides properties of the initial state S_0 of FL into two categories: principled elements and unexplained ones. He attributes the latter to something independent of FL: genealogical path-dependent evolutionary processes or some properties of the brain that remain unknown. Here, we may raise a question: what are the principled elements of S_0? Chomsky (2001b) claims that "The principled elements of S_0 are the conditions imposed on FL by the systems with which it interacts." In order for L to interact with sensorimotor (SM) systems for perception and articulation and thought systems (TS) for concept and intention, it must be designed to meet the interface conditions[1)] imposed by these external systems: the information contained in the expressions generated by L must be legible to the performance systems. Thus, the aim of the strongest minimalist thesis (SMT) is to offer principled explanation of L in terms of the principled elements of S_0 and general properties of organic systems, going beyond explanatory adequacy.

Then, specifically what are the principled elements of S_0? In other words, what can be regarded as much-discussed interface conditions (ICs)? Here are some properties that I believe to be candidates of IC:

(4) a. two interpretive components: the phonological component Φ and the semantic component Σ
b. elements specified in LEX; the Inclusiveness Condition
c. empirically-proved/conceptually-required operations such as Merge, Agree
d. the structural relations created by Merge: dominate, c-command, term-of
e. duality of semantic properties of L: theta-role-related semantic properties, and scopal and discourse-related semantic properties (such as focus, topic, new/old information, specificity, etc.)
f. cyclic derivations: derivation by phase
g. the principle of full interpretation
h. whatever conceptually required properties of L

In the following sections, we will discuss some of the properties listed in (4), pointing out problems in connection with the interpretation of these properties.

3. Inclusiveness Condition

[1]These conditions are sometimes called "bare output conditions" or "legibility conditions".

Language L consists of one generative component and two interpretive components. The unique generative component, the narrow syntax (NS), generates a set of derivations, mapping a lexical array LA to a derivation D_{NS}, which I call a "derivational unit". The phonological interpretive component Φ maps D_{NS} to PHON, which will be accessed by sensorimotor systems, and the semantic interpretive component Σ maps D_{NS} to SEM, which will be accessed by conceptual-intentional systems. It is assumed that Σ is uniform throughout all L's, and NS is as well, if parameters are restricted to the lexicon LEX. Φ, however, is highly variable among L's. The Minimalist Program assumes that all mappings satisfy the Inclusiveness Condition, which requires that no operations add any new elements to a derivation but rearrange those in LEX; introducing a bar or an index to a category is not allowed.

Consider the operation Merge:

(5) Merge
 Take two objects α and β to form a new object γ.

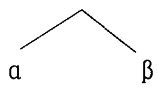
The label of the new object γ, LB(γ), is either LB(α) or LB(β). Therefore, Merge actually produces either of the following structures:

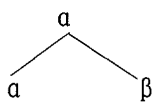
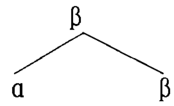
(6) a. {LB(α), {α, β}}
 b. {LB(β), {α, β}}

Collins (2002) argues that Merge in (5) violates the Inclusiveness Condition, if it is assumed to generate either structure in (6). To put it another way, Merge as operated above is actually "marking" one of the two merged objects as the label of the new object as in (7), which Collins (2002) argues violates the Inclusiveness Condition, as we have done in the GB-theory by assigning indices to syntactic objects.

(7) a. $\{\underline{\alpha}, \beta\}$
b. $\{\alpha, \underline{\beta}\}$

He argues that Merge should produce a structure like (8) without label but neither like (9).

(8) $\{\alpha, \beta\}$ 

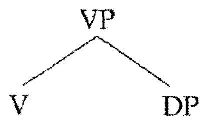
(9) a.  b. 

Chomsky (2001b) argues that there are three ways of identifying labels of phrases: first, the worst case is that we identify labels by explicit rules, as we have practiced even in the GB-theory as in (10):

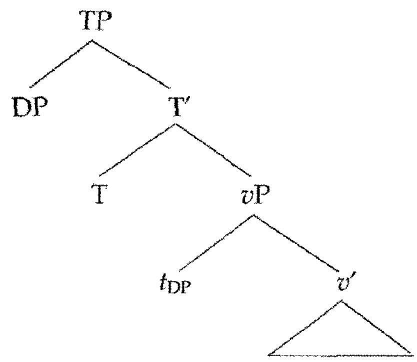
(10) CP → C IP

The phrase structure like (10) says that the label of the phrase that consists of C and IP is CP. A preferable method to identify labels is to predict them by a general rule: the selector/target projects (see Chomsky 1995). For example, the label of the structure in (11a) is identified as VP because V selects/subcategorizes for DP, and the label of the highest node is identified as TP in (11b) because the subject DP has moved taking T′ as its target. A more attractive way is that we need no labels. Therefore, the phrase structure for "the boy kissed the girl" may look like (12).

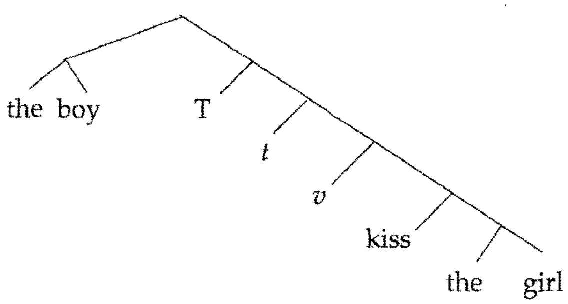
(11) a.



b.



(12)



Then, how do we identify labels of nonterminal nodes in (12)? Collins (2002) claims that we may identify the label of a structure by using grammatical features and operations in (13) that are required in other parts of grammar:

(13) a. Subcategorize for (α, β): if α subcategorizes for β, then α projects.

b. Theta-mark (α, β): if α theta-marks β, then α projects.

c. Have an EPP-feature (α, β): If α has an EPP-feature that can be satisfied by β, then α projects.

d. Agree (α, β): if α as probe Agrees with β, then α projects.

'The boy' and 'the girl' are identified with DP, the node immediately above *v* with *v'*, and the node immediately above T with T', because D subcategorizes for NP, *v* for VP, and T for *v*P, respectively. The nodes immediately above T and *v* are also identified with T' and *v'*, because T and *v*, respectively, Agree with the subject DP and the object DP. 'Kiss the girl' is identified with VP, because V theta-marks DP. The highest node of the structure is identified with TP, because T has an EPP-feature that is satisfied by DP.[2]

Collins's (2002) label-free phrase structure does not simply claim that we do not have to mark labels of branching nodes, as we can see in (12), because we are able to predict them with the help of other features in the grammar. It claims that the labels of phrasal categories such as NP and VP are not needed at least in NS. Therefore, all operations and conditions should be reformulated that make reference to nonterminal categories such as X-bar and XP. To take an example, the Chain Uniformity Condition, which is proposed to block movement of a nonterminal category to a terminal category position and vice versa, cannot be maintained.

(14) Chain Uniformity Condition (Chomsky 1995: 253)
A chain is uniform with regard to phrase structure status.

I will not, however, discuss this subject further, since it is not the topic of the present paper (see Collins (2002)).

4. Converge, Crash, and Optimal Computation

[2]In fact, the discussion above is not correct, because we make reference to phrasal categories like DP, V' and VP. See the discussion directly below.

Consider one of the important conditions in the Minimalist Program: the Phase Impenetrability Condition (Chomsky 2000: 108).

(15) Phase impenetrability condition (PIC)
The domain of H is not accessible to operations, but only the edge of H.

Given a phase HP = [α [H [β]]], β is the domain of H, and α and H are its edges.[3] The PIC presupposes that all syntactic derivations are cyclic in the sense that all derivations proceed phase-by-phase. In order for an element to move out of β, the domain of H, it must move to an edge position first of all, because an operation applying to HP cannot access to an element within β. In that sense, the edge plays a role of escape hatch; an operation applying to C can see only the edge of *v*, so that an element within the domain of *v* can move to [SPEC, C] only through the escape hatch at the edge of *v*. The derivation of sentence (15) shows that this is the case:

(16) What did John buy?

There is no way of moving object (OB) *what* directly to its surface position [SPEC, C] in the grammar that assumes the PIC. We will further discuss this subject in section 6.

Chomsky (2002) extends "the derivation by phase" of NS to Φ and Σ, suggesting that there is a single cycle: Φ and Σ proceed in parallel with the NS derivation. Therefore, there is an operation "transfer" incorporating Spell-Out, which was first introduced in Chomsky (1993) as an operation that maps the output of overt syntax to Φ.

[3]We take phases to be CP, *v*P and (probably) DP, but not TP.

(17) Transfer

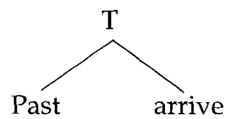
Hand the phase β over to Φ and Σ.

When the computation reaches the phase HP, the operation Transfer sends over the phase β to Φ and Σ to have it accessed by SM and TS, respectively. Then, there is no LF in a strict sense, because the computation maps LA to <PHON, SEM> phase by phase cyclically.

As we have discussed above, language L generates a set of derivations, each derivation consisting of a set of derivational units, D_{NS}'s. The operation Transfer each delivers D_{NS} to Φ and Σ to map it to the pair <PHON, SEM>, which will be accessed by SM and TS, respectively. We say that D_{NS} *converges* if the pair <PHON, SEM> each satisfy IC, that is, they are defined in terms of principled elements of S_0; otherwise, it *crashes*. We may expect that the condition for optimal computation requires that every derivation converge: no derivation crash. This condition may be too strong, but if it can hold, we may be able to eliminate a lot of unnecessary and redundant steps in derivations: comparison of derivations to decide which derivation is simpler, backtracking and look-ahead to determine whether it is legitimate to apply an operation at a certain point, and "non-local" operations, which have in general no place in a grammar in which every derivation must converge.

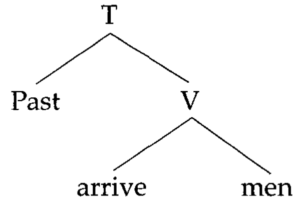
Then, we may ask whether it is possible to formulate a "crash-free" syntax satisfying the strong version of IC that every derivation is convergent. We can think of two extremely different views on this matter: "free-generation" syntax and "crash-free" syntax. The former allows the grammar to freely generate structures including ill-formed ones; then, a set of constraints, conventionally called filters, discard ill-formed ones and pass through only well-formed ones, as we have generally practiced in some degree in generative grammar. The latter, however, constrains every step of grammatical operations, so that end-products of each derivation are all well-formed. There would be no comparison of derivations to select only convergent derivations. If it is tenable, as Frampton and Gutmann (2002) claims, the "crash-free" syntax is "computationally" much cheaper than the "free-generation" syntax. Chomsky (2001b), however, argues that the "crash-free" syntax which requires that every derivation be a convergent one and no derivations ever crash may be too strong; we may weaken the requirement: every derivation can be extended to a convergent one.

To understand how the strong version of Frampton and Gutmann's (2002) "crash-free" syntax differs from a weaker version of Chomsky's (2001b) "failure-proof" syntax, consider the derivation of *men arrived,* formed from LA of three heads *men, arrive* and Past. In Chomsky's (2000, 2001a and b) system, two objects are allowed to merge as long as one of them selects the other. Therefore, a derivation is possible in which Past selects *arrive* before *arrive* selects *men.* But the derivation is doomed to crash, because, according to the principle of strict cyclicity, *arrive* will never have a chance to select its argument *men.* In other words, the Past tense that selects the verb *arrive* will create the structure in (18).

(18)


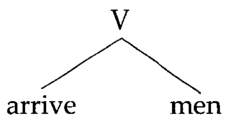
Once we build a structure like (18), we cannot countercyclically turn back to make *arrive* select its argument *men.* However, we have a different way to reach a convergent derivation with the LA: the verb *arrive* merges with *men,* the result of which is selected by Past, creating the well-formed structure in (19).

(19)



Frampton and Gutmann (2002) argues that we can prevent this kind of nonconvergent derivation in which Past selects *arrive* before *arrive* selects *men*. Suppose we assume that every derivation starts only with a head that does not select anything, in this case *men*. *Men* cannot merge with the head Past, because the tense only selects a verb. Therefore, the derivation must begin with *arrive* merging with *men*, creating (20).

(20)



If it is tenable, as Frampton and Gutmann (2002) argues, we can maintain the strong version of "crash-free" syntax, further approximating to the requirement of optimal computation.

5. Structural Relations

A grammar that meets the SMT employs only structural relations that the operation Merge creates:

(21) a. membership relation
b. dominance relation (= contain)
c. the relation of "term of"
d. a derived relation c-command (= sister of contain)

Merge, which takes two objects α and β, creates a new object {α, β} and a membership relation, which defines that α and β are members of the new object. The iterative application of Merge yields the relations "dominate" (i. e., contain) and "term-of".[4] We can define a derived relation "c-command" from the relations created by iterative application of Merge. It is assumed that any relation other than (21) must be empirically motivated, and ". . . is prima facie departure from the SMT" (Chomsky 2001b: 5).

If it is true, then what happens to the most cherished Head-SPEC relation[5] since the GB-theory in which the tensed I m-commands, governs, and assigns the nominative Case to its specifier? Even in Chomsky (1993) the covert Head-SPEC relation is created to check the accusative Case of direct object. But this relation is not allowed in the SMT because the structural relation that requires "m-command" cannot be a member of the structural relations listed in (21). Then, we may still save the SPEC-Head relation, because a SPEC certainly c-commands its head. Chomsky (2001b: 6) argues, "There is a relation R (SPEC, H), namely c-command, but no relation R (LB, H) where LB is the label of SPEC, . . . (unless LB = SPEC)." To put

---

[4]This is the definition of terms given in Chomsky (1995):

a. K is a term of K (that is, the entire set or tree is a term).
b. If L is a term of K, then members of the members of L are terms of K.

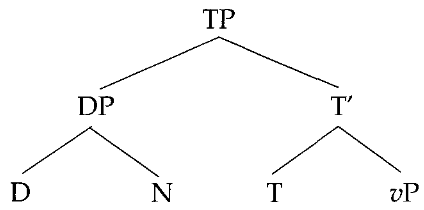
As an example, consider the VP 'likes him,' which has the structure K = $\{V_{likes}, \{V_{likes}, D_{him}\}\}$

K (= VP) itself is a *term*.
K has two members:
member 1: the label V_{likes}
member 2: a two-member set $\{V_{likes}, D_{him}\}$, each being a member of member 2.
Thus, V_{likes} and D_{him} are *terms*.

[5]We used to call this relation SPEC-Head relation; it is, in fact, a Head-SPEC relation, because the relation is established under the assumption that the head m-commands, governs, and further affects its SPEC.

another way, the only SPEC-Head relation that is allowed in the SMT is when the SPEC itself is a head. Then, consider a typical SPEC-Head relation:

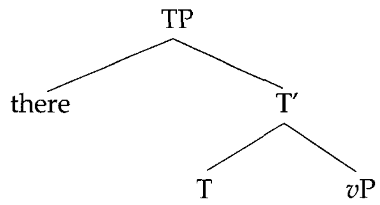
(22)



T is not in the minimal search space of the head D of DP (i. e., is not c-commanded by the head D); so no syntactically significant relation can be established between D and T.

However, consider the following structure in which expletive *there* directly merged to [SPEC, T].[6]

(23)



Chomsky (2000, 2001a and b) claims that the c-command relation can be established between *there* and T in a structure like (23), if we take the expletive to be a head acting as the probe for Agree which deletes the uninterpretable [person] feature of *there*, taking T as the goal.[7] We can easily notice

---

[6] This is the only case in which external Merge suffices to check an EPP-feature.

[7] If I correctly understand, Chomsky (2000, 2001a and b) has never treated as a head expletive *there* that is to move later in the derivation as in the following structure:

T-seems [there to be a man in the room]]

If *there* acts as a head in (23), I cannot find reason why it should that Agree in this case is an operation different from the normal Agree, which we take to hold between α and β: ". . . an uninterpretable feature of α must be in an appropriate relation[8] to interpretable features of some β" (Chomsky 2001b: 11). Now, consider the typical case of Agree between T and a nominal. T contains uninterpretable ϕ-features, which activate T as a probe, and nouns inherently contain complete interpretable ϕ-features, which assign values to (and delete) the uninterpretable ϕ -features of the probe: in this case, those of T.

Let us consider the Agree relation between *there* (the probe) and T (the goal). It seems that the expletive with an uninterpretable [person] feature meets the qualifications for a probe. But, then, is T qualified as a goal? The definition of Agree given above indicates that T cannot be a goal, because it does not contain any "interpretable" ϕ-features. Furthermore, uninterpretable features are given "unvalued"; Agree assigns values to (and deletes) them. To make T a goal in structures like (23), we have to make at least two additional assumptions: (a) a category with an uninterpretable ϕ-feature set can be a goal; (b) since the function of the operation Agree is to give values to the uninterpretable features of the probe (and delete them) based on the values of interpretable features of the goal, we have to extend the definition of Agree, so that it allows an element with "valued"[9] uninterpretable ϕ-features to act as a goal. I do not know whether these additional assumptions are within the SMT.

One alternative approach to this subject is to assume that Agree applies not between the expletive *there* and T, but between the expletive and the nominal that T agrees with in

---

not be a head above. I will not discuss the consequences of analyzing all occurrences of the expletive as a head.

[8]An appropriate relation here means the c-command relation.

[9]We assume here that "unvalued" ϕ-features of T are valued by Agree that applied between T and some nominal.

(23). An immediate problem in this case is that two probes (EXPL and T) Agree with a goal (nominal), unlike an ordinary multiple agreement in which a probe agrees with multiple goal s.[10] I leave this subject open for further research.

6. Uninterpretable Features, Agree and Move

The uninterpretable features that are most discussed fall into three types:

(24) a. Φ-features on the probe (T, *v*)
b. structural Case on the goal (N or D)
c. EPP-feature on the probe

Uninterpretable features are distinguished in the lexicon from interpretable ones in that they are given without values: e. g., [uPerson, uNumber, . . .]. The Principle of Full Interpretation (FI) requires that they be valued and deleted under Agree for the D_{NS} to converge.[11] An important question we can ask at this point is why uninterpretable features and Agree exist in L. As we have just said, if uninterpretable features are to be deleted by the operation Agree for a derivation to converge, why are they given in LEX in the first place at all?

Chomsky (2000, 2001a and b) suggests that we find the answer to the question in properties of internal Merge (= Move). In other words, internal Merge requires just these uninterpretable features for it to operate:

(25) a. the target for Move is determined by (24a)

---

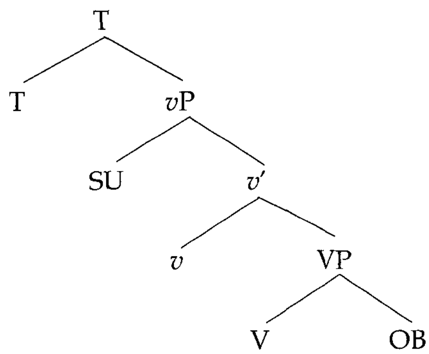
[10]See Chomsky (2001a) for multiple Agree. For a different approach to Agree in general, see Framption and Gutmann (2000).

[11]Probably some uninterpretable features must be transferred to Φ before they are deleted, since they may have phonetic reflexes.

b. the element to be moved is determined by (24b)
c. the availability of the SPEC position of the probe with an EPP-feature for Move is determined by (24c)

Let us consider the typical clausal structure in (26) to show how uninterpretable features bring about the application of internal Merge. Suppose both SU(bject) and OB(ject) are DPs whose nominal elements contain a "complete" interpretable ϕ -feature set and uninterpretable structural Case.

(26)

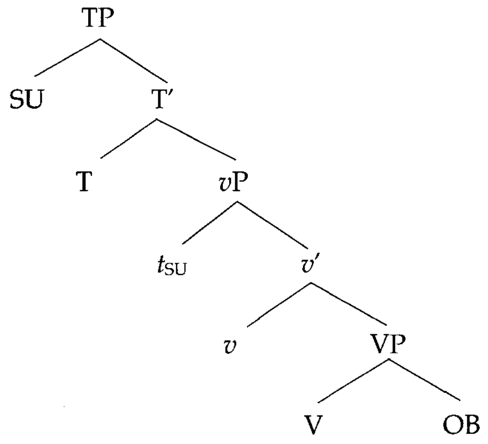


T and *v*, however, contain a "complete" uninterpretable ϕ -feature set. Agree is applicable between α and β, when both of them contain uninterpretable features and at least one of them is "complete."[12] In this case, Agree takes place between [T, SU] and [*v*, OB], but internal Merge applies only in the latter, moving SU to [SPEC, T] and yielding (27). But see directly below to find out the case in which internal Merge applies between [*v*, OB].

---

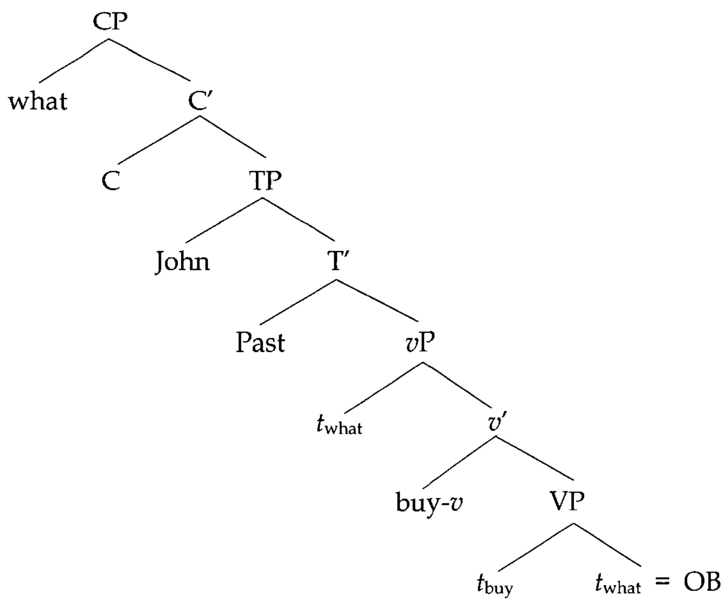
[12] Let us call it "the ϕ-complete Hypothesis": α must have a complete set of ϕ-features to delete uninterpretable features of the paired matching element β (Chomsky 2001a (3b)).

(27)



To understand properties of internal Merge, let us consider here one of the most intriguing movements in English: *wh*-movement as in (28).

(28)



Internal Merge, as defined in (25), is a composite operation consisting of Agree + Identify + Pied-piping, which suggests that Agree is a prerequisite for internal Merge. Then, the immediate question we can raise here is whether the same Agree operation applying between *v* and non-*wh*-nominal applies between *v* and *wh*-nominal *what*. If they are the same, then why is the EPP-feature of *v* sensitive only to *wh*-elements but not ordinary nominals? Furthermore, it is never the case that a *wh*-element is "spelled out" at the SPEC position of *v*P in English; in English at least, the SPEC position of *v*P functions only as an intermediate station of a *wh*-element moving to its final destination.[13] Then, we may conclude that what causes the *wh*-element movement to [SPEC, C] causes the *wh*-element movement to [SPEC, *v*].

Suppose the uninterpretable [wh]-feature of C causes movement of a *wh*-phrase to its SPEC position (from the [SPEC, *v*] position). The uninterpretable [wh]-feature here may be taken to be an EPP-feature of C. In Chomsky (2001b), an EPP-feature is given as an OCC(urrence)-feature which means "an occurrence of some β" (Chomsky 2001b: 10).[14] We may assume that the *wh*-movement to [SPEC, *v*] is brought about by the same feature. If it is tenable, the set of features participating in Agree between *v* and an ordinary nominal are different from those in Agree between *v* and a *wh*-phrase.[15] It seems that the derivation of sentences like (29) clearly shows that two different operations apply to the head *v*:

(29) Who does everyone believe John to like?

---

[13]The [SPEC, C] position also functions as an escape hatch of a *wh*-element as in the following sentence:

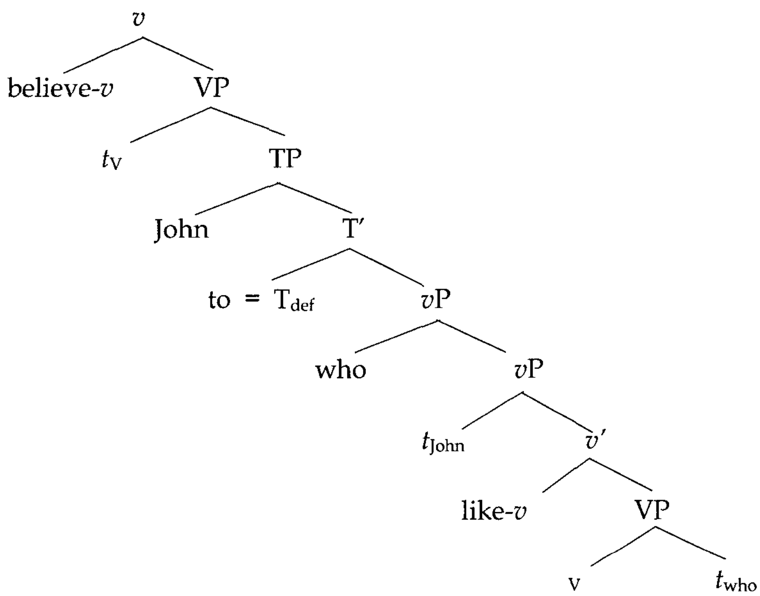
Who do you think [$_{CP}$ *t'* that [Mary loves *t*]]?

[14]An OCC-feature may be taken to be a kind of c-selectional feature that specifies that a certain category may occur in the SPEC position of a head with the OCC-feature.

[15]At present I have no idea about the set of features that take part in Agree between *v* and a *wh*-phrase.

Who inserted as direct object of the verb *like* first moves to the SPEC position of the embedded *v*P, producing structure (30):

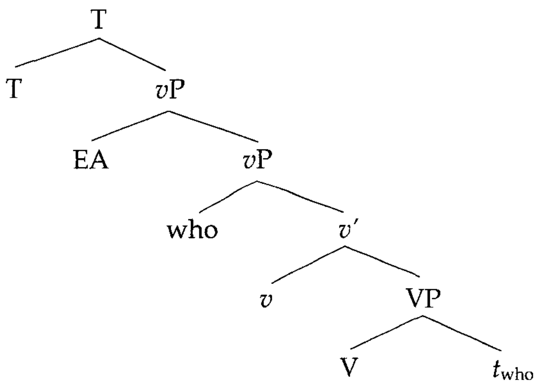
(30)



In order for *who* to move to its surface position, it first has to move to the outer SPEC of the verbal complex *believe-v* in (30). There seem to be two different applications of Agree taking place in (30): between *(believe-)v* and the external argument of the embedded clause *John* checking and deleting ϕ -features of the former and accusative Case of the latter, and between *(believe-)v* and *who* to initiate *wh*-movement to the outer SPEC position of *believe-v*. This derivation shows not only that the head *(believe-)v* Agrees with two different elements but that it violates the Minimal Link Condition (MLC): the head *(believe-)v* selects *who* rather than closer *John* as an element to satisfy its EPP-feature. I suggest here that we revive A and A′ distinction, which Chomsky (2000, 2001a and b) explicitly discarded, and Rizzi's (1990) relativized minimality. I further claim that movements to [SPEC, C] and [SPEC, *v*][16) are A′-movement, whereas movement to [SPEC, T] is a typical

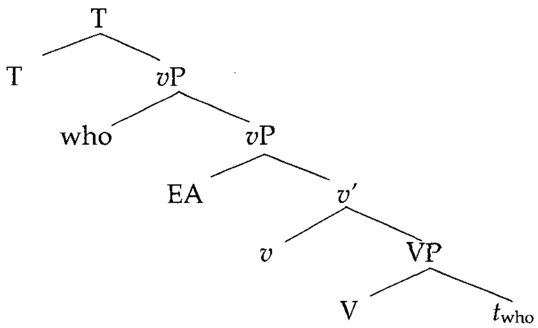
A-movement. There are a number of advantages in doing so. First of all, if we analyze movement of *who* in (30) as an A′-movement, it does not violate the MLC because it moves over argument *John*. Another advantage is that there is no need to "tuck in" a *wh*-object under the external argument (EA) as in (31) to meet the MLC for the movement of EA to [SPEC, T], following Richards (1997), because the outer SPEC position of *v*P will be an A′-position.

(31)



Instead, we can maintain one of the fundamental assumptions of the Minimalist Program that internal Merge always merges an element with the root,[17)] which generates (32) rather than (31).

(32)



[16]The so-called Object Shift, another movement to [SPEC, *v*], may be an A-movement.

[17]This is sometimes called the Extension Condition.

Furthermore, we do not have to assume the principle of simultaneous application within a phase. Consider (33) which shows the typical A′-movement.

(33) what C [he T [t'_{wh} [t_{he} see t_{wh}]]]

The derivation of (33) seems to show two MLC violations: first, raising of *he* to [SPEC, T] over *what* in the outer SPEC of *v*, and raising of *what* to [SPEC, C] over *he* in [SPEC, T]. To avoid the MLC violation, Chomsky (2001b) assumes that all operations apply simultaneously within a phase.[18] The principle of simultaneous application, at least in this case, seems to me that we are forced to pretend not to see anything although there is actually something there. If we assume, however, that the outer SPEC position of *v* occupied by a *wh*-phrase is an A′-position, adopting the A and A′ distinction and Relativized Minimality, we can dispense with the principle of simultaneous application.

7. Conclusions

We have briefly reviewed some of the principled elements of the initial state S_0 of FL, much discussed in the Minimalist Program (see (4)), and the consequences of their application to the analyses of some English constructions. In particular, we have considered Collins' (2002) suggestion that the strict interpretation of the Inclusiveness Condition compels us to eliminate phrasal labels from structural representations and to reformulate operations and conditions like the Chain Uniformity

[18]I will not discuss the problems related to the simultaneous application principle any more here. Those who are interested in this subject see Epstein and Seely (2002).

Condition in (14) that make reference to nonterminal categories such as X-bar and XP. We have also discussed Chomsky's (2000, 2001a and b) suggestion that the expletive *there* which is externally Merged as a specifier of finite T be a head acting as a probe that Agrees with the finite T; we have concluded that the operation Agree applying to a pair (*there*, T) in (23) is not a "typical" Agree operation, which means that we need some additional assumptions for Agree to apply to structures like (23). I have also suggested that we revive the A and A′ distinction. And I have shown that with the distinction between A- and A′-notion the Relativized Minimality could explain a number of problems related to *wh*-movement in English.

References

Brody, M. 1995. *Lexico-Logical Form: A Radically Minimalist Theory.* Cambridge, MA: MIT Press.

Chomsky, N 1964. *Current Issues in Linguistic Theory.* Mouton, The Hague.

Chomsky, N. 1965. *Aspects of the Theory of Syntax.* Cambridge, MA: MIT Press.

Chomsky, N. 1993. A minimalist program for linguistic analysis. In K. Hale and S. J. Keyser, eds., *The View from Building 20: Essays in Linguistics in Honor of Sylvain Bromberger.* Cambridge, MA: MIT Press.

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

Chomsky, N. 2000. Minimalist inquiries. In Martin et al., eds.; *Step by Step: Essays in Minimalist Syntax in Honor of Howard Lasnik.* Cambridge MA: MIT Press.

Chomsky, N. 2001a. Derivation by phase. In Kenstowicz, ed., *Ken Hale: A life in Language.* Cambridge MA: MIT Press.

Chomsky, N. 2001b. Beyond explanatory adequacy. Ms. MIT.

Collins, C. 2002. Eliminating labels. In S. D. Epstein and T. D. Seely, eds., *Derivation and Explanation in the Minimalist Program.* Malden, MA: Blackwell.

Epstein, S. 1999. Un-principled syntax: the derivation of syntactic relations. In Epstein and Hornstein, eds., *Working Minimalism.* Cambridge, Mass: MIT Press.

Epstein, S. D. and T. D. Seely. 2002. Rule applications as cycles in a level-free syntax. In S. D. Epstein and T. D. Seely, eds., *Derivation and Explanation in the Minimalist Program.* Malden, Mass: Blackwell.

Frampton, J. and S. Gutmann. 2002. Crash-proof syntax. In S. D. Epstein and T. D. Seely, eds., *Derivation and Explanation in the Minimalist Program.* Malden, MA: Blackwell.

Hiraiwa, K. 2000. Multiple agree and the defective intervention constraint in Japanese. Ms. MIT.

Holmberg, A. 1999. Remarks on Holmberg's generalization. *Studia Linguistica* 53, 1-39.

Jenkins, L. 2000. *Biolinguistics.* Cambridge, UK: Cambridge University Press.

Kenstowicz, M., ed. 2001. *Ken Hale: A Life in Language.* Cambridge MA: MIT Press.

Lasnik, H. 1999. *Minimalist Analysis.* Malden MA, Oxford: Blackwell.

Richards, N. 1997. *What Moves Where When in Which Language?* Doctoral Dissertation, MIT. Distributed by MIT Working Papers in Linguistics.

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

Uriagereka, J. 1999. Multiple spell-out. In S. D. Epstein and N. Hornstein, eds., *Working Minimalism.* Cambridge, Mass: MIT Press.

Hong Bae Lee
Dept. of English Language and Literature
Sogang University
Sinsu 1, Mapo, Seoul
121-742
Phone: 02) 705-8298
E-mail: hblee@sogang.ac.kr

Received: April 30, 2003
Accepted: May 26, 2003