

Computational C_{HL}; A Computational Implementation of Minimalist Syntax

Kazuhiko TAJIMA

Computational C_{HL}; 極小理論の自然言語処理への応用

田 島 和 彦

要 約

本稿は Noam Chomsky の最新統語分析法である極小理論 (Minimalist Program) をコンピュータ上にどのように導入できるのかを考察する。ここでは、極小理論の C_{HL} (Computation of Human Language) の各モジュールである Merge、Move、および C-command を使用しない Binding (束縛) 等を示す。自然言語処理において一般的に提唱されている Principle-based Parsing とは異なり、本稿はより極小理論の分析法に則した Strongly Derivational Universal Rule-Based Parsing を紹介する。

Key words: Natural Language Processing, Minimalist Syntax, Noam Chomsky,
Strongly Derivational Rule-based Parsing

1 Introduction

The purpose of this paper is to outline an algorithm for a computational implementation of Minimalist syntax¹. It has been claimed that organs of the mind are best captured in computational terms (i.e., mathematical patterns in nature) (e.g., Boeckx (2006) and Gallistel (2005), among others). Chomsky (1995) dubs the syntactic computation in the mind C_{HL} , read as Computation of Human Language. It includes syntactic computations such as Merge (Generalized Transformation formation), Move (Singular Transformation) and AGREE (feature checking operation), etc. Given the term C_{HL} , there is in principle no a priori reason to think that a computational implementation of such an approach should turn out to be impossible. In particular, I will describe how a parsing model for Binding and Movement can be better accomplished based on a model of comprehension/ perception.

The Minimalist parsing model that I will illustrate is based on a comprehension/ perception (i.e., a hearer model), rather than one for generation/ production (i.e., a speaker model). At bottom, the Minimalist Approach is akin to a generation model, and as such, says little on how we comprehend/ perceive a language. The actual designs of Chomsky's various recent models undeniably resemble a sort of bottom-up production (with cyclic derivations)². Yet, what many researchers in computational linguistics and in psycholinguistics are interested in is instead the comprehension/ perception side. Such models form the basis of applied Computational Linguistics. It would also seem necessary for psycholinguists working on language perception/ processing issues. (cf. Siloni (2005) on Garden Path sentences, for example.)

Although it appears to be the case that the distinction between a speaker or a hearer design is generally overlooked, I will claim that we must envisage a theory of C_{HL} , based upon a comprehension/ perception design, that is compatible with either the speaker or the hearer models. I maintain that “the chicken or the egg” in the grammar (that is, a speaker-based model or a hearer-based model) turns on an important theoretical question.

The parsing model that I advocate is also strongly derivational in the sense of Epstein et al. (1998) and Epstein & Seely (2006), in that representational constructs are entirely eliminated. It is strongly derivational, since operations such as Binding, as illustrated below, are satisfied “online” in the sense of Lasnik (2001), at a point in a derivation without the notion c-command (Epstein et al. (1998)). Also, following Epstein & Seely (2006), in which A-Movement is eliminated, I will discuss how Move can be dispensed with. This parsing model, hence, contrary to the principle-based parsing methods, inherits

what Epstein et al. (1998) calls the strongly derivational universal-rule approach.

Previous computational implementation of generative grammar, such as Berwick and Fong (1996), Berwick and Fong (1996), Berwick et al (1999), Fong (2004) and Stabler (1992, to appear), for example, are confined to either representational (principle-based) or generation designs, or both.

In next section, I will define a recursive structure building operation in computational terms. In the section that follows, I will outline a feature copying (or movement) convention that replaces the literal movement to account for unbounded dependencies. Chomsky (1995), incidentally, suggests that his feature movement is more economical than movement of words. In section 4, I will suggest how the core cases of binding of anaphors can be captured derivationally in terms of the same feature copying convention, but without using the notion c-command. Finally, the final section suggests how this hearer design can be extended to a speaker model.

2 Parsing Merge

The parser that I employ in this project is a bottom-up chart parser that Allen (1995), for instance, depicts with feature *unification* ability (cf. Shieber (1986)). What it does is concatenate two “terms” (constituents) X and Y, and forms a set that contains three terms $\{X, \{X, Y\}\}$ ³. This corresponds to the operation Merge in terms of Chomsky’s (1994) basic procedure of Bare Phrase Structure. What the operation Merge does is, then, combine at most and at least two terms to form a set out of them.

Assuming that Lexical items are comprised of sets of phonological, semantic, and formal (syntactic) features⁴, and that Move as well as Merge are morphologically driven (i.e., triggered by some features in lexical items), Merge comes into play when a head term containing an uninterpretable (or unvalued) feature triggers concatenation with a term that involves some matching feature that deletes the uninterpretable feature in the head. Thus, a term that contains an uninterpretable feature is the head term that triggers Merge (cf. Adger (2003) and Veselovska (1997)). For instance, in the VP *like it*, the transitive verb *like* contains the uninterpretable c-selectional (subcategorization) [CF uD] feature (CF = Complement Feature), and triggers Merge with *it* that contains the [CAT D] feature (CAT = Category)⁵.

Merge is then achieved in the basic schemata (1), where for the purposes of utilizing *unification* in parsing, variables are indicated with the prefix “?” as is customarily done in computer languages. Note that, although the arrow “ \rightarrow ” is used, it is not a Context Free Grammar, but a Grammar that concatenates two sets X and Y, whose linear realization is either $X \text{ — } Y$ or $Y \text{ — } X$ order. (CF = Complement Feature, SF = Specifier Feature, where each feature is defined as the ordered pair of Attribute and Value). For instance, the VP

like it is rendered as {like, {like, it}}, where *like* is the label (indicating its head) and “{like, it}” a 2-membered (order-irrelevant) set⁶.

```
(1)  {{{[CAT ?A] [LEVEL ?L] [TNS ?T]...[PF ?X]}
      {CF: [CF uD] [CASE uAcc]...}
      {SF: [SF uD] [CASE Nom]...}}
     {{{[CAT ?B] [LEVEL ?K] [TYPE pron]
      [P ?P] [N ?N] [G ?G] [CASE Acc] [PF ?Y]}
      {CF: [CF nil]...} {SF: [SF nil]...}}}
```

(UNLESS

```
  (OR (IF (OR (AND (EQ ?L (+MIN-MAX))
                 (OR (EQ ?K (+MIN-MAX))
                     (EQ ?K (-MIN+MAX))))
        (AND (OR (EQ ?L (+MIN-MAX))
                 (EQ ?L (-MIN+MAX))
                 (EQ ?K (+MIN-MAX))
                 (AND (OR (EQ ?L (-MIN+MAX))
                         (EQ ?L (+MIN-MAX))
                         (EQ ?K (-MIN-MAX))))
              (PRINT (?LL (-MIN+MAX))))
        (IF (AND (EQ ?L (+MIN-MAX))
                 (OR (EQ ?K (+MIN-MAX))
                     (EQ ?K (-MIN+MAX))))
            (PRINT (?LL (-MIN-MAX))))
    (EXIT_ABORT))
```

```
→ {{{[CAT ?A] [LEVEL ?LL] [TNS ?T]...[PF (?X ?Y)]} /*←label*/
   {CF: [CF uD] [LEVEL max] [P ?PP] [N ?NN]
   [G ?GG] [CASE Acc]...}
   {SF: [SF uD] [CASE Nom]...}
   {{{[CAT ?A] [LEVEL ?L] [TNS ?T]...[PF ?X]}
    {CF: [CF uD] [CASE uAcc]...}
    {SF: [SF uD] [CASE Nom]...}}
   {{{[CAT ?B] [LEVEL ?K] [TYPE pron] [P ?P] [N ?N]
    [G ?G] [CASE Acc] [PF ?Y]}
    {CF: [CF nil]...}
    {SF: [SF nil]...}}}
```

(1) concatenates two “terms” X and Y, and forms a set that contains three terms {X, {X, Y}}. In accordance with (1), structures are built incrementally. Structure building is Markovian; the grammar only looks at what it has in deciding what to do next and allow neither look-ahead nor look-back. The phrasal status of a term is derivationally determined. For instance, X is a head term, minimal projection [LEVEL (+MIN, −MAX)], iff X contains an uninterpretable c-selectional feature that requires an agreeing term Y. Y, on the other hand, can be either [LEVEL (+MIN, −MAX)] or [LEVEL (+MIN, +MAX)] if it is a terminal term that doesn’t project. If Y is phrasal, then it is of course [LEVEL (+MIN, +MAX)]. X is a phrasal term if it meets the criteria in (2).

- (2) X is a phrasal term (a maximal projection) if
- a) all c-selectional features are checked off, or
 - b) X agrees with Y and Y deletes an uninterpretable feature in X.

(2a) corresponds to Adger’s (2003) proposal. In addition, (2b) is necessary to take care of cases when X is a maximal projection but its uninterpretable features are not checked off, but copied onto its dominating node. This process is discussed in next section. All other instances are intermediate projections: that is, [LEVEL (−MIN, −MAX)].

Clumsy though it appears to be, the restriction clause stated in terms of logical connectives in the rule (1), repeated in (3), ensures all the possible concatenation patterns listed in (4) that make up a maximal projection [LEVEL (+MIN, +MAX)] (from (4a) to (4f)), as well as an intermediate projection [LEVEL (−MIN, −MAX)] ((4g) and (4h)). If the restriction clause (3) does not match incoming strings, concatenation fails.

(3) (UNLESS

```

(OR (IF (OR (AND (EQ ?L (+MIN−MAX))                /*(4a/4b)*/
              (OR (EQ ?K (+MIN−MAX))
                  (EQ ?K (−MIN+MAX))))))
  (AND (OR (EQ ?L (+MIN−MAX))                /*(4c/4d)*/
        (EQ ?L (−MIN+MAX)))
    (EQ ?K (+MIN−MAX)))
  (AND (OR (EQ ?L (−MIN+MAX))                /*(4e/4f)*/
        (EQ ?L (+MIN−MAX)))
    (EQ ?K (−MIN−MAX))))
(PRINT (?LL (−MIN+MAX))))
(IF (AND (EQ ?L (+MIN−MAX))                /*(4g/4h)*/
      (OR (EQ ?K (+MIN−MAX))

```

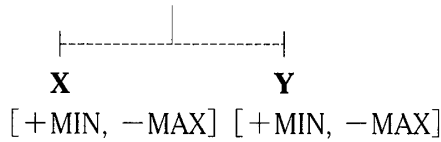
```

(EQ ?K (- MIN + MAX)))
(PRINT (?LL (- MIN - MAX))))
(EXIT_ABORT))

```

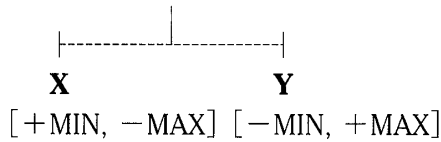
(4) a. *Head-Complement*

$$\mathbf{Z} = [-\text{MIN}, +\text{MAX}]$$



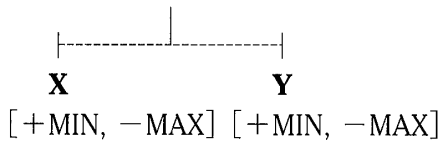
b. *Head-Complement*

$$\mathbf{Z} = [-\text{MIN}, +\text{MAX}]$$



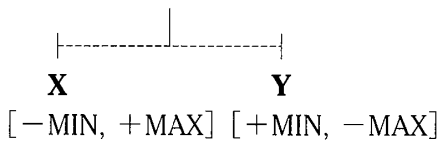
c. *Specifier-Head*

$$\mathbf{Z} = [-\text{MIN}, +\text{MAX}]$$



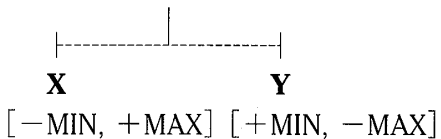
d. *Specifier-Head*

$$\mathbf{Z} = [-\text{MIN}, +\text{MAX}]$$

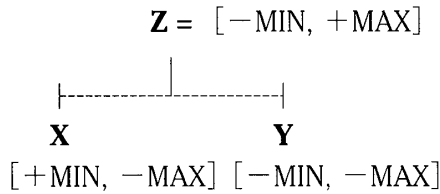


e. *Specifier-Intermediate Projection*

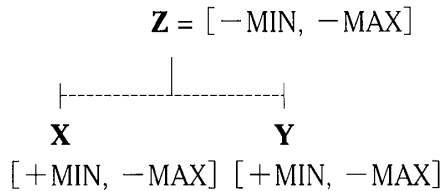
$$\mathbf{Z} = [-\text{MIN}, +\text{MAX}]$$



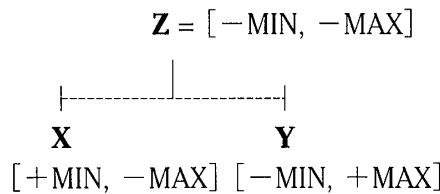
f. *Specifier-Intermediate Projection*



g. *head-complement (Intermediate Projection)*



h. *head-complement (Intermediate Projection)*



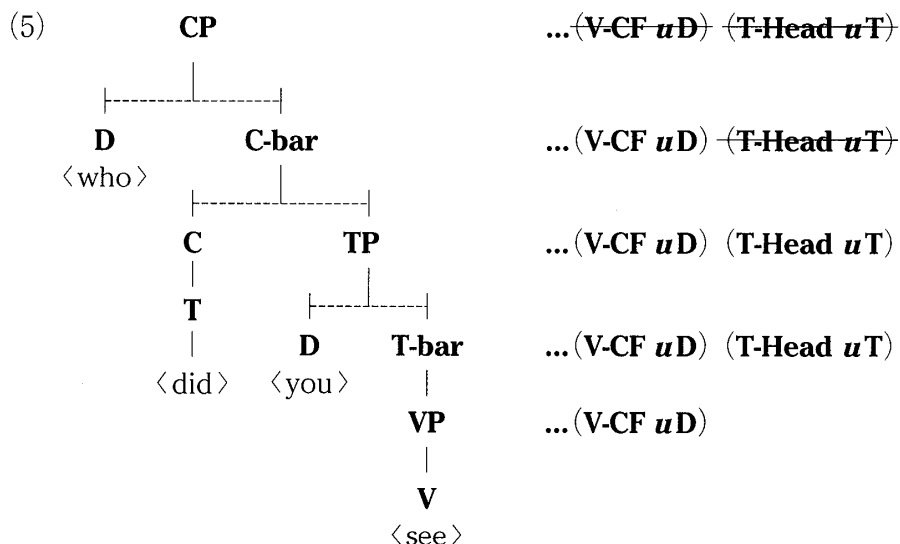
Chomsky (1995; 231) states that features can be either intrinsic or optional. I will claim that Complement Features are intrinsic whereas Specifier Features can be either intrinsic or optional. The intrinsic Specifier Features, as far as English is concerned, include so-called the EPP features: $[\text{SF } uD]$ in T, $[\text{SF } uWH]$ in C and transitive v^7 . On the other hand, optional Specifier Features occur in head terms when the heads occur with optional specifiers. Following Adger (2003), I assume that Merge applies when a head term contains an uninterpretable feature that demands merger with another term with an appropriate feature that deletes the uninterpretable feature in the head. Thus, when an optional specifier appears, the head that contains it must involve a specifier feature for it in its projection.

This assumption makes it possible to maintain a stronger Minimalist claim of economy to the effect that ALL syntactic operations are necessitated by lexical (uninterpretable or strong) features. This is even more so if one wishes to maintain that Move is decomposed into Copy and Merge. Hence, whenever UG allows a syntactic operation (Move or Merge), then it is obligatory. In other words, “if UG allows it, then it must apply” (Lasnik & Uriagereka (2005; 66)).

3 Parsing Move

Recall that I am illustrating a comprehension/ perception model. Thus, upon receiving

an input string like *Who did you see?* (abstracting away from the VP-internal Subject as well as the VP shell analyses for the purposes of exposition), the parser generates the tree structure in (5), capturing the head movement of *did* and Wh-movement of *who*. Note that structures are built up in the bottom-up fashion in (5).



Basically, I claim that although the Binary Branching Hypothesis is maintained, when a binary branching tree becomes a unary branching tree, it implies the application of Move. Thus, as depicted in (5), in accordance with Chomsky's (1995) Inclusiveness Condition (i.e., the LF object must be built only from the features of the lexical items), the undeleted uninterpretable Complement Feature of *see* (shown as (V-CF uD)) can be copied (or percolated) onto its mother term, and this copying can reiterate until an appropriate term (*who* in Spec CP in (5)) can match and delete the "V-CF uD" feature (as shown as ~~(V-CF uD)~~ in (5)) within some local domain such as a Phase. If, on the other hand, such a feature is not deleted, the derivation is said to "crash" as in Chomsky (1995). (The Movement of T is indicated as the feature (T-Head uT))⁸.

To recapitulate, this "copy (or move) feature" approach is in compliance with the binary branching hypothesis, in that when a unary branching configuration occurs, it is because there is a "hidden" branching. Such a hidden branching only takes place when some uninterpretable features are copied onto its mother node. Since feature movement of some sort is necessary anyhow, this approach should be justified in the theory of grammar. Thus, this system obeys the binary branching hypothesis. Notice also that this analysis holds for movements targeting specifiers; the so-called Wh-movement and A-movement. I leave open how movement creating adjunction structures is to be dealt with (see fn. 6).

One might notice that this analysis that allows the option of copying/ percolation of uninterpretable features comes to resemble the SLASH feature of GPSG/ HPSG; however, I

claim this is a logically necessary move, if we take seriously what the Inclusiveness Condition of Chomsky (1995) asserts, along with the binary branching hypothesis within the derivational architecture, as argued for in Epstein & Seely (2005). It is interesting to note that the Minimalist parsing model depicted here now becomes comparable with those parsing theories without movement such as GPSG/ HPSG (cf. Meyers (1994)), albeit the difference in the positions one takes with respect to the Nature vs. Nurture Debate.

4 Parsing Binding without c-command

Chomsky (1995) has suggested that Binding conditions apply at the level of LF; that is, Binding is considered as a representational principle and c-command that expresses possible antecedent — anaphor relations is thought to express representational relations. C-command, originally due to Reinhart (1976), is defined in (5).

(6) *C-command* (from Hornstein, Nunes & Grohmann (2005))

α c-commands β iff (i) β is a sister of α , or (ii) α is a sister of γ and γ dominates β .

Lasnik & Uriagereka (2005) suggests that the core cases of Binding Principle A (that is, an anaphor — reflexive pronouns and reciprocal pronouns — must be bound within some local domain) be recast in terms of AGREE (feature valuation between probe and goal), with its local domain roughly corresponding to Phase. Lasnik & Uriagereka's (2005) version of Principle A that does not depends on the use of indices is stated in (7).

(7) a. An anaphor has a c-commanding antecedent in D

b. X is the binding domain D for Y iff X is the minimal maximal projection which includes Y and Z, where Z is Y's probe.

They claim (7b) corresponds a Phase, and to what used to be termed as clause mate.

Following Epstein et al (1998) and Epstein & Seely (2006), in the parsing approach that I advocate, I claim that Binding can be defined without c-command, as (8);

(8) Upon merger of α and β , where α is an potential antecedent, and β a construct that includes the uninterpretable features of an anaphor, a binding relationship is established only if the phi-features in α and β match, as long as some Phase does no intervene in the feature inheritance between α and β .

The reflexive pronoun *herself*, for instance, is defined with the following feature specifications in (9). Irrelevant details “...” are omitted.

- (9) *herself*: {[CAT D] [P *u*3rd] [N *u*Sg] [G *u*fem]...[Case *u*Case] [PF *herself*]}

The uninterpretable phi-features (i.e., the 3rd Person, singular Number and feminine Gender) are copied onto the D's mother nodes iteratively until an appropriate item bearing the same feature is found. Thus, Binding Principle A is satisfied “online”, at any point in a derivation, as argued for in Belletti & Rizzi (1988)⁹.

This uninterpretable feature movement is exactly the same as what is needed to capture unbounded dependency effect, and applies as long as it does not cross over the domain defined in (7b). (What counts as a Phase for movement and binding cases must be worked out.) This captures the contrast in (10).

- (10) a. Hilary recommends *herself* for the post.
 b. *Hilary recommends *himself* for the post.
 c. *Hilary wonders if George recommends *herself* for the post.

Among the examples in (10), only (10a) converges. (10b) is excluded because the uninterpretable masculine Gender feature of *himself* is not deleted. Similarly, (10c) shows that the uninterpretable phi-features of *herself* are not deleted, because they are prevented from percolating across the domain defined in (7b).

C-command can be dispensed with from the definition of Binding. Consider the contrast in (11).

- (11) a. The president congratulated *herself*.
 b. *Supporters of the president congratulated *herself*.

The ill-formedness of (11b) is traditionally attributed to the lack of c-command requirement in the representational conception of Binding. In the well-formed (11a), the uninterpretable phi-features *herself* are copied onto the node dominating *congratulated herself*. Upon merger with *the president*, they match the phi-features of *the president*, and the uninterpretable features are deleted.

On the other hand, in (11b), the uninterpretable phi-features of *herself* on the node dominating *congratulated herself* do not match those of *supporters of the president*. What is relevant here are the phi-features of the head term *supporters*, and not those of *the president*. Thus, these features are incompatible, and the uninterpretable phi-features of *herself* are not deleted. Specifically, the phi-features of *the president* and those in *herself* are never valued in the computation of (11b). This amounts to show that what c-command was claimed to do can be automatically dealt with in terms of the

uninterpretable feature copying convention.

Furthermore, details aside, this derivational feature copying analysis for Binding should in principle extend to other configurations where c-command is standardly taken to be relevant (see Hornstein, Nunes & Grohmann (2005); examples are theirs).

(12) *Bound Variable Pronoun Licensing*

- a. I gave/sent every check to *its* owner.
- b. ??I gave/sent *his* paycheck to every worker.

(13) *Negative Polarity Item Licensing*

- a. I sent no presents to *any* of the children.
- b. *I sent *any* of the packages to none of the children.

(14) *Minimality of Wh Fronting*

- a. Which check did you send ~~which~~ check to whom?
- b. *Whom did you send which check to ~~whom~~?

The moral of the discussion thus far is that c-command can be dispensed with in an approach that assumes a derivational structure-building with the binary branching hypothesis. In addition, phenomena such as Binding, Bound Variable Pronoun Licensing, Negative Polarity Item (NPI) Licensing, and Minimality of Wh Fronting reduce to the same effect, and can be dealt with in the same feature percolation convention.

5 Derivational vs. Representational models

A representational theory such as Brody (1995) claims that Move be eliminated in favor of Chain Formation, since a theory that contains both concepts is redundant. Succinctly put, the present derivational model comes to resemble the generation/production representational model of Brody (1995) in that both the model I have supported thus far and Brody's model are well-suited with the hearer model, capturing displacement effects with Copy Feature (in my approach) or Form Chain (in Brody (1995)). In this way, a derivational design and a representational one appear to converge. Furthermore, it seems that a speaker design based on the hearer model will offer a formal model simulating aspects of the human language faculty by implementing it as a computer program.

Although both this strongly derivational model and Brody's representational model appear to offer the same empirical coverage, however, my derivational design, in contrast to Brody's model, is more in accord with the current development of the Minimalist program in such a way that the multiple-interface access model in terms of multiple Transfer (Chomsky (2001, 2004)) or multiple Spell-out (Uriagereka (2000)) can be

accommodated¹⁰.

6 Conclusion

Epstein et al. (1998; 20) states “...this strongly derivational universal-rule approach, in which iterative rule-application characterizes syntactic derivations while constraints on output levels of representation (hence levels themselves) are altogether eliminated (see Chomsky 1994, 1995), exhibits explanatory advantages over the existing, representational, ‘rule-free,’ principle-based (hence, representation-based) theories — at least in the domain of accounting for the absolutely central construct ‘syntactic relation.’” Based on this conception of strongly derivational universal-rule approach, I have tried to argue for a feature copying/ movement convention so as to make it possible to run Minimalist Syntax on a computer. The uninterpretable feature manipulations depicted here accommodate the syntactic operations Merge and Move, as well as Binding of anaphors in the same way. I have also suggested that a generation/ production design must be built on the basis of a comprehension/ perception design.

Acknowledgement

I would like to thank J. Emonds, W.L. Lai and J.J. Nakayama for discussions of the issues in this paper. Although references are updated in appropriate places, this paper was originally written in 2003, and is presented here almost as it was then. Enlarged extended versions of this paper will appear elsewhere.

References

- Adger, D. 2003. *Core Syntax — A Minimalist Approach*. Oxford University Press, Oxford, UK.
- Allen, J. 1995. *Natural Language Understanding*—2nd Edition. The Benjamin/Cummings Publishing Co. Inc., Englewood Cliffs, NJ.
- Belletti, A. & L. Rizzi. 1988. “Psych-verbs and theta theory.” *Natural Language and Linguistic Theory*, 6. 291–352. Kluwer Academic Press, Dordrecht, The Netherlands.
- Berwick, R. & S. Fong. 1995. “A Quarter Century of Computation with Transformational Grammar.” in J. Cole, et al. *Linguistics and Computation*. Stanford, CSLI Publications, Stanford, CA.
- Berwick, R. et al. 1999. *Principle-based Parsing: Computation and Psycholinguistics*. Kluwer Academic Press, Dordrecht, The Netherlands.
- Boeckx, C. 2006. *Linguistic Minimalism*. Oxford University Press, Oxford, UK.
- Brody, M. 1995. *Lexico-Logical Form; A Radically Minimalist Theory*. MIT Press, Cambridge, MA.
- Chomsky, N. 1965. *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.
- Chomsky, N. 1994. Bare Phrase Structure. *MIT Occasional Papers in Linguistics*. MIT, Cambridge, MA.
- Chomsky, N. 1995. *The Minimalist Program*. MIT Press, Cambridge, MA.
- Chomsky, N. 2001. “Derivation by Phase” in M. Kenstowicz (ed.) *Ken Hale*. MIT Press, Cambridge, MA.

- Chomsky, N. 2004. "Beyond explanatory adequacy," in A. Belletti (ed.) *Structures and Beyond: The Cartography of Syntactic Structures*. Oxford University Press, Oxford, UK.
- Elman, J. et al. 1996. *Rethinking Innateness*. MIT Press, Cambridge, MA.
- Emonds, J. 2006. "The Three Levels of Insertion in a Phrasal Domain: English Nominalizing Suffixes." Talk presented at Mie University Symposium, Mie, Japan.
- Epstein, S. D. et al. 1998. *A Derivational Approach to Syntactic Relations*. Oxford University Press, Oxford, UK.
- Epstein, S. D. & T. D. Seely. 2006. *Derivations in Minimalism*. Cambridge University Press, Cambridge, UK.
- Gallistel, R. C. (2005) "The nature of learning and the functional architecture of the brain." In *Proceedings of the International Congress of Psychology*. Psychology Press, London, UK.
- Hornstein, N., J. Nunes & K. Grohmann. 2005. *Understanding Minimalism*. Blackwell Publishers, Oxford, UK.
- Kayne, R. 1994. *The Antisymmetry of Syntax*. MIT Press, Cambridge, MA.
- Lasnik, H. 2001 "Derivation vs. representation in modern transformational syntax", in M. Baltin & C. Collins (eds.) *The Handbook of Contemporary Syntactic Theory*. 2001, 197–217. Blackwell Publishers, Oxford, UK.
- Lasnik, H. & J. Uriagereka. 2005. *A Course in Minimalist Syntax; Foundation and Prospects*. Blackwell Publishers, Oxford, UK.
- Meyers, A. 1994. *A Unification-based Approach to Government and Binding Theory*. Ph. D. Thesis. New York University, New York, NY.
- Radford, A. 1997. *Syntax — A Minimalist Introduction*, Cambridge University Press, Cambridge, UK.
- Radford, A. 2004. *Minimalist Syntax — Exploring the Structure of English*. Cambridge University Press, Cambridge, UK.
- Reinhart, T. 1976. *The Syntactic Domain of Anaphora*. Ph. D. Thesis. MIT, Cambridge, MA.
- Rumelhart, D. & J. McClelland. 1986. *Parallel Distributed Processing*. MIT Press, Cambridge, MA.
- Siloni, T. 2005. Grammatical Processing. Paper presented at GLOW 2005.
- Shieber, S. 1986. *An Introduction to Unification-Based Approaches to Grammar*. Stanford, CSLI Publications, Stanford, CA.
- Stabler, E. 1992. *The Logical Approach to Syntax*. MIT Press, Cambridge, MA.
- Stabler, E. 2006. *Computational Minimalism: Acquiring and Parsing Languages and Movement*. Blackwell Publishers, Oxford, UK.
- Tajima, K. 1987. "Wh-Q/Wh-Rel Asymmetries and Conditions on A-bar Chains," in *The Proceedings of the Chicago Linguistic Society 23*, University of Chicago. Chicago, IL.
- Uriagereka, J. 2000. "Multiple Spell-out" in S. D. Epstein & N. Hornstein (eds.) *Working Minimalism*. MIT Press, Cambridge, MA.
- Veselovska, L. 1997. "Possessives in Czech". *Papers in Slavic Linguistics*. Kluwer Academic Press, Dordrecht, The Netherlands.

¹ An introduction to the central technical aspects of the Minimalist syntax can be found in Adger (2003), Chomsky (1995; chap. 1), Hornstein, Nunes & Grohmann (2005), Lasnik and Uriagereka (2005), Radford (1997, 2004). Notice that the model I present here assumes the innate Language Faculty, and differs from those that presuppose some form of connectionism following Elman et

- al. (1996) and Rumelhart & McClelland (1986).
- ² Hence, according to Chomsky's (1995) model, a lexical array called NUMERATION is created from the lexicon, which forms a set of pairs (LI, i) , where LI is a lexical item and the index i indicates the number of occurrences of that lexical item available for computation. Then, the operation called SELECT picks out relevant lexical items from the numeration, reducing its index by 1, for the operation Merge to construct binary branching constituent structures. As a result of some iterative applications of these, along with Move, a root sentence is constructed. See also Chomsky (1965; Chap. 1).
- ³ Chomsky (1994; 12) offers the following definition of term.
- For any structure K ,
- i. K is a *term* of K (the entire set or tree is a term), and
 - ii. if L is a *term* of K , then the members of the members of L are *terms* of K .
- If the basic "underlying" structure is universally head-initial as in Kayne (1994) or head-final as in Emonds (2006), a simple Context-Free Grammar parser seems to suffice, in stead of a parser based on the bare phrase structure analysis.
- ⁴ In the text, for the sake of discussion, irrelevant semantic and phonological features are simply ignored, as are some irrelevant formal (syntactic) features.
- ⁵ Adger (2003; 333) claims that all the c-selectional features of an element have to be checked before that element is merged with a c-selecting head term, which is expressed in terms of [-interpretive] feature matching. He proposes the following Hierarchy of Projections to impose selectional properties among functional categories, which is stated as,
- i. Clausal: $C > T > (\text{Neg}) > (\text{Perf}) > (\text{Prog}) > (\text{Pass}) > v > V$
 - ii. Nominal: $D > (\text{Poss}) > n > N$
- ⁶ I will refrain from discussing the operation Adjoin, which seems to be "everyone's perennial trouble maker" (Lasnik & Uriagereka (2005; 280)), defined (in Chomsky 1994) to from an ordered pair of the label, as in $\langle X, X \rangle, \{X, Y\}$.
- ⁷ See, for example, Adger (2003) and Radford (2004), for such claims. Epstein & Seely (2006), inter alia, discusses the problems in the analyses with EPP features.
- ⁸ It so happens that this analysis does not allow us to maintain the Copy Theory of Movement. One of the often cited arguments for it comes from the chain binding effect. Consider (i), in which the ambiguity of binding options is claimed to be possible due to the copies left behind.
- (i) a. Which pictures of herself did [Pecola think [~~which pictures of herself~~ [Claudia had lost ~~which pictures of herself~~]]]?
 b. Pecola wonders [~~which pictures of herself~~ [Claudia had lost ~~which pictures of herself~~]]].
- However, such ambiguity is missing in the so-called null operator constructions, such as (ii).
- (ii) Here are pictures of *herself* that [Cyndee feels [Jane has lost ~~pictures of herself~~]].
- In (ii), the antecedent of *herself* can only be the closest subject *Cyndee*. Thus, I conjecture that we still need some form of LF reconstruction. See Tajima (1987) for additional asymmetries between A-bar chains headed by Wh expressions and null operators.
- ⁹ Needless to say, just as parsing predicts the structural ambiguity in *The soldiers shot the boy with guns*, we also expect to find ambiguity in the Binding in (i) and (ii), because equally economical alternative derivations converge with the parser (cf. Lasnik (2001)).
- (i) Dryads told Nereids stories about *himself*.

(ii) Which pictures of *herself* did [Pecola think [Claudia had lost]]?

- ¹⁰ Hornstein, Nunes & Grohmann (2005) compares Chomsky's (1995) approach which rests on the lexicalist assumption that lexical items enter a derivation fully inflected, with Chomsky's (2001, 2004) AGREE-based approach which deals with Inflectional Morphology in syntactic terms (the so-called Distributed Morphology). They claim that these two approaches have "basically the same empirical coverage" (Hornstein, Nunes & Grohmann (2005; 325)), and "it's too early to see which of these alternative approaches to cover movement is on the right track..." (Hornstein, Nunes & Grohmann (2005; 329)). For obvious reasons, Chomsky's (1995) approach is best suited for a model of comprehension/ perception, given that each word in a sequence of sounds that enters our ears is fully inflected. Due to space limitation, I will not discuss this issue here.

(Received October 1, 2007)