[REVIEW]

Explorations in Maximizing Syntactic Minimization

By Samuel D. Epstein, Hisatsugu Kitahara and T. Daniel Seely, Routledge, New York, 2015, xx+242pp.

TAKAHIRO TOZAWA

*Kitami Institute of Technology**

Keywords: *the Minimalist Program, simplest Merge, third-factor principle, interface condition, labeling algorithm*

1. Introduction

The Minimalist Program (MP) is a research program that seeks to explain linguistic phenomena with the minimal theoretical apparatus. The underlying hypothesis of MP is the Strong Minimalist Thesis (SMT): “Language is an optimal solution to legibility conditions” (Chomsky (2000: 96)). In other words, the syntax satisfies interface conditions in the most efficient way. Here, the important aspect is that if the syntax satisfies the interface conditions by using stipulated syntactic devices, the syntax ceases to be an optimal solution to the interface conditions. Then, the task of MP is to eliminate stipulated syntactic devices as far as possible by reducing them to third-factor principles (language-independent principles) and interface conditions. In carrying out this task, operation Merge is assumed as the only indispensable syntactic device. Assuming syntactic devices other than Merge requires empirical justification.

The book under review attempts to account for properties of language without any stipulated syntactic devices following the Minimalist spirit. The book is a collection of eleven papers written by Epstein, Kitahara, and Seely (EKS) (one of them is co-authored with Miki Obata). Each

* I would like to thank Yoshihisa Goto, Satoru Kanno, Ryoichi Kondo, Nobuhiro Miyoshi, Yuta Tanaka, and anonymous *EL* reviewers for their insightful comments and suggestions. All remaining errors and inadequacies are, of course, my own. This work is supported in part by the Japan Society for the Promotion of Science (Grant-in-Aid for Young Scientists (B), No. 15K16769).

2. Overview

(1) a. $[_{CP} C_{\varphi} [_{TP1} T [_{vP} \text{Bill ate rice}]]]$
 b. $[_{CP} C [_{TP1} T_{\varphi} [_{vP} \text{Bill ate rice}]]]$ (C-to-T feature inheritance)
 c. $*[_{CP} C [_{TP2} \text{Bill} [_{TP1} T_{\varphi} [_{vP} \text{Bill ate rice}]]]]$
 (counter cyclic movement of Bill)

- In (1b), C is in a sister relationship with TP₁. However, at the stage of the derivation in (1c), C is no longer in a sister relationship with TP₁. C is a sister of TP₂, which is a violation of (2). Therefore, the derivation in (1c) is disallowed. Here, the problem is how *Bill* moves to [Spec, TP] without violating (2). This problem is avoided if the movement of *Bill* creates another root structure. Specifically, in (1d), *Bill* moves to TP₁ so that two root structures (CP and TP₂) are formed. C is a sister of TP₁, thus satisfying (2). It is important to note that the root is undecided in this multi-rooted structure. If the derivation proceeds further, it is unclear whether a syntactic object (SO) merges with CP or TP₂. This causes the derivation to halt. Therefore, the multi-rooted structure needs to be reduced to a single

root structure. EKS argue that Transfer transforms two roots into a single root. In (1d), Transfer applies to TP₂ (the shaded area) so that only single root CP remains.¹ The same line of argumentation carries over to the vP phase.

Second, EKS assume the principle of recoverability of deletion (Recoverability) and the principle of minimal computation (Minimal Computation) to be general laws.² Recoverability requires that at least one copy of a moved element should be pronounced. Minimal Computation requires that at most one copy should be pronounced. For example, consider (3):

- (3) [TP Bill T [vP Bill [v' v left]]] (cf. p. 182)

There are two copies of *Bill* (one in [Spec, vP] and one in [Spec, TP]). Recoverability allows both copies to be pronounced.³ However, Minimal Computation prohibits both copies from being pronounced. The number of pronounced copies must be at most one. From this, it follows that only one copy of *Bill* is pronounced. Then the question arises of which of the copies is pronounced. EKS follow Chomsky (2013) in assuming that the structurally highest copy is pronounced. In (3), the structurally highest copy is *Bill* in [Spec, TP]. Thus, the copy in [Spec, TP] is pronounced, and the copy in [Spec, vP] is deleted. With regard to copy deletion, EKS assume that a copy is deleted according to its occurrence: all sisters of the occurrence undergo deletion. We emphasize that all sisters of the occurrence undergo deletion. In (3), *v'* is the occurrence of *Bill* in [Spec, vP] and *Bill* is the only sister of *v'*. Then, *Bill* undergoes copy deletion under the order to delete all sisters of *v'*. With the two assumptions in mind, EKS provide a principled account of *that*-trace effects.⁴

- (4) a. Who do you think left?
b. *Who do you think that left? (p. 183)

The derivations of the embedded clauses in (4a, b) are shown in (5a, b), respectively.

¹ One might wonder whether Transfer of CP rather than TP₂ results in a single root structure. EKS state, “the phase-edge must be left for the derivation to continue” (p. 181). For this reason, TP₂, rather than CP, is removed by Transfer.

² EKS (p. 191) suggest that Recoverability and Minimal Computation may be third-factor principles.

³ Note that Recoverability is satisfied provided that one of the copies is pronounced.

⁴ In (4b), the subject trace follows the complementizer *that* as shown in (i).

(i) *Who do you think that *t* left?

The subject trace cannot be right-adjacent to the complementizer *that*. This is called the *that*-trace effect.

- (5) a. $[_{CP_2} \text{who}_3 [_{CP_1} [C_{\text{null}}] \begin{array}{l} \nearrow [_{TP_1} T [_{vP} \text{who}_1 [_{v'} v \text{ left}]]]] \\ \nwarrow [_{TP_2} \text{who}_2 \end{array}]]]$ (cf. p. 184)
- b. $[_{CP_2} \text{who}_3 [_{CP_1} [C \text{ that}] \begin{array}{l} \nearrow [_{TP_1} T [_{vP} \text{who}_1 [_{v'} v \text{ left}]]]] \\ \nwarrow [_{TP_2} \text{who}_2 \end{array}]]]$ (cf. p. 187)

In (5a), *who* moves to TP_1 and CP_1 , forming a multi-rooted structure. Then, there are three copies of *who*: *who*₁ in [Spec, vP], *who*₂ in [Spec, TP], and *who*₃ in [Spec, CP]. Note that *who*₃ is the highest copy. The structural position of each copy is calculated concerning its occurrence. *Who*₂ is structurally higher than *who*₁ because the occurrence of *who*₂ (TP_1) is structurally higher than the occurrence of *who*₁ (v'). Furthermore, *who*₃ is structurally higher than *who*₂ because the occurrence of *who*₃ (CP_1) is structurally higher than the occurrence of *who*₂. Therefore, *who*₃ occupies the structurally highest position so that *who*₃ is pronounced and *who*₁ and *who*₂ undergo copy deletion. Here, let us pay attention to the copy deletion of *who*₂. Its occurrence is TP_1 , and therefore, *who*₂ is deleted under the order to remove all sisters of TP_1 . Here, note that all sisters of TP_1 are deleted. Then, null *C*, which is the sister of TP_1 , is also deleted. The deletion of null *C* is exempt from the Recoverability requirement. This is because null *C* has no phonological feature, and therefore, the deletion of null *C* is irrelevant to whether it satisfies Recoverability. Therefore, the deletion of null *C* does not violate Recoverability. Moreover, deletion of all the copies except for *who*₃ satisfies Recoverability and Minimal Computation. Thus, (4a) is grammatical. Next, let us turn to the derivation in (5b). The derivation is the same as that in (5a) except that the complementizer *that* occupies the *C* head. Here, consider the copy deletion of *who*₂. The occurrence of *who*₂ is TP_1 . Then, *who*₂ undergoes deletion in accordance with the instruction to delete all sisters of TP_1 . The complementizer *that* is also required to be deleted because it is the sister of TP_1 . However, Recoverability requires that *that* should not be removed, because if it were, it would not be pronounced anywhere. These conflicting requirements cause (4b) to be ungrammatical.

Furthermore, EKS account for why the movement of an adjunct in (6), unlike the movement of the subject in (4b), does not show the *that*-trace effect.

- (6) Why do you think [t_{wh} C_{that} [John likes Mary t_{wh}]] (p. 188)
- Under EKS's analysis, the embedded clause has the structure in (7).

- (7) $[_{CP2} t_{wh} [_{CP1} \text{that} \begin{array}{l} \nearrow \\ \searrow \end{array} \begin{array}{l} [_{TP1} T [_{vP} \text{John likes Mary } t_{wh}]]] \\ [_{TP2} \text{John} \end{array}]]]$

Why directly moves to [Spec, CP] without going through [Spec, TP]. Then, the complementizer *that* is not required to be deleted because there is no requirement to delete all sisters of TP₁. Pronouncing *that* satisfies Recoverability. Thus, (6) is grammatical. Previously, the subject-adjunct asymmetry (the contrast between (4b) and (6)) was captured with the help of γ -marking.⁵ However, γ -marking is a language-specific device. Under EKS's analysis, the asymmetry is given a straightforward account without γ -marking.

Chapter 10 seeks to eliminate MOM: Merge is preferred over Move. To begin with, following Chomsky (2013), EKS assume that Merge involves neither projection nor linear order, as shown in (8).

- (8) Merge (α, β) $\rightarrow \{\alpha, \beta\}$ (p. 202)

The merger of α and β creates just the unordered set consisting of α and β . EKS call this *simplest Merge*. According to simplest Merge, the SO formed by Merge has no label. However, SOs need to be interpreted at the interfaces. EKS adopt Chomsky's (2013) labeling algorithm shown in (9) and (10).

- (9) Suppose SO = {H, XP}, H a head and XP not a head. Then minimal search will select H as the label, and the usual procedures of interpretation at the interfaces can proceed.
- (10) Suppose SO = {XP, YP}, neither a head. Here minimal search is ambiguous, locating the heads X, Y of XP, YP, respectively. There are, then, two ways in which SO can be labeled: (A) modify SO so that there is only one visible head, or (B) X and Y are identical in a relevant respect, providing the same label, which can be taken as the label of the SO. (p. 202)

Chomsky argues that the labeling algorithm is just *minimal search*, which may be a third-factor principle. (9) states that the label of the SO consisting of H and XP is H.⁶ (10) dictates that the label of the SO {XP, YP} cannot be determined, and there are two ways to avoid label indeterminacy. The first way is that either XP or YP is moved. For example, con-

⁵ See Lasnik and Saito (1992).

⁶ For example, the SO { v , VP} is labeled as v since minimal search of a lexical item locates the v head.

sider the merger of the subject DP and ν P in (11).

- (11) [DP [_{TP} T [DP [_{ν P} ν VP]]]]

In the SO {DP, ν P}, minimal search locates D and ν , resulting in labeling ambiguity. Therefore, {DP, ν P} cannot be labeled. If DP undergoes movement, the copy of DP is invisible for labeling. Therefore, minimal search locates only the ν head. As a result, {DP, ν P} is labeled as ν . The second way is that if XP and YP share a prominent feature, it is the label of {XP, YP}. For instance, consider the merger of DP and TP in (11). DP and TP share prominent features: ϕ -features. Therefore, {DP, TP} is labeled as ϕ . With Chomsky's labeling algorithm in mind, let us turn to the contrast between (12a) and (12b).

- (12) a. *There is likely [_{TP} a man to be t in the room].
 b. There is likely [_{TP} t to be a man in the room]. (p. 205)

Thus far, this contrast has been explained by MOM. According to MOM, Merge of *there* in embedded [Spec, TP] in (12b) overrides the movement of *a man* to embedded [Spec, TP] in (12a). Thus, (12b) blocks (12a). EKS argue that the contrast is predicted by Chomsky's (2013) labeling algorithm. In (12a), DP *a man* internally merges with TP, forming the SO {DP, TP}. DP and TP do not share ϕ -features because non-finite T has no ϕ -features. Then the label of {DP, TP} cannot be determined, which is in violation of Full Interpretation (an interface condition). Therefore, (12a) is ungrammatical. In (12b), DP *there* externally merges with TP, forming {DP, TP}. It is different from (12a) in that it moves to the matrix clause at the later stage of the derivation. Then, {DP, TP} is labeled as T because DP is a copy and is invisible for labeling. Therefore, (12b) is grammatical. This way, the contrast is explained without resorting to MOM. Thus, MOM can be eliminated from the theory of grammar.

Chapter 11 argues that the criterial freezing effects are attributed to morpho-phonological, conceptual-intentional (C-I) requirements rather than a syntactic constraint on movement. Let us consider (13a, b).

- (13) a. You wonder [which dog C [_{TP} John likes t]].
 b. *Which dog do you wonder [t C [_{TP} John likes t]]?

(cf. p. 227)

In (13a), *which dog* moves to the embedded interrogative CP edge and stays there. In (13b), *which dog* moves further from the embedded interrogative CP edge, which is not allowed. This is the criterial freezing effect. EKS introduce Rizzi's (2014) syntactic constraint-based analysis and point out some problems. Under Rizzi's analysis, (13a, b) have the structures in (14a, b).

- (14) a. You wonder [_Q [_Q which dog] [_Q C_Q [_{TP} John likes *t*]]]
(cf. p. 227)

- b. Which dog do you wonder [_Q *t* C_Q [_{TP} John likes *t*]]

In (14a), DP *which dog* internally merges with CP. According to Chomsky's labeling algorithm in (10), the SO {DP, CP} is labeled as Q because DP and CP share a prominent feature Q. Then, note that label Q of {DP, CP} becomes the maximal projection of Q, and Q of *which dog* becomes the non-maximal projection. This is because the Q feature of *which dog* "projects" to the higher phrase consisting of DP and CP. Consequently, *which dog* in the CP edge becomes a non-maximal projection. Rizzi proposes the syntactic constraint on movement shown in (15).

- (15) Maximality: Only maximal objects with a given label can be moved. (p. 227)

(15) prevents non-maximal elements from undergoing movement. Now, let us turn to (14b). Rizzi ascribes the criterial freezing effect to the violation of (15). As we have already seen, *which dog* in the embedded CP edge is a non-maximal projection. Then, further movement of *which dog* from the CP edge violates (15). Therefore, (14b) is ungrammatical.

However, EKS argue that there are some problems with Rizzi's (2014) analysis. They point out that appealing to the UG-specific constraint in (15) is not desirable given that the task of MP is to reduce language-specific devices to third-factor principles and interface conditions. Furthermore, EKS argue that (15) prevents the unification of External Merge (EM) and Internal Merge (IM). (15) is a constraint on IM (movement), not EM. Consider (16).

- (16) [_{TP} T [DP [_{VP} v VP]]]

The label of the SO {DP, vP} is not determined. That is, it is unclear whether the SO is a maximal element or not. In spite of that, the SO externally merges with T. From this then, it is plausible that (15) does not apply to EM. Then, (15) leads to a differentiation between EM and IM. Assuming EM and IM increases the richness of UG, which is inconsistent with the Minimalist guidelines: reduce the content of UG to the minimum.⁷ Thus in contrast to Rizzi's (2014) syntactic constraint-based analysis, EKS present an interface-based analysis of criterial freezing effects. EKS's analysis is indicated in (17).

⁷ Chomsky (2004) unifies EM and IM into one operation Merge.

(17) *Which dog do you wonder [α t [C_Q [$_{TP}$ John likes t]]]? (p. 230)

First, *which dog* moves to the embedded CP, forming {DP, CP}. It undergoes further movement to the matrix clause so that embedded clause α is labeled as C_Q because DP is a copy and is invisible for labeling.⁸ EKS argue that there is no problem with movement from the embedded CP edge but the movement causes a violation of interface requirements. EKS assume (i) that the label of C_Q yields a yes-no question interpretation, and (ii) that yes-no questions require T-to-C movement and rising intonation. Then, C_Q (the label of α) in (17) results in a yes-no question interpretation, and therefore, it requires T-to-C movement and rising intonation. The yes-no question interpretation leads to gibberish at the C-I interface, and the requirement of T-to-C movement and rising intonation causes a morpho-phonological problem. Let me explain this in detail. Regarding gibberish at the C-I interface, EKS note that yes-no questions are interpreted to be a request for information. For example, the yes-no question *does John like this dog* is interpreted as “Answer this, Does John like this dog?” If the yes-no question is the complement of *wonder* as in *I wonder* [$_{yes-no\ Q}$ *John like this dog*], it means, “I wonder, answer this, Does John like this dog?” However, we cannot wonder a request for information. Thus, the yes-no question interpretation of the embedded clause in (17) yields gibberish at the C-I interface.

With regard to the morpho-phonological problem, EKS assume that English embedded clauses ban T-to-C movement and rising intonation. Then, although C_Q (the label of α) in (17) requires T-to-C movement and rising intonation, they are banned because the clause headed by C_Q is the embedded clause. Therefore, the label C_Q of the embedded clause does not satisfy the morpho-phonological requirement.

In this way, the criterial freezing effects are ascribed to the violation of the morpho-phonological, and C-I requirements.⁹ Note that EKS’s analysis does not face the problems that Rizzi’s (2014) analysis faces. EKS’s analysis accounts for criterial freezing effects without the UG-specific constraint in (15). If (15) is eliminated from the theory of grammar, we no longer need to distinguish EM from IM. Then, EM and IM are unified into only

⁸ See the discussion of (10) and (11) above.

⁹ Note that Chomsky (2015) attributes the criterial freezing effects to gibberish at the C-I interface. EKS’s analysis is different from Chomsky’s in that criterial freezing effects result not only from the C-I problem but also from the morpho-phonological problem.

one operation Merge, which is compatible with the Minimalist guidelines.

3. Discussion and Concluding Remarks

The most distinguishing characteristic of the book under review is that it seeks to explain linguistic phenomena with minimal theoretical apparatus. As we have seen, *that*-trace effects are attributed to Recoverability and Minimal Computation, which are candidates of third-factor principles, the contrast between (12a) and (12b) is explained without MOM, and criterial freezing effects are explained without any syntactic constraint on movement. However, problems remain. In this section, we restrict our attention to the *that*-trace effects and multi-rooted structures and indicate some residual problems. Let us start by discussing the ungrammaticality of (18a):

- (18) a. *Robin met the man that Leslie said that was the mayor of the city. (p. 190)
- b. Leslie said [_{CP2} OP₃ [_{CP1} that
-
- [TP1 [T was] [OP₁ the mayor of the city]]]
- [TP₂ OP₂

(18a) shows that the null operator movement exhibits the *that*-trace effect. The derivation of the embedded clause is indicated in (18b). The null operator moves to TP₁ and CP₁, forming a multi-rooted structure. Notice that the null operator has no phonological feature. Then, the deletion of the phonological feature of OP₂ does not occur. Therefore, the deletion of *that* (the sister of TP₁) also does not occur because there is no requirement to delete all sisters of TP₁. Thus, the derivation in (18b) should have no problem at PF. Although, EKS note, “the deletion algorithm does not distinguish *wh*-elements such as *who* and a null operator OP” (p. 192, note 8), EKS need to explain why a null operator behaves as though it had a phonological feature. Concerning this potential problem, an anonymous reviewer suggests one possible solution: the relative operator in (18b) is a deleted *wh*-phrase in accordance with Chomsky (1977), rather than a null operator. Then the structure of the relative clause in (18a) is (19) (irrelevant details omitted).

- (19) the man [_{CP3} who₄ that Leslie said [_{CP2} who₃ [_{CP1} that
-
- [TP1 [T was] [who₁ ...]]]
- [TP₂ who₂

Not only *who*₂ but also *that* undergoes deletion under the order to delete

all sisters of TP_1 . However, Recoverability bans deletion of *that*. This contradiction causes the *that*-trace effect. Although this analysis may be convincing, there remains an apparent problem with the analysis: the highest copy *who*₄ is also deleted. Deletion of *who*₄ violates Recoverability because *who* is not pronounced anywhere. Then, we incorrectly exclude the grammatical sentence *Robin met the man that Leslie said was the mayor of the city* owing to the Recoverability violation. Drawing inspiration from the reviewer's suggestion, we suggest yet another solution to the problem of the ungrammaticality of (18a). The problem might be solved if we adopt the head raising analysis of relative clauses (Schachter (1973) and Kayne (1994)). Under Kayne's analysis, *man* moves from the base-generated position to [Spec, CP_3] of the relative clause. That is, the copies of *man* occupy the positions of the copies of *who* in (19). Then, the copy of *man* in [Spec, CP_3] of the relative clause is pronounced, and the other copies of *man* are deleted. When *man* in [Spec, TP] is deleted under the order to delete all sisters of TP_1 , the complementizer *that* (the sister of TP_1) is also deleted. However, Recoverability prohibits deletion of *that*. Thus, (18a) is ungrammatical. In this way, if we adopt the head raising analysis, the problem might be solved.

Next, let us discuss multi-rooted structures, which are crucial for the account of *that*-trace effects. We focus on the derivation of a vP phase. For example, consider the derivation of the vP phase of *what did John buy?*

In (20), v -to- V feature inheritance occurs, and then *what* moves to VP_1 and the vP edge, thus creating a multi-rooted structure. Furthermore, *buy* undergoes head-movement to v , thus deriving the vP structure.¹⁰ Here, let us focus on the copy deletion of *buy*₁ and *what*₂. With regard to *buy*₁, it is deleted with reference to *what*₁, which is the occurrence of *buy*₁. With regard to *what*₂, it is deleted with reference to VP_1 , which is its occurrence. Here, notice that *buy*₂ is also the sister of VP_1 . Then, deletion applies to *buy*₂ as well as *what*₂ under the order to delete all sisters of VP_1 . However, deletion of *buy*₂ would violate Recoverability: if *buy*₂ were deleted, it would not be pronounced anywhere. This way, although *buy*₂

¹⁰ Although head-movement occurs at PF, *buy* is taken to occupy v when the copy deletion (a PF operation) applies.

must be deleted, Recoverability prevents it from being deleted. This is a contradiction. Thus, *what did John buy* should be ungrammatical, contrary to fact. Regarding this potential problem, an anonymous reviewer suggests that *buy*₂ may not be deletable if we assume that Recoverability requires that deleted elements must be semantically recoverable. Note that the *v*-V complex (*v-buy*₂) has semantic content. Then, the deletion of *v-buy*₂ violates Recoverability because if it were deleted, it would not be semantically recoverable. Therefore, *v-buy*₂ is not deleted and remains, as it is. Incidentally, note that the deletion of *that* in (19) does not violate Recoverability contra EKS. This is because it has no semantic content. Therefore, the deletion of *that* does not violate Recoverability although it is deleted together with *who*₂ under the order to delete all sisters of TP₁. The deletion of *that* is obligatory because there is a requirement to delete all sisters of TP₁. Therefore, the *that*-trace effect is accounted for. Although the anonymous reviewer's suggestion is persuasive, a question arises as to how only the copy of *what*₂ in (20) is deleted. Specifically, if *what*₂ is deleted under the order to delete all sisters of VP₁, *v-buy*₂ (the sister of VP₁) is also deleted. Therefore, if *v-buy*₂ is retained as the reviewer suggests, we need to explain about how only *what*₂ undergoes copy deletion.

Another anonymous reviewer suggests that the problem of the derivation of *what did John buy* can be solved if we assume with Fukui and Takano (1998) that head-movement is a case of movement to a specifier. Consider (21).¹¹

The only difference between (20) and (21) is that *buy* moves to [Spec, *vP*] in (21). Deletion applies to *what*₂ and *v*, which are sisters of VP₁. It is important to note that deleting *v* does not violate Recoverability because *v* has no phonological feature. Therefore, the problem can be solved if we adopt Fukui and Takano's (1998) view that head-movement is a case of movement to a specifier.

Although there may be some problems to solve, EKS successfully explain linguistic phenomena with minimal theoretical apparatus. Therefore, this book sets a good example of how to conduct linguistic research in accor-

¹¹ Note that the bracket structure in (21) does not reflect the word order among lexical items and phrases. It only represents structural relations.

dance with SMT. I highly recommend this book to those interested in MP.

REFERENCES

- Chomsky, Noam (1977) "On *Wh*-Movement," *Formal Syntax*, ed. by Peter W. Culicover, Thomas Wasow and Adrian Akmajian, 71–132, Academic Press, New York.
- Chomsky, Noam (2000) "Minimalist Inquiries: The Framework," *Step by Step: Essays on Minimalist Syntax in Honor of Howard Lasnik*, ed. by Roger Martin, David Michaels and Juan Uriagereka, 89–155, MIT Press, Cambridge, MA.
- Chomsky, Noam (2004) "Beyond Explanatory Adequacy," *Structures and Beyond: The Cartography of Syntactic Structures*, Volume 3, ed. by Adriana Belletti, 104–131, Oxford University Press, Oxford.
- Chomsky, Noam (2007) "Approaching UG from Below," *Interfaces + Recursion = Language?: Chomsky's Minimalism and the View from Syntax-Semantics*, ed. by Uli Sauerland and Hans-Martin Gärtner, 1–29, Mouton de Gruyter, Berlin.
- Chomsky, Noam (2008) "On Phases," *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*, ed. by Robert Freidin, Carlos P. Otero and Maria Luisa Zubizarreta, 133–166, MIT Press, Cambridge, MA.
- Chomsky, Noam (2013) "Problems of Projection," *Lingua* 130, 33–49.
- Chomsky, Noam (2015) "Problems of Projection: Extensions," *Structures, Strategies and Beyond: Studies in Honour of Adriana Belletti*, ed. by Elisa Di Domenico, Cornelia Hamann and Simona Matteini, 3–16, John Benjamins, Amsterdam.
- Fukui, Naoki and Yuji Takano (1998) "Symmetry in Syntax: Merge and Demerge," *Journal of East Asian Linguistics* 7, 27–86.
- Kayne, Richard S. (1994) *The Antisymmetry of Syntax*, MIT Press, Cambridge, MA.
- Lasnik, Howard and Mamoru Saito (1992) *Move α : Conditions on Its Application and Output*, MIT Press, Cambridge, MA.
- Rizzi, Luigi (2014) "Cartography, Criteria, and Labeling," ms., University of Geneva.
- Schachter, Paul (1973) "Focus and Relativization," *Language* 49, 19–46.

[received July 15, 2016, revised and accepted February 3, 2017]

Common Courses of Human Sciences
 Kitami Institute of Technology
 165 Koen-cho, Kitami-shi
 Hokkaido 090-8507
 e-mail: tozawata@mail.kitami-it.ac.jp