[REVIEW]

A Syntax of Substance

By David Adger, Linguistic Inquiry Monograph 64, MIT Press, 2013, x+189pp.

NOBU GOTO

Toyo University

Keywords: *roots, Self Merge, Universal Extended Projection, PP Peripherality*

1. Introduction

Chomsky (2008) argues that structure building is done via *Merge*, which takes two syntactic objects α and β , forming $\{\alpha, \beta\}$, and, on the other hand, structure labeling is done via a *labeling algorithm* that operates to identify the *heads* of structures built by *Merge* through *minimal search*. So, if *Merge* takes a head *H* and a phrase *XP*, forming $\{H, XP\}$, the *labeling algorithm* identifies *H* of $\{H, XP\}$ by *minimal search* as head, labeling $\{H, XP\}$ as *H*. As is well known, this system cannot uniquely identify a head in *H-H* structures where both constituents are heads and *XP-YP* structures where neither *XP* nor *YP* is a head. Thus, in response to this labeling problem, Chomsky (2013), particularly focusing on *XP-YP* structures, developed a system that can overcome the labeling problem by making use of Moro's (2000) principle of *dynamic antisymmetry* and the notion of *feature-sharing* (*Agree*). In the book under review, entitled *A Syntax of Substance*, David Adger crucially departs from Chomsky's approach and proposes a new system, where the labeling problem does not arise. Here, Adger attempts to defend his proposed system by providing two generalizations concerning relational nominals: "The Optionality Generalization for Relational Expressions (OGRE)" and "PP Peripherality."

In this review, I summarize Adger's system in Section 2, and briefly show his analysis involving the two generalizations in Section 3, and conclude in Section 4 with a brief discussion of a weak point in his system, suggesting a possible approach that is not explored in his book to solve the labeling problem.

2. The Proposals and Consequences

First, Adger proposes that “there are no functional heads,” assuming that a single lexicon consists of “RLex” and “CLex” (p. 21):

- (1) a. RLex = $\{\sqrt{1}, \dots, \sqrt{n}\}$, the set of LIs (roots)
- b. CLex = $\{l_1, \dots, l_n\}$, the set of category labels

He then proposes that the operands of *Merge* may be identical, and if they are identical, the output of *Merge* is a singleton set. Calling such an operation “Self Merge,” he schematizes the three possibilities of *Merge* as follows (p. 19):

- (2) a. Merge(X, Y), X distinct from Y, $\rightarrow \{X, Y\}$ (*External Merge*)
- b. Merge(X, Y), X part of Y, $\rightarrow \{X, Y/X\}$ (*Internal Merge*)¹
- c. Merge(X, Y), X = Y, $\rightarrow \{X, X\} = \{X\}$ (*Self Merge*)

In his system, it is assumed that RLex, CLex, and *Merge* are responsible for structure building, and that structure is built from RLex plus *Merge* where elements of CLex are simply labels for the structures built by *Merge*.

With respect to structure labeling, on the other hand, Adger adopts Starke’s (2001) view that “there exists an ‘fseq’—a sequence of functional projections—such that the output of *Merge* must respect fseq.” In fact, taking fseqs as “extended projections,” he assumes that the extended projections of any root are given axiomatically by Universal Grammar (UG), defining them as follows (p. 21):

- (3) A Universal Extended Projection of a category C (UEP_C) is a sequence of labels drawn from CLex ($l_s, \dots, l_t$), where l_s is the Start Label and l_t is the Terminal Label.

Hence, in his system, the order of functional categories is specified universally in the UEPs, and extended projections are nothing but a property of UG. For any particular language to employ allowable sequences of the UEPs, Adger assumes the binary Cartesian product of CLex as a set of “Label Transition Function (LTFs),” which he calls Λ :²

- (4) $\Lambda = \text{CLex} \times \text{CLex} = \{\langle N, \text{Cl} \rangle, \langle N, N \rangle, \langle \text{Cl}, N \rangle, \langle \text{Cl}, \text{Cl} \rangle, \langle N, \text{Num} \rangle, \dots\}$

Specifically, he assumes that Λ itself is subject to no constraints, and it al-

¹ Y/X signifies that X is contained in Y.

² The set in (4) is just a possible Λ in a language, particularly for the UEP of N. Cl is the category that a classified noun bears, Num is the category that a counted nominal has, and D is the category of a determined nominal projection (cf. (6)).

lows mappings from any category to any other, and conjectures that part of the acquisition process determines what the content of Λ is, and evidence for this in particular languages is found in the morphology and in the distributional patterns found in the primary linguistic data. To put Λ into practice, he devises a unified “labeling function” as follows, where α and β are taken to be syntactic objects (p. 22):

(5) a. Transition Labeling

If $\alpha, \beta \in \gamma$, then $\text{Label}(\gamma) = \text{some } L \in \text{CLex}$, such that there are (possible nondistinct) f and $g \in \Lambda$ such that $f(\text{Label}(\alpha)) = g(\text{Label}(\beta)) = L$.

b. Root Labeling

$\text{Label}(\{\sqrt{x}\}) = \text{some } L \in \{N, V, A\}$

Adger proposes (5a) as a means of specifying the embedding relation between one functional category and another and (5b) as a way of specifying the category of a root. This labeling function capitalizes on the idea that the order of functional categories must be given anyway. Technically speaking, the function takes an unlabeled syntactic object as its argument and provides it with a label. In labeling, he assumes that *minimal search* is involved in the relevant process: it inspects the unlabeled object to see what it immediately contains and uses that information to provide the new label. So, in his system, the UEPs (given by UG), Λ (a result of the acquisition process allowing only certain transitions between labels) and the labeling function (that operates on the basis of the labels of its immediate constituents and the relevant extended projection) are assumed to be responsible for structure labeling.

To sum up so far, Adger’s system separates structure building from structure labeling. In his system, structure building is done via (1)–(3) (*Merge*) and structure labeling is done via (4) and (5) (*Label*), which is dependent on a language-particular instantiation of the UEPs.

Now then, let us see how Adger’s system works, particularly in English. He assumes that Λ for English is partially specified as in (6) (see footnote 2):

(6) $\Lambda = \{ \langle N, Cl \rangle, \langle Cl, Num \rangle, \langle Num, D \rangle, \dots \}$

Under this assumption, he derives, for example, the root of the word $\sqrt{\text{cat}}$ as in (7) (p. 24):

He argues that this “unary structure” is derived by iterated applications of *Merge* and *Label*, under Self Merge (2c), Root Labeling (5b), and Transition Labeling (5a) (cf. Brody (2000)). This instantiates the UEP of N as a syntactic object that consists of the root $\sqrt{\text{cat}}$ contained in a series of higher structure each bearing a label that respects the relevant UEP sequence. He calls these “Rooted Extended Projections (REPs).”

Adger argues that the same holds in unergative, unaccusative, and transitive structures introducing binary structures, as well. For example, assuming that Λ for English is specified as in (8), he argues that a derivation of “Lilly jumps” can be as in (9) (p. 25):

(8) $\Lambda = \{ \dots, \langle V, v^* \rangle, \langle D, v^* \rangle, \dots \}$

This binary structure is also derived by iterated applications of *Merge* and *Label*, in this case, under Self Merge (2c), Root Labeling (5b), External Merge (2a), and Transition Labeling (5a). In (9), Self Merge first operates to yield the REPs, rooted by $\sqrt{\text{Lilly}}$ and $\sqrt{\text{jump}}$, which are labeled by Root Labeling, and then External Merge operates to concatenate them together, creating a structure where v^* has both a complement and a specifier that is labeled by Transition Labeling.

It is important to note here that in Adger’s system the binary structure is symmetrical. Therefore, to ensure asymmetrical interpretations that need to be imposed by the semantic interface for identification of function-argument structure and by the articulatory or acoustic interface for identification of linear order, he defines notions of complement and specifier that read these asymmetries off of the extended projection information in the tree, as follows: if a mother and a daughter are in the same extended projection, and the daughter is lower in that projection, then the daughter is a complement of the mother; otherwise, the daughter is a specifier of the mother. So, in (9), because V and v^* are in the same extended projection, and V is lower

than v^* , V is taken as the complement of v^* , and because D and v^* are not in the same extended projection, D is taken as a specifier of v^* . In his system, therefore, these relations are treated asymmetrically by both the semantics (where complements are composed before specifiers) and by the linearization systems (where complements are linearized after specifiers). Adger argues that the system that can identify the relevant notions in such a way is compatible with a standard approach to the syntax semantics map (Kratzer (1996)) and the syntax phonology map (Kayne (1994)).

Incidentally, with respect to the phonology and morphology of functional morphemes, Adger stipulates that bound morphemes are the pronunciation of functional categories attached to roots via extended projections, whereas some free functional morphemes are spell-outs of these categories that are not so attached, and other free morphemes, like auxiliaries are spell-out of structures built up from lexical roots, as described above for $\sqrt{\text{cat}}$. Thus, in (9), he stipulates that the third person singular agreement morpheme that will eventually appear on $\sqrt{\text{jump}}$ is realized by pronouncing the sequence of $T-v^*-V-\sqrt{\text{jump}}$ with the subject $\sqrt{\text{Lilly}}$ in the specifier of T .

In Adger's system, therefore, three consequences are brought about: [1] there are no heads; [2] a root cannot *Merge* with another syntactic object; [3] roll-up derivations are impossible. Significantly, his system with the consequence [1] does not face the labeling problem which arises in a system where labels are determined by properties of heads (Chomsky (2008)). This is simply because heads are eliminated and LTFs are adopted in their place (see (4)). The consequence [2] is also an important corollary of his system. In his system, since roots on their own are not in the domain of the labeling algorithm, i.e., roots undergo Self Merge, which creates a structure which can be labeled by Root Labeling (see (5b)), it follows that arguments cannot be introduced as sisters to lexical roots, and the semantic relation between a root and an argument must be negotiated by functional structure, as described above for "Lilly jumps" with v^* . Remarkably, this consideration leads to the conclusion that the notion of " θ -domain" is unnecessary. Arguments can, in principle, be introduced anywhere, and each of a series of syntactic *Merge* operations correlates with the introduction of a semantic argument. In the next section, we see that this forces Adger to propose a new mode of argument introduction. Among others, he seems to regard [3] as the most important consequence. To see why, consider a structure like (10). Suppose that in (10) all nodes labeled X are in the same extended projection (EP), that the subscripts indicate the height of the label in that EP, and that X_3 has moved from inside X_4 (cf. p. 43):

What is important to note here is that it is impossible to determine which node is the complement: since both daughters of X_5 are in the same EP as the mother, lower in that EP, and can potentially be complements, no asymmetry can be imposed by the interfaces and the structure is uninterpretable. Thus, roll-up derivations are ruled out in his system as a consequence of the computational system.

In terms of these outcomes, Adger attempts to defend his system by providing an explanation for the two generalizations: “OGRE” and “PP Peripherality.”

3. The Analysis

Let us briefly review Adger’s analysis of the two generalizations. First, as a property of relational nominals, Adger establishes the following generalization (“OGRE”) (p. 61):

- (11) Across languages, relational nominals systematically take their apparent arguments optionally, in contrast to verbs, which vary idiosyncratically in whether any particular argument is optional.

The relevant contrast is the following (see p. 60 and p. 58):

- (12) a. an edge (of the table)
b. Lilly killed *(the mouse)

Adger points out that optionality is unclear under the standard claim that PPs are generated as complements to their Ns, as in [N PP] (Chomsky (1970) and Jackendoff (1977)). So instead he proposes, in terms of the consequence [2], that PPs are, in fact, some structural distance from their Ns and are generated as specifiers to “ $\bar{p}$.”³ He assumes that $\bar{p}$ is a relational category that is responsible for the function-argument structure that encodes relationality and for the introduction of the prepositional case-marking morphology. To ensure the separate introduction of Ns and PPs through use of $\bar{p}$, he proposes that there are two modes of argument introduction: (I)

³ To capture the prepositional and case (K) marking properties of the head, Adger uses the Hebrew letter $\bar{p}$, which looks like a P and sounds like a K.

Event Identification corresponding to the projection of functional category labels above a single root, or (II) a combination of Functional Application and Predicate Modification corresponding to a dual rooted structure involving a light root and a heavy root. Thus, under (I) and (II), he derives “edge of the table,” as follows (cf. p. 78):

It is important to note here that the derivation for N, rooted by $\sqrt{\text{edge}}$ and introduced via (II), does not involve the derivation for $\bar{p}$, which is rooted by $\sqrt{\text{PART}}$ and introduced via (I). Adger argues that these separate modes of argument introduction capture the relevant optionality: because (II) involves a syntactic derivation that is “at the whim of the speaker,” it is predicted to be an option in the derivation of a nominal extended projection.

One remarkable result of Adger’s analysis of relational nominals is that N is a specifier. To enhance this further, he proposes that N can contain a fair amount of nominal material in the independent specifier position of $\bar{p}$, to the exclusion of PP, and argues that the relational nominal projects sufficient structure to allow *Merge* of intersective APs, numerals, cardinal quantifiers, and some markers of definiteness before it is Merged with $\bar{p}$. So, for example, he derives “three short edges of the table” as follows (cf. p. 6):⁴

⁴ In a particular language, allowable sequences of universal orders of a root are specified in Λ under UEPs. Hence, in what follows, we ignore crosslinguistic differences in the order of Ns and simplify the notation of PP for the sake of convenience.

Note that this crucially differs from a structure that is more similar to the standard view:

- (15) [_{NumP} three [_{AP} short [_{NP} edges [_{PP} of the table]]]]

Unlike in this standard analysis, Adger's analysis of relational nominals can make a straightforward prediction with respect to the interaction of the syntax of APs, PPs, and Ns. In his system, since N can be in the specifier position of $\bar{P}$ together with modifiers, it is predicted that Ns and APs make constituents, crucially excluding PPs, as in [[N AP] ... PP] (where the order of N and AP is irrelevant). He then argues that the prediction is supported by the following generalization ("PP Peripherarity") that, according to him, holds in head-initial languages such as Hawaiian, Romance, Semitic, and Celtic (p. 97):⁵

- (16) When N precedes AP(s) and PP(s) in the extended noun phrase, the order is N>AP>PP.

The relevant contrast is the following (p. 105):

- (17) *Gaelic, a Celtic language*

- a. an dealbh mòr brèagha aig Màiri
the picture big beautiful at Mairi
'Màiri's big beautiful picture'
- b. *an dealbh aig Màiri mòr brèagha
the picture at Mairi big beautiful
'Màiri's big beautiful picture'

Adger points out that the order in (17a) is entirely unexplained in the standard account assuming a structure where an N and a PP make a constituent as in [[AP [N PP]]] (see (15)). Specifically, he notes that to derive the relevant order under the standard approach, we must appeal to an unmotivated movement of PP or an unacceptable remnant roll-up movement of N (see the consequence [3] above) in a way that simply recapitulates the empirical facts. In contrast, however, Adger demonstrates that the relevant order can easily follow from his system given that an N and an AP make a constituent, excluding PPs (see (14)). He suggests that the example in (18) below can be good evidence that N and AP make a constituent, where *na dealbhan snog* "the pictures nice" is replaced by *iad* "those" (pp. 106–107):

⁵ Adger introduces PP Peripherality as a crosslinguistic generalization, but he does not discuss the relevant order in head-final languages very much, so we will not take it up here.

- (18) a. na dealbhan snog ud
 the pictures nice that
 ‘those nice pictures’
 b. iad sin
 they that
 ‘those (guys)’

In his book, Adger carefully examines the interaction of binding and linear order in Gaelic, defending his analysis of PP Peripherarity further.

4. Conclusion with a Small Discussion

It was the labeling problem with Chomsky (2008) that drove Adger to write this book. Adger solves the problem by developing an entirely new system in which such a problem does not arise. In the process of the development, a lot of new machineries are introduced (RLex, CLex, Self Merge, UEPs, LTFs, Λ , etc.). Each of them seems to be motivated and to some extent fairly reasonable in that they are given as products of the independently motivated architecture of the grammar and the familiar methodological “Occam’s razor” sort of considerations that relate to theoretical simplicity. For example, notions like RLex and CLex are forced in any system that takes roots to be categoryless, such as Distributed Morphology (p. 170, footnote 10), and Self Merge can be taken as a fundamental operation that comes for “free” by removing a stipulation in the standard version of *Merge* (pp. 18–19), and what UEPs, LTEs, and Λ do are, in fact, necessary for all systems to specify the category of a root (p. 29) and the embedding relation between one functional category and another (p. 19) (cf. the cartographic approach to syntactic structure advocated by Rizzi (1997)). But one concern, among others, is how his system can systematically predict pronunciation sites. In Chomsky’s system, pronunciation sites correspond to complements of phases in general (VP for v^* and TP for C), but it seems unclear how Adger’s system determines them beyond simple stipulation (see the discussion around pronunciation of morphemes in Section 2). Moreover in his system, it is unclear when and how roots can participate in actual pronunciation. He states that roots like $\sqrt{\text{BE}}$ and $\sqrt{\text{will}}$ enter into pronunciation, but roots like $\sqrt{\text{PART}}$ and $\sqrt{\text{REP}}$ do not. Clearly, the basic problem is why that must be so. But unfortunately, Adger does not respond to this question in the book. In this regard, the proposed system seems to be somewhat weak in the treatment of pronunciation, or more specifically, in the treatment of the *Spell-Out/Transfer* operation, a necessary operation for all systems to

interact with the interfaces.⁶

Hence I would like to suggest the possibility that the labeling problem may be solved even in Chomsky's (2008) system if we consider the optimal application of *Spell-Out/Transfer*. Consider the following situation which Adger regards as "the most recalcitrant situation: the specifier of v^* ":⁷

The labeling problem arises when the subject DP and the phase v^*P are Merged (Chomsky (2008: 160, footnote 34)). Chomsky himself does not notice, but it seems likely that we can solve this problem once we realize that *Spell-Out/Transfer* has already applied to the complement of v^* , i.e. VP, when the DP enters the derivation, as in (19). In fact, given that *Spell-Out/Transfer* has the effect of reducing at least one of the XPs to a simplex syntactic object (see Goto (2013), Nakamura (2012), and Narita (to appear)), it can be assumed that the structure in (19) will be as follows after VP-*Spell-Out/Transfer*:

⁶ Since it is not his main goal of the book to solve pronunciation problems, I am not going to fault his system for this alone. However, we must be open about this weakness, because this will be helpful to readers wishing to develop his system further. Sometimes a problem can be a blessing in disguise.

⁷ In addition to "the most recalcitrant situation," Adger claims that the labeling problem will also arise in Chomsky's system (i) when two LIs [lexical items] are merged at the first step of the derivation, and (ii) when an LI is internally merged with some higher projection in the course of the derivation (p. 11). However, these two cases would not necessarily be problematic under Chomsky's system. With respect to (i), if the Marantz-Borer conception (see Marantz (2006) and Borer (2005) among others) is adopted, as Adger indeed does in his book, then the first relevant constructions will be of the form of *f-root*, where *f* is one of the functional elements determining category and *root* does not qualify as a label (cf. Chomsky (1970)), so that these constructions will be labeled *f*, as intended, because no other element is visible to the labeling algorithm (see Chomsky (2013: 47)). On the other hand, with respect to (ii), given that labeling of derived XP-YP structures can be achieved by a device such as *sharing of most prominent features* of XP-YP known as *Agree*, or more specifically, known as "SPEC-Head agreement" (see Chomsky (2013: 45)), the derived structures will also be labeled as intended: searching for the relevant XP-YP structures, the labeling algorithm finds the same most prominent element in both terms, taking that to be the label of the structures. The labeling of the relevant structures can probably also be achieved by another device such as *Spell-Out/Transfer* of the *interior* of a phase, part of the definition of the so-called *Phase-Impenetrability Condition* (see the suggested approach to labeling below).

This indicates that by *VP-Spell-Out/Transfer* only the DP and the v^* phase head are in the same derivational space. Therefore, considering this crucial step, the labeling problem can be solved even in Chomsky's system: since the residual structure is an instance of H-XP structures, the labeling algorithm can unambiguously select v^* as the label of the syntactic object by minimal head-detection. In this manner, there is still the possibility that we can overcome the recalcitrant situation in Chomsky's system, without immediate recourse to the new mechanisms as Adger.⁸

That said, Adger's technical innovations and empirical findings are highly appreciated. They offer not only new perspectives but also interesting challenges to the standard system. If we can construct a theory of LTFs that can determine pronunciation sites in predicted ways, with a possible expansion of further empirical considerations, then the day may come when his system will be standard.

REFERENCES

- Borer, Hagit (2005) *In Name Only: Structuring Sense*, Oxford University Press, Oxford.
- Brody, Michael (2000) "Mirror Theory: Syntactic Representation in Perfect Syntax," *Linguistic Inquiry* 31, 29–56.
- Chomsky, Noam (1970) "Remarks on Nominalization," *Readings in English Transformational Grammar*, ed. by Roderick A. Jacobs and Peter S. Rosenbaum, 184–221, Ginn, Waltham, MA.
- Chomsky, Noam (2008) "On Phases," *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*, ed. by Robert Freidin, Carlos Otero and

⁸ Due to limitations of space, a detailed discussion of the *Spell-Out/Transfer* approach to labeling is not possible here. For empirical support for this position, see Goto (2013), Nakamura (2012) and Narita (to appear). An interesting research topic would be to investigate how the *Spell-Out/Transfer* approach can deal with all of the empirical facts discussed in Adger's book. We leave this inquiry for future research. Incidentally, it is important to note that if the *Spell-Out/Transfer* approach is adopted, Adger's worries about Chomsky's system that "requires assumptions about the invisibility of trace to labeling and the simultaneity of operations in a phase" (his footnote 3; pp. 169–170) disappear as the label of the relevant structure is determined as soon as *Spell-Out/Transfer* applies, with no trace and simultaneous operations. For relevant discussion, see Narita (2012).

- Maria Luisa Zubizarreta, 133–166, MIT Press, Cambridge, MA.
- Chomsky, Noam (2013) “Problems of Projection,” *Lingua* 130, 33–49.
- Goto, Nobu (2013) “Labeling and Scrambling in Japanese,” *Tohoku: Essays and Studies in English Language and Literature* 46, 39–73.
- Jackendoff, Ray (1977) *X-Bar-Syntax: A Study of Phrase Structure*, MIT Press, Cambridge, MA.
- Kayne, Richard (1994) *The Antisymmetry of Syntax*, MIT Press, Cambridge, MA.
- Kratzer, Angelika (1996) “Severing the External Argument from Its Verb,” *Phrase Structure and the Lexicon*, ed. by Johan Rooryck and Laurie Zaring, 109–138, Kluwer, Dordrecht.
- Marantz, Alec (2006) “Phases and Words,” ms., New York University. <https://files.nyu.edu/ma988/public/Phase_in_Words_Final.pdf>
- Moro, Andrea (2000) *Dynamic Antisymmetry*, MIT Press, Cambridge, MA.
- Nakamura, Taichi (2012) “A Short Note on Multiple Spell-Out,” ms., Tohoku University.
- Narita, Hiroki (2012) “Remarks on the Nature of Headedness and Compositionality in Bare Phrase Structure,” *Proceedings of Sophia University Linguistic Society* 26, 81–126.
- Narita, Hiroki (to appear) *Endocentric Structuring of Projection-Free Syntax: Phasing in Full Interpretation*, John Benjamins, Amsterdam.
- Rizzi, Luigi (1997) “The Fine Structure of the Left Periphery,” *Elements of Grammar: Handbook of Generative Syntax*, ed. by Liliane Haegeman, 281–337, Kluwer, Dordrecht.
- Starke, Michal (2001) *Move Dissolves into Merge*, Doctoral dissertation, University of Geneva.

[received December 20, 2013, revised and accepted April 22, 2014]

Department of Law
Toyo University
Bunkyo-ku, Hakusan
Tokyo 112–8606
e-mail: ngoto@toyo.jp