

The Derivation of Japanese Lexical V-V Compounds and the Interpretation of V1 + *Komu*: An Asymmetry Theory Approach

Tatsuhiko Okubo

This study deals with the following Japanese Lexical V-V Compounds (henceforth, JLVVCs) and explicates how such compounds are derived under Asymmetry Theory, a theory proposed in Di Sciullo (2005):

- (1) asobi-kurasu ‘play-live’, osi-akeru ‘push-open’, oi-harau ‘chase-banish’, simai-komu ‘put-into’, mi-otosu ‘see-drop’, naki-sakebu ‘cry-shout’

Semantic aspects of JLVVCs have traditionally been focused by a lot of researchers (Kageyama (1993), Yumoto (2005), among others) from the standpoint of Lexical Conceptual Structure (LCS). However, there has been no much study for this phenomenon from a (generative) syntactic point of view. This study aims to focus on syntactic aspects of JLVVCs and makes the derivation of such compounds obvious. This enables us to get a theoretical bonus that the LCS approach cannot: the LCS approach requires a redundant theoretical principle to relate the first verb (V1) with the second verb (V2), while my approach needs not such principle to do the same task.

Before getting into main arguments, let us start by observing basic facts on JLVVCs; namely, undivisibility of JLVVCs. The examples in (2) indicate that V1s of JLVVCs are not passivised.

- (2) *kakare-komu ‘written-into’ (cf. kaki-komu ‘write-into’), *osare-aku ‘pushed-open’ (cf. osi-akeru ‘push-open’)

(Kageyama (1993: 87))

This means that in forming JLVVCs, any syntactic operations cannot be related with the compounds provided that passivization is one of the syntactic operations. Although JLVVCs cannot undergo any syntactic operations, this does not mean that we cannot derive them syntactically, or rather once an element becomes a word any syntactic operations cannot be carried out.

As noted earlier, this study employs Asymmetry Theory (AT) established in Di Sciullo (2005), which is based on a minimalist version of generative grammar (Chomsky (2000, 2001)). In this theory, word formation mainly occurs only in a morphological component. There are three operations in AT that play an important role in word formation: M-Shift, Agree, and M-Flip. First, let us explain how

M-Shift functions to derive a word *writer*. M-Shift differs Merge defined in Chomsky (2000, 2001) in that the former combines a tree as one unit, a minimal tree as in (3), with a comp position of the other tree:

- (3) [Spec [Head Comp]]
 (4) a. [Spec [writ- Comp]]
 b. [Spec [-er Comp]]
 c. [Spec [-er [Spec [writ- Comp]]]]

M-Shift serves to combine (4a) with (4b) to produce (4c). At that time, (4a) is attached to the comp position underlined in (4b). Due to this operation, because (4a) and (4b) become one unit, any other elements cannot occur between Tree₁ and Tree₂ (cf. *writ-novel-er, *writ-book-er, etc.).

Next, I introduce Agree. This operation is not so radically different from a usual Agree operation defined in Chomsky (2000, 2001). Thus, I employ the latter familiar one. Then, let us illustrate how this operation works in order to derive *writer*:

- (5) a. [Spec [writ-_[+A, -Pred] Comp]]
 b. [Spec [-er_[-A, +Pred] Comp]]
 c. [Spec [-er_[-A, +Pred] [Spec [writ-_[+A, -Pred] Comp]]]]

There are two important points that must be noted here. One is that each lexical item must have a set of features that identifies the status of the item. The other is that in AT, in which affixes are assumed to be functional categories, lexical categories must not select functional categories, so that in (5), *writ-* cannot select *-er* as its comp position. Di Sciullo (2005a: 26) calls this “the hierarchy of homogeneous projection hypothesis”. According to this hypothesis, in (5), the two items must enter into an Agree relation, driven by an uninterpretable feature [-A] of *-er*. Each uninterpretable feature must be valued and deleted via the operation.

Last but not least, M-Flip is defined as the operation that makes mirror structures to create correct linear orders of words as the following data:

- (6) a. [Spec [-er [Spec [writ- Comp]]]]
 b. [[[[Comp writ-] Spec] -er] Spec]

Moreover, I employ one more important principle not used in AT but traditionally used in accounts of JLVVCs; Transitivity Harmony Principle

(Kageyama (1993)). According to this principle, generally, verbs with external arguments can be combined each other and ones that do not have such arguments each other. So, for instance, transitive verbs can be combined with unergative verbs or transitive verbs but not with unaccusative verbs.

Using all of these operations and a principle, we can explain how JLVVCs are derived. There are five types of JLVVCs in principle as the following:

- (7)
- a. [Spec [V2_{transitive}[-A, +Pred] [Spec [V1_{transitive}[+A, -Pred] Comp]]]]
 - b. [Spec [V2_{transitive}[-A, +Pred] [Spec [V1_{unergative}[+A, -Pred] Comp]]]]
 - c. [Spec [V2_{unergative}[-A, +Pred] [Spec [V1_{unergative}[+A, -Pred] Comp]]]]
 - d. [Spec [V2_{unergative}[-A, +Pred] [Spec [V1_{transitive}[+A, -Pred] Comp]]]]
 - e. [Spec [V2_{unaccusative}[-A, +Pred] [Spec [V1_{unaccusative}[+A, -Pred] Comp]]]]

For example, *nage-taosu* ‘throw-push down’, which is formed according to the pattern in (7a), is derived as follows:

- (8)
- a. [[-A] [taosu_[-A, +Pred] [[+A] [nageru_[+A, -Pred] [+A]]]]] by M-Shift
 - b. [[=A] [taosu_[=A, +Pred] [[+A] [nageru_[+A, -Pred] [+A]]]]] by Agree
 - c. [[[[nage-]] taosu]] by M-Flip

First, M-Shift functions in order to combine the two words *nageru* and *taosu* as in (8a). Second, Agree values the uninterpretable features of the complex word and deletes such features. Here, based on Di Sciullo’s analysis, I introduce a new feature [$\pm A$] that means that if [+A] any lexical item can be inserted into the position. Note that a spec position is a position for an external argument and a comp position is a position for an internal argument position. The [-A] feature in *taosu* enters into an Agree relation with the [+A] feature in the spec position of *nageru*. After M-Shift and Agree are accomplished, M-Flip applies in order to make a mirror image of the derivation in (8c).

Next, using *asobi-kurasu* ‘play-live’ as an example, let us explain how pattern (7c) is derived in this system as the following:

- (9)
- a. [[-A] [kurasu_[-A, +Pred] [[+A] [asobu_[+A, -Pred] [-A]]]]] by M-Shift
 - b. [[=A] [kurasu_[=A, +Pred] [[+A] [asobu_[+A, -Pred] [-A]]]]] by Agree
 - c. [[[[asobi-]] kurasu]] by M-Flip

The derivation in (9) explained in a similar way. First M-Shift applies and next, Agree occurs. At this stage, there is a difference between (9b) and (8b). In (8b), V1

is a transitive verb, so there are two [+A] features. However, in (9b), V1 is an unergative verb with only one [+A] feature in its spec position. This verb's comp position has a [-A] feature to be deleted. However, this feature cannot be deleted because there is no driving force for Agree. However, note that what unergative verbs mean is that act of the verbs is done by the Actor and the effect of the act orients to the Actor. This means that an external position is identified with an internal position, so that I assume that if V1 is an unergative verb, there is an Agree relation between an external position and an internal position as in (9b). The derivation in (9c) shows that a correct linear order is created by M-Flip.

The same logic is applied into *ahure-otiru* 'overflow-fall' formed according to the pattern in (7e) as the following:

- (10) a. [[-A] [otiru_[-A, +Pred] [[-A] [ahure_[+A, -Pred] [+A]]]]] by M-Shift
 b. [[=-A] [otiru_[-A, +Pred] [[=-A] [ahure_[+A, -Pred] [+A]]]]] by Agree
 c. [[[[ahure-]] otiru]] by M-Flip

The same problem as in (9b) occurs in this example. Suppose, however, that unaccusative verbs show the event in which Actor is equal to Patient, namely an external argument equals to an internal argument. Then, the same Agree relation performed in (9b) can be established here. Thus, there is no problem in the derivations in (10).

If these arguments that have been carried out are on the right track, it follows that we get a certain theoretical bonus. The LCS approach must require a principle Obligatory Identification of Subjects (Yumoto (2005)); each external argument in V1 and V2 must be linked. However, this study does not need such redundant principle. For example, consider the derivations in (8). In this derivation, the [-A] feature is in the spec position of V2 and the [+A] feature in that of V1. Because of this featural relationship between the two positions, Agree takes place. Moreover, this operation establishes a certain relationship, a subject identification. Given that this Agree operation is also used in syntax, it is an optimal one in the grammar of human language. Contrary to this, Obligatory Identification of Subjects is used only in word formation of JLVVCs. This is very narrowly defined principle as compared with Agree, and such principle must be abolished.

In this study, adopting AT, I have proposed five derivations for JLVVCs. Each derivation was created by very simple operations, namely, M-Shift, Agree, and M-Flip and a principle Transitivity Harmony Principle. Theoretically, the approach in this study is superior to the LCS approach in Yumoto (2005) in that there is no requirement of a redundant principle Obligatory Identification of Subjects.