Case and Word Order Variation*

Myong-Hi Chai & Sae-Youn Cho
(Chosun College of Science & Technology) (Honam University)

Chai, Myong-Hi & Cho, Sae-Youn. 2004. Case and Word Order Variation. *The Linguistic Association of Korea Journal, 12*(3), 81-103. The purpose of this study is to provide a formal analysis of word order variation (or "freezing" phenomena) in a free word order language like Korean under the Domain Theory. Though Korean is known as a relatively free word order language, the canonical word order based on obliqueness is sometimes fixed under some conditions. To identify those conditions, we claim that Case consists of Potential Case and Realized Case, and that both kinds of Case, together with the information on Animacy of the NP, play a role in restricting the word order variation in the language. The two generalizations captured from various scrambling data are represented into two LP constraints, i.e. the PC Constraint (LP2) and the RC and ANI(MACY) Constraint (LP3). Adopting the Domain Theory proposed by Reape (1994, 1996), we define word order in terms of "word order domains".

Key Words: word order variation, freezing, potential vs. realized case, animacy, Domain Theory, word order domain, LP constraint

1. Introduction

It is well known that the Korean language has various case morphemes and is a relatively free word order language. It is also believed that there might be some relationships between the case morphemes and the word order variation. Generally, in Korean, arguments are easily scrambled in a sentence as long as they precede their head.

However, the freedom of word order among arguments can be

* Earlier versions of this study were presented at the WECOL 2000 conference and Kyung Hee Workshop on Case in May 2004. We appreciate the participants for their comments, and of course all the remaining errors are of our own. We also appreciate the two anonymous reviewers of this journal for their comments.

restricted when they carry the same case morpheme. To account for such restriction on scrambling possibilities in Japanese and Korean, Kuno (1980) and Chung (1998) have proposed the so-called Case-Over-Case constraint (COC), which says that the word order of two NPs with the same case is difficult to switch. As illustrated in (1), the COC enables us to correctly predict that the two NPs, *Mary-ka* and *kyoswu-ka*, cannot be scrambled because they share the same case.[1)]

(1) a. Mary-ka kyoswu-ka toy-ess-ta.
Mary-Nom professor-Nom become-Pst-Dec
b.*Kyoswu-ka Mary-ka toy-ess-ta.
professor-Nom Mary-Nom become-Pst-Dec
'Mary became a professor.'

However, the previous analyses based on surface case alone cannot explain a difference in scrambling possibilities between examples in (1) and (2), both of which are double nominative constructions and seemingly look alike.

(2) a. Mary-ka ton-i manh-ta.
Mary-Nom money-Nom a.lot.of-Dec
b. Ton-i Mary-ka manh-ta.
money-Nom Mary-Nom a.lot.of-Dec
'Mary has a lot of money.'

The two NPs, *Mary-* and *ton-* 'money', with same case, i.e. Nom, must not be scrambled under the previous analyses, but they CAN be scrambled in (2b), unlike those in (1b). This points to the fact that the previous analyses based on the COC are not sufficient to explain the scrambling phenomena in Korean.

In order to appropriately account for the scrambling phenomena, a

1) The abbreviations used in this paper are as follows: Nom for Nominative, Dat for Dative, Acc for Accusative, Gen for Genitive, Pres for Present Tense, Pst for Past Tense, and Dec for Declarative Mood.

theory must answer what constraints restrict the possibility of scrambling among arguments. To answer this question, we claim that, unlike the previous analyses, not only the surface case but also all the possible cases of an NP should be considered. We also claim that the information on "animacy" of NPs plays an important role.

We clarify our proposal through empirical data in section 2. The proposed claim is interpreted in the framework of Head-Driven Phrase Structure Grammar (HPSG) in section 3. In section 4, we show that the newly proposed theory can provide a simpler analysis for various scrambled sentences in Korean. In section 5, the theoretical implications of this proposal will be discussed.

2. Restrictions on scrambling: A Proposal

2.1. Potential cases

In order to find out the constraints which restrict scrambling possibilities among arguments, we need to consider the sentences in (1) and (2) again, which are represented in (3) and (4) respectively.

(3) a. Mary-**ka/*eykey** kyoswu-**ka** toy-ess-ta.
Mary-**Nom/*Dat** professor-**Nom** become-Pst-Dec
b.***Kyoswu-ka** **Mary-ka** toy-ess-ta.
professor-Nom Mary-Nom become-Pst-Dec
'Mary became a professor.'

(4) a. Mary-**ka/eykey** ton-i manh-ta.
Mary-**Nom/Dat** money-**Nom** a.lot.of-Dec
b. **Ton-i** **Mary-ka/eykey** manh-ta.
money-Nom Mary-Nom/Dat a.lot.of-Dec
'Mary has a lot of money.'

As illustrated in (3) and (4), the first NP *Mary-* in (3a) does not alternate its case, and cannot be switched with the second NP *kyoswu-ka*. On the other hand, *Mary-*, the first NP in (4), alternates its

case between Nom and Dat, and CAN be switched with the second NP *ton-i*, no matter what case the first NP is realized with. This observation enables us to assume that the value of case that an NP may possibly have in a given sentence may differentiate the possibilities of scrambling among NPs.

If it is true that the difference in case alternation makes a difference in scrambling possibilities, we need to consider such case alternation in the analysis of scrambling. We assume that an NP can have a set of possible cases with regard to its head but can carry only one (or null) case morpheme out of the set when realized in syntax.

According to this assumption, we propose a new concept of case, that is, a noun has two kinds of cases, i.e. Potential Case (PC) and Realized Case (RC). We also propose that PC has as its value a set of all the cases that an NP may possibly bear with respect to its head, and that RC has as its value a set of the actually realized case on the nominal. Note that the value of RC may be an empty set or a singleton set, and must be a subset of the PC value, as in (5).[2)]

(5) NP $\left[\text{CASE} \left[\begin{array}{ll} \text{P(OTENTIAL) C(ASE)} & \{..., [1], ...\} \\ \text{R(EALIZED) C(ASE)} & \{ ([1]) \} \end{array}\right]\right]$

Under this proposal, the examples in (3) and (4) can be informally represented as in (3′) and (4′). Since the first NP *Mary-* in (3′), which does not alternate its case, has {Nom} for the PC value, it cannot have {Dat} or any other case set for the RC value by the definition of (5), as in (3′b). Thus, it cannot be switched with the other NP *kyoswu-ka*, which has the same PC value {Nom}, as in (3′c). On the other hand, the first NP *Mary-* in (4′a), which alternates its case, has {Nom, Dat} for the PC value, so it may have {Nom} or {Dat} for the RC value as

2) Note that we do not care whether the PC and the RC are structurally assigned or lexically (or morphologically) assigned. We just care about the result, i.e. what case the NP might carry potentially or apparently. Note all the cases an NP can potentially carry in a certain expression are called "Potential Case", while the case the NP is apparently realized with in the given expression is called "Realized Case".

in (4'a) and (4'b) respectively. Thus, it can be switched with the other NP *ton-i* which has a different PC value, as in (4'c).

(3') a. Mary-**ka** kyoswu-**ka** toy-ess-ta.
M-[RC {Nom}] professor-[RC {Nom}] become-Pst-Dec
[PC {**Nom**}] [PC {**Nom**}]

b. *Mary-**eykey** kyoswu-**ka** toy-ess-ta.
M-[RC {Dat}] professor-[RC {Nom}] become-Pst-Dec

c. ***Kyoswu-ka** **Mary-ka** toy-ess-ta.
professor-[PC {Nom}] M-[PC {Nom}] become-Pst-Dec
'Mary became a professor.'

(4') a. Mary-**ka** ton-**i** manh-ta.
Mary-[**RC {Nom}**] money-[**RC {Nom}**] a.lot.of-Dec
[PC {Nom, Dat}] [PC {Nom}]

b. Mary-**eykey** ton-**i** manh-ta.
Mary-[**RC {Dat}**] money-[**RC {Nom}**] a.lot.of-Dec

c. **Ton-i** **Mary-ka/eykey** manh-ta.
money-[**PC {Nom}**] Mary-[**PC {Nom, Dat}**] a.lot.of-Dec
'Mary has a lot of money.'

With the new concept of potential case given, we can make a generalization on the above data, as in (6). The generalization in (6) enables us to argue that all the possible cases that an NP may have in a given construction be considered to appropriately account for the word order variation among NPs.

(6) **Generalization 1** (Preliminary)
Two NPs cannot be scrambled in a sentence when they have the same Potential Case (PC) value.

2.2. Realized case and animacy

Though the difference in word order variation between (3') and (4') could be accounted for by means of the potential cases of NPs, RC still

seems to play a role in deciding the possibility of scrambling. Though the first NP *Mary-* in (7a) has a different PC value from that of the second NP *chinkwu-* ('friend'), the scrambling possibility of the two NPs varies depending on their RC values, as in (7b-c). However, this result points out the problem of the explanations for (4'), where the two NPs are freely scrambled regardless of their RC values.

(7) a. Mary-ka/eykey chinkwu-ka manh-ta.
Mary-[**PC {Nom, Dat}**] friend-[**PC {Nom}**] a.lot.of-Dec
b. */??Chinkwu-ka Mary-ka manh-ta.
friend-[**RC {Nom}**] Mary[**RC {Nom}**] a.lot.of-Dec
c. (?)Chinkwu-ka Mary-eykey manh-ta.
friend-[**RC {Nom}**] Mary-[**RC {Dat}**] a.lot.of-Dec
'Mary has a lot of friends.'

We can find out what causes the problem in explaining the contrast between (7) and (4'), if we reconsider those examples with regard to the animate status of NPs. That is, in (7'=(7)) two NPs, *Mary-* and *chinkwu-* ('friend'), have the same value for ANI(MACY), i.e. '+', while in (8=(4')) the two NPs, *Mary-* and *ton-* ('money'), have as ANI values '+' and '-' respectively. It follows that this difference in the information on animacy of NPs may differentiate the scrambling possibility of the NPs in cooperation with the RC value.

(7') a. Mary-ka/eykey chinkwu-ka manh-ta.
Mary-[PC {Nom, Dat}] friend-[PC {Nom}] a.lot.of-Dec
[ANI +] **[ANI +]**
b. */??Chinkwu-ka Mary-ka manh-ta.
friend-[RC {Nom}] Mary[RC {Nom}] a.lot.of-Dec
c. (?)Chinkwu-ka Mary-eykey manh-ta.
friend-[RC {Nom}] Mary-[RC {Dat}] a.lot.of-Dec
'Mary has a lot of friends.'

(8) a. Mary-ka/eykey ton-i manh-ta.
Mary-[PC {Nom, Dat}] money-[PC {Nom}] a.lot.of-Dec
[ANI +] **[ANI -]**

b. ton-i **Mary-ka** manh-ta.
money-[RC {Nom}] Mary-[RC {Nom}] a.lot.of-Dec
[ANI -] [ANI +]

c. **Ton-i** **Mary-eykey** manh-ta.
money-[RC {Nom}] Mary-[RC {Dat}] a.lot.of-Dec
[ANI -] [ANI +]

'Mary has a lot of money.'

As noted in (8), the two NPs cannot be freely switched regardless of their RC values, if they have different ANI values. However, as in (7'), if they have the same ANI value, their scrambling possibility varies depending on their RC values.

The above observation enables us to make another generalization about restrictions on scrambling among arguments as in (9).

(9) **Generalization 2** (preliminary)
Two NPs cannot be scrambled when they have the same values for Realized Case (RC) and ANIMACY (ANI).

3. Implementation in HPSG

3.1. P&S (1987): COP

Pollard & Sag (1987) have assumed that for each language there is a Constituent Order Principle (COP) specific to that language. The essence of the COP is simply that the phonology of a phrasal sign is a function of the sign's DAUGHTERS value, as indicated in (10).

(10) **Constituent Order Principle** (Preliminary)

$$\textit{phrasal-sign}\begin{bmatrix} & \end{bmatrix} \Rightarrow \begin{bmatrix} \text{PHON} & \textit{order-constituents} & ([1]) \\ \text{DTRS} & [1] & \end{bmatrix}$$

To precisely state the COP of each language, they propose we need some Linear Precedence (LP) Constraints specific to that language. To

provide an account for the word order of the complements in English, which often occur in a fixed order, they propose an LP, based on their position in the obliqueness hierarchy, roughly formulated as in (11).

(11) COMPLEMENT ≪ COMPLEMENT

The symbol "≪" designates that less oblique elements should precede more oblique elements. However, to account for the word order variation in a free word order language like Korean, it is believed that we need some LP constraints restricting the freedom of word order.

3.2. Reape (1994, 1996): Domain Theory

Following the HPSG tradition in which word order is not defined in terms of the terminal strings of tree configurations, Reape (1994, 1996) has proposed that word order is defined by "word order domains". He claims that only phrasal signs are specified for the DOM(AIN) attribute, and phrasal word orders within their DOMs are composed compositionally from their daughter domains, by means of the operation "domain-union", represented as the big circle "○" over the list. Basically, the daughter domain is domain-unioned into its mother's domain. For example, let A be a list <a, b> and B a list <c, d>, then the two lists "domain-union" each other producing a list C, ○<A, B, C>, which is one of the six sequences in (12).

(12) a. <a, b, c, d> b. <a, c, b, d>
c. <a, c, d, b> d. <c, a, b, d>
e. <c, a, d, b> f. <c, d, a, b>

Note that the unioned list C contains all and only the members of A and B, and the relative order of the elements of A and B should be preserved in C.[3)]

3) On the contrary, a sequence may not free its members in the mother's DOM. Imagine A is the case, then C may be one of the three sequences in (ia-c), out of six. It follows

Meanwhile, the COP, a universal principle with regard to word order, proposed in (11) by P&S (1987) is revised so that the PHON value of a phrasal sign is required to be the concatenation of the PHON values of the elements of the DOM sequence, instead of those of DTRS, if it is specified. Note that the small circle " ∘ " is the list concatenation.

(13) **Constituent Order Principle** (Revised)

$$\textit{phrasal-sign}\begin{bmatrix} \end{bmatrix} \Rightarrow \begin{bmatrix} \text{PHON} & [1] \circ \ \ldots \ \circ [N] \\ \text{DOM} & \langle [\text{PHON } [1]], \ \ldots, \ [\text{PHON } [N]] \rangle \end{bmatrix}$$

3.3. LP constraints in Korean

With the theoretical tools such as the COP and the Domain Theory given, we can now go back to the discussion on restricting the freedom of word order in Korean. In section 2, we have observed that though in Korean, arguments are relatively freely scrambled in a sentence, they often occur in a fixed order when they meet a certain condition. We also have observed that the conditions are related not only to the case an NP actually carries but also to the case the NP may possibly have with regard to its head sister. Thus, we have proposed that the case value consists of two attributes, i.e. PC and RC. Those conditions have been captured as two generalizations. The Generalization 1 says that two NPs cannot be scrambled when they have the same PC value, and the Generalization 2 indicates that two NPs cannot be scrambled when they share the RC and ANI values.

---

that the sequences in which an element of B is interleaved into the list A, are impossible.

(i) a. <<a, b>, c, d> b. <c, <a, b>, d> c. <c, d, <a, b>>
d. *<<a, c, b>, d> e. *<<c, a, b>, d> f. *<c, <a, d, b>>

To specify whether or not a daughter domain can be unioned into the mother's domain, he proposes the UNIONED feature, which has a boolean value. The value of UNIONED can be determined by lexical functors or language specific principles. It follows that in (i) the list A has the [UNIONED -] specification by an appropriate method, and the bracket must not be open. Though this feature is not exploited in this study, we are sure it is needed to explain further scrambling data.

To account for the word order "freezing" in Korean under the Domain Theory, the two generalizations need to be slightly revised, as in (14).

(14) a. **Generalization 1** (Revised)
Two NPs cannot be scrambled **in the Domain** when they have the same Potential Case (PC) value.
b. **Generalization 2** (Revised)
Two NPs cannot be scrambled **in the Domain** when they have the same values for Realized Case (RC) and ANIMACY (ANI).

These two revised generalizations imply that the word order under our approach is dealt with on the Domain level, not on the PHON level.

We assume that in a canonical order, NPs are ordered depending on their position in the obliqueness hierarchy.[4] Nevertheless, they can be freely scrambled, as long as they precede their head and meet the conditions stated in the two generalizations. Those conditions are rendered to LP constraints, which specify when the word order should be fixed, as illustrated in (15)-(17).

(15) **LP 1 (Head-Final Constraint)**
[DOM] ⇒ [DOM [ ] < head]

(16) **LP 2 (PC Constraint)**
[DOM] ⇒ [DOM NP_1[PC α] ≪ NP_2[PC α]]

(17) **LP 3 (RC & ANI Constraint)** (Preliminary)

$$[\text{DOM }] \Rightarrow \left[\text{DOM } NP_1 \left[\begin{matrix}\text{RC} & \beta \\ \text{ANI} & [1]\end{matrix}\right] \ll NP_2 \left[\begin{matrix}\text{RC} & \beta \\ \text{ANI} & [1]\end{matrix}\right]\right]$$

Note, under the Domain Theory, all the statements on linear precedence constraints are made within the DOM value. The symbol "<" indicates that the left element precedes the right one in a domain. LP1 says that the head always comes after its complements, specifiers,

4) Here we adopt the version of traditional obliqueness hierarchy, following Pollard and Sag (1994:24). Therefore, subjects come first, and other complements appear in the order primary object, secondary object, then oblique arguments.

or subjects. The usage of the symbol "≪" indicates that NPs are arranged in accordance with the obliqueness hierarchy if they satisfy the constraints in (16) and (17). In other words, if two NPs have the same PC value (α in the above) or the same RC and ANI values (β and [1] in the above), they cannot be scrambled but occur in a fixed order according to their position in the obliqueness hierarchy. It follows that a less oblique NP precedes a more oblique NP when they meet the conditions stated in LP 2 and 3 respectively. Note that once the word order of two NPs is fixed in a certain domain in terms of any of the above three LP Constraints, the fixed order must be preserved in the higher domains.

3.4. A Case-Driven Analysis for basic data

To show how these case-related LP constraints and Domain Theory interact to account for the relation between case and the scrambling possibilities among NPs, we examine the examples of *toy-ta* ('become') and *manh-ta* ('a lot of') again.[5)]

(18) a. [$_S$ Mary-**ka** [$_{VP}$ kyoswu-**ka** toy-ess-ta]].
M-[PC {Nom}] professor-[PC {Nom}] become-Pst-Dec
[RC {Nom}] [RC {Nom}]
[ANI +] [ANI +]

b. ***Kyoswu-ka** **Mary-ka** toy-ess-ta.

c. DOM(VP) = <kyoswu-ka, toy-ess-ta>

d. DOM(S) = <Mary-ka, kyoswu-ka, toy-ess-ta>

'Mary became a professor.'

(19) a. [$_{S2}$ Mary-**ka/eykey** [$_{S1}$ ton-i manh-ta]].
Mary-[PC {Nom, Dat}] money-[PC {Nom}] a.lot.of-Dec
[ANI +] **[ANI -]**

---

5) The examples in (18)-(20), especially those in (19) and (20), might be given different syntactic structures depending on the theories. Note, however, under the Domain Theory word order is dealt with on the DOM list, with distinguishing the tectogrammatical structrure from the phenogrammatical structure. Therefore, the syntactic structure itself might not be that important with regard to word order.

b. **ton-i** **Mary-ka/eykey** manh-ta.
[RC {Nom}] [RC {Nom}/{Dat}] a.lot.of-Dec
c. DOM(S_1) = <ton-i, manh-ta>
d. DOM(S_2) = <Mary-ka/eykey, ton-i, manh-ta> ∨
<ton-i, Mary-ka/eykey, manh-ta>
'Mary has a lot of money.'

(20) a. [$_{S2}$ Mary-**ka/eykey** [$_{S1}$ chinkwu-**ka** manh-ta]].
Mary-[PC {Nom, Dat}] friend-[PC {Nom}] a.lot.of-Dec
[**ANI** +] [**ANI** +]
b. */??Chinkwu-ka Mary-ka manh-ta.
friend-[**RC {Nom}**] Mary-[**RC {Nom}**] a.lot.of-Dec
c. (?)Chinkwu-ka Mary-eykey manh-ta.
friend-[**RC {Nom}**] Mary-[**RC {Dat}**] a.lot.of-Dec
d. DOM(S_1) = <chinkwu-ka, manh-ta>
e. DOM(S_2) = <Mary-ka/eykey, chinkwu-ka, manh-ta> ∨
<chinkwu-**ka**, Mary-**eykey**, manh-ta>
'Mary has a lot of friends.'

In (18), DOM(VP) consists of two elements, *kyoswu-ka* and *toy-ess-ta*, as in (18c), which are in a head-complement relation. Thus, the argument *kyoswu-ka* precedes the head *toy-ess-ta* according to the Head-Final Constraint (LP1). The higher domain, DOM(S), adds one element *Mary-ka* to DOM(VP), as in (18d). Since the first NP *Mary-ka*, which does not alternate its case, shares the PC, RC and ANI values with the second NP *kyoswu-ka*, they cannot be scrambled but keep the order of the obliqueness hierarchy, satisfying both the PC Constraint (LP2) and the RC & ANI Constraint (LP3). Thus, the scrambled example (18b) is ungrammatical.

In DOM(S_2) of (19), the first NP *Mary-*, alternating its case, does not share the values for PC and ANI with the second NP *ton-i*. Thus, they are not constrained by LP2 or LP3 but scrambled as in (19b).

In DOM(S_2) of (20), the first NP *Mary-*, alternating its case, has a different PC value from that of the second NP *chinkwu-ka*, and thus there is no way for LP2 to restrict the two NPs from scrambling each

other. However, since the two NPs share the ANI value, unlike the two NPs in (19), their order is fixed when they are realized with the same case as in (20b), according to LP3. That is, the less oblique NP *Mary-ka* must precede the more oblique NP *chinkwu-ka* in DOM(S_2). However, when they are realized with different RC values, they are not restricted by the application of LP3. Thus, the two NPs may be scrambled as in (20c) and (20e).[6] Note that the Head-Final Constraint holds in all the domains.

4. Further data analysis

In this section, we will show that our analysis provides a simpler explanation for the scrambling and "freezing" facts in various constructions.[7]

4.1. Double accusative construction

First, we start with the so-called "double accusative" construction, in which there are two NPs marked with the same accusative case.

(21) a. [$_S$ John-i [$_{VP}$ **Mary-lul/eykey** **ton-ul** cwu-ess-ta.]]
J-[PC{Nom}] M-[PC{Acc, Dat}] money-[PC{Acc}] gave
[RC {Nom}] [RC {Dat}/**{Acc}**] [RC **{Acc}**]
[ANI +] [ANI +] [ANI -]
b. John-i **ton-ul** **Mary-lul/eykey** cwu-ess-ta.
c. DOM(VP) = <Mary-lul/eykey, ton-ul, cwu-ess-ta> ∨
<ton-ul, Mary-lul/eykey, cwu-ess-ta>

---

6) Some native speakers of Korean feel that though the scrambled sentence (21c) is much better than (21b), it is not as good as (21a). That may also be explained in our theory. That is, the two NPs satisfy only a half of the condition indicated in LP3, and thus the grammaticality is in-between.

7) Note the term "freezing", (originally from Mohanan 1992), is used here as the opposite concept to scrambling or freedom of word order. That is, by the "freezing" facts, we refer to some constructions in which two NPs cannot be scrambled but are "frozen" as their canonical word order.

d. DOM(S) = <John-i, Mary-lul/eykey, ton-ul, cwu-ess-ta> ∨
<John-i, ton-ul, Mary-lul/eykey, cwu-ess-ta> ∨
<Mary-lul/eykey, John-i, ton-ul, cwu-ess-ta> ∨ ...
'John gave money to Mary.'

As illustrated in (21), since *Mary-*, the second argument of *cwu-* ('give'), can alternate its case between Acc and Dat, it has as PC value the set {Dat, Acc}, which is different from that of the third argument *ton-* ('money'), i.e. {ACC}. Therefore, the order of the two NPs is not constrained by LP2. Also, since the two NPs have a different ANI value, they are not constrained by LP3 regardless of their RC values. Therefore, the two NPs are freely scrambled in DOM(VP) and DOM(S) as in (21c-d) respectively. Note that the first NP *John-*, which has different PC and RC values from those of other NPs, is also freely scrambled. Thus, the number of the possible DOM(S)s is as many as 3!, i.e. six.[8)]

4.2. *Seltukha-* ('persuade') construction

The verb *seltukha-* ('persuade') requires two NPs and one VP (VP_1 below) to complete a sentence. In (22), *seltukha-yss-ta* takes *John-i* for its subject and *Mary-eykey/ul* and *Tom-eykey/ul ton-ul cwu-lako* for its two complements. Inside the VP, the head verb *cwu-* ('give') subcategorizes for *Tom-eykey/lul* and *ton-ul* as its complements.

---

8) We could think of another "double accusative" construction as in (i). In (i) the two NPs *kaci-* ('branch') and *namwu-* ('tree') cannot be scrambled, even when they have different RC values. This word order freezing is easily explained. That is, since *namwu-* is the specifier of the other NP *kaci-*, they are subject to LP1, i.e. Head-Final Constraint. Thus, *namwu-* must precede its head *kaci-*.

(i) a. John-i [$_{NP}$ Namwu-**lul/uy** kaci-**lul**] cal-lass-ta.
tree-[PC {Acc, Gen}] branch-[PC {Acc}] cut-Pst-Dec
[RC {Acc}/{Gen}] [RC {Acc}]
[ANI -] [ANI -]
b. *John-i [$_{NP}$ kaci-**lul** Namwu-**lul/uy**] cal-lass-ta.
'John cut the branch of the tree.'

(22) a. [$_S$ John-i [$_{VP2}$ **Mary-eykey/lul** [$_{VP1}$ **Tom-eykey/ul** ton-ul
J-[PC{Nom}] M-**[PC{Dat,Acc}]** **T-[PC{Dat,Acc}]** money-[PC {Acc}]
[RC {Nom}] **[RC {Dat}/{Acc}]** **[RC {Dat}/{Acc}]** [RC {Acc}]
[ANI +] **[ANI +]** **[ANI +]** [ANI -]
cwu-lako] seltukha-yss-ta.]]
give-Comp persuade-Pst-Dec

b. *John-i **Tom-eykey/ul** **Mary-eykey/lul** ton-ul cwu-lako seltukhayssta.
(Interpreted as 'John persuaded Tom to give money to Mary.')

c. DOM(VP_1) = <Tom-eykey/ul, ton-ul, cwu-lako> ∨
<ton-ul, Tom-eykey/ul, cwu-lako>

d. DOM(VP_2) = <**Mary-eykey/lul, Tom-eykey/ul,** ton-ul, cwu-lako> ∨
<**Mary-eykey/lul,** ton-ul, **Tom-eykey/ul,** cwu-lako> ∨ ...

e. DOM(S) = <John-i, **Mary-eykey/lul, Tom-eykey/ul,** ton-ul, cwu-lako>
∨ <John-i, **Mary-eykey/lul,** ton-ul, **Tom-eykey/ul,** cwu-lako> ∨...
'John persuaded Mary to give money to Tom.'

In the above example, both the NP *Mary-*, subcategorized for by the main verb *seltukha-* ('persuade'), and the NP *Tom-*, subcategorized for by the embedded verb *cwu-* ('give'), alternate their case between Dat and Acc, which means they have the same PC value, i.e. {Dat, Acc}. Thus, they cannot be switched in the DOM(VP_2) and DOM(S) in terms of LP2, and it follows that the scrambled example in (22b) is bad. On the other hand, the other NPs *John-* and *ton-* ('money'), which have different PC and ANI values from others, are freely scrambled in the DOM(VP_1), DOM(VP_2), and DOM(S) respectively, because no LP constraints restrict their scrambling.

4.3. Periphrastic causative construction

In this subsection, we will show that the periphrastic causative construction in which complicated case marking is involved is also analyzed in the same way under our approach.[9]

9) For the Korean periphrastic causative construction, Bratt (1996) gives two different

(23) a. [S John-i [VP2 **Mary-ka/eykey/lul** [VP1 **Tom-eykey/ul** ton-ul
[PC {Nom}] **[PC {Nom,Dat,Acc}]** **[PC {Dat,Acc}]** [PC {Acc}]
[RC {Nom}] **[RC {Nom}/{Dat}/{Acc}]** **[RC {Dat}/{Acc}]** [RC {Acc}]
[ANI +] **[ANI +]** **[ANI +]** [ANI -]
cwu-key] sikhi-n-ta]]
give-Comp make-Pres-Dec

b. John-i **Tom-eykey/ul** **Mary-ka** ton-ul cwu-key sikhi-n-ta.
[RC {Dat}/{Acc}] [RC {Nom}]

c.*John-i **Tom-eykey/ul** **Mary-eykey/lul** ton-ul cwu-key sikhi-n-ta.
[RC {Dat}/{Acc}] [RC {Dat}/{Acc}]
(Interpreted as 'John makes Tom give money to Mary.')
'John makes Mary give money to Tom.'

As indicated in (23), in Korean causative construction, the causee NP can alternate its case among Nom, Dat, and Acc, but the scrambling possibilities depend on what case is realized on the causee NP. Since *Mary-* and *Tom-* have different PC values, i.e. {Nom, Dat, Acc} and

---

structures depending on the case the causee NP carries. That is, when the causee is marked with nominative case, the sentence is given a bi-clausal structure, as in (ia), while it has a mono-clausal structure when the causee is marked with dative case or accusative case, as in (ib).

(i) a. [S_2 John-i [S_1 Mary-**ka** Tom-eykey/ul ton-ul cwu-key] sikhinta.]
M-[PC {Nom}] T-[PC {Dat, Acc}]
[RC {Nom}] [RC {Dat}/{Acc}]
[ANI +] [ANI +]

b. [S J-i [VP_2 Mary-**eykey/lul** [VP_1 Tom-eykey/ul ton-ul cwu-key]] sikhinta.]
M-[PC {Dat, Acc}] T-[PC {Dat, Acc}]
[ANI +] [ANI +]

c. John-i **Tom-eykey/ul** **Mary-ka** ton-ul cwu-key sikhinta.

d. *John-i **Tom-eykey/ul** **Mary-eykey/lul** ton-ul cwu-key sikhinta.
(Interpreted as 'John makes Tom give money to Mary.')
'John makes Mary give money to Tom.'

Even if these structures were given to the periphrastic causatives, the examples in (i) can be analyzed in the same way under our approach. That is, in (ia), since the two NPs have different PC and RC values, LP2 and LP3 are not applicable. It follows that the two NPs are freely scrambled as in (ic). However, the example in (ib), where the two NPs share the PC value as well as the RC and ANI values, is constrained by both LP2 and LP3. Thus, the scrambled example (id) is correctly predicted as ungrammatical.

{Dat, Acc} respectively, LP2 is not applicable. Also since the two NPs have the same ANI value (+), they cannot be scrambled when they have the same RC value, by means of LP3, as in (23c), while they can when they have different RC values as in (23b).

However, what we should note in (23c) is that the scrambling possibility does not change if and only if the two NPs, *Mary-* and *Tom-* are realized as any one of the members of the set {Dat, Acc}, which is the common part of the PC values of the two NPs. That is, the two NPs cannot be switched even when they are differently marked with Dat and Acc respectively, or vice versa. To incorporate this, we need to slightly modify the RC and ANI Constraint (LP3) as in (24).

(24) **LP 3 (RC & ANI Constraint)** (Revised)

$$[\text{DOM} \quad] \Rightarrow \text{DOM } NP_1 \text{ PC } \beta \cup \gamma \ll NP_2 \text{ PC } \beta \cup \kappa$$

$$: \text{where } \gamma \neq \left[\kappa, \ \beta \supset \left[\begin{array}{ll} \text{RC} & \beta' \\ \text{ANI} & [1] \\ \beta' \text{ and } \beta'' \subset \beta & \end{array}\right]\right] \qquad \left[\begin{array}{ll} \text{RC} & \beta'' \\ \text{ANI} & [1] \end{array}\right]$$

Note that β′ and β″ are a subset of β, the common set of the PC values of the two NPs, and they may or may not be the same set. The revised LP3 constraint says that the canonical word order of two NPs should be fixed when both the NPs not only have the same ANI value but also have as the RC value the set which consists of one of the members which commonly occur in the PC values of the two NPs. With this revised LP3, we can correctly explain the fact that in (23c) *Tom-* and *Mary-* cannot be scrambled even when they are realized with different RC values if and only if the RC values are out of the set {Dat, Acc}. This revised LP3 is expected to account for other scrambling data as well as the periphrastic causative construction without stipulating anything.

4.4. *Yaksokha-* ('promise') construction

The *yaksokha-* construction, showing a slightly different case marking

behavior from the *sultukha-* ('persuade') construction, can also be provided a similar analysis under our approach. The verb subcategorizes for two NPs and a VP, as illustrated in (25). In this example, the NP *Tom-* subcategorized for by the embedded verb *cwu-* ('give') cannot be switched with the second NP *Mary-* when it is realized as Dative case as in (25b), while it can be freely scrambled with other NPs as well as *Mary-* when it is realized as Accusative case as in (25c).

(25) a. [$_S$ John-i [$_{VP2}$ **Mary-eykey** [$_{VP1}$ **Tom-eykey/ul** ton-ul
[PC {Nom}] **[PC {Dat}]** **[PC {Dat,Acc}]** [PC {Acc}]
[RC {Nom}] **[RC {Dat}]** **[RC {Dat}/{Acc}]** [RC {Acc}]
[ANI +] **[ANI +]** **[ANI +]** [ANI -]
cwu-keysstako] yaksokha-n-ta]]
give-Comp promise-Pres-Dec

b. *John-i **Tom-eykey** **Mary-eykey** ton-ul cwukeysstako yaksokhanta.
[RC {Dat}] [RC {Dat}]
(Interpreted as 'John promised Tom to give money to Mary.')

c. (?)John-i **Tom-ul** **Mary-eykey** ton-ul cwukeysstako yaksokhanta.
[RC {Acc}] [RC {Dat}]

d. Mary-eykey **John-i ton-ul** Tom-eykey/ul cwukeysstako yaksokhanta.
'John promised Mary to give money to Tom.'

Under our approach, this scrambling fact can be explained as well. The two NPs, *Mary-* and *Tom-*, have different PC values, so they are not constrained by the LP2. However, since they have the same ANI value, their word order is fixed when they are realized with the same RC value, i.e. Dat, as in (25b) in terms of LP3. Otherwise, they can be freely scrambled as in (25c).[10] Meanwhile, other NPs, *John-i* and

10) Some native speakers of Korean do not like the example (25c) very much, but they agree that the example, as it is, is not as bad as (25b). Note that the example becomes much better when we put *Mary-eykey* between *cwukeysstako* and *yaksokhanta*. It seems to be a little difficult to some speakers to switch the two NPs sharing a PC element as well as the ANI value though they are realized with different RC values. That is because it might cause a difficulty in processing. However, we regard that the example like (25c) is still syntactically grammatical, though it is a little difficult to process.

ton-ul, may be freely scrambled in the domains as shown in (25d) because the former NP does not share the PC or RC values with other NPs nor does the latter NP share the PC and ANI values. It follows that LP2 or LP3 does not restrict the NPs from scrambling one another.

4.5. *Mit-* ('believe') construction

The verbs like *mit-* ('believe') or *sayngkakha-* ('think') have at least two different subcategorizations. That is, they subategorize for one NP and a clause as in (26a) or two NPs and a VP as in (26b).[11] In each example, the nominative NP *Mary-ka* of the embedded clause cannot be switched with the nominative NP *John-i* of the main clause, as in (26c), while the accusative NP *Mary-lul* can, as in (26d).

(26) a. [$_{S2}$ John-i [$_{S1}$ Mary-ka papo-lako] mitnunta/sayngkakhanta.]
J-[PC {Nom}] M-[PC {Nom}]] idiot-Comp believe/think
[RC {Nom}] [RC {Nom}]
[ANI +] [ANI +]

b. [$_{S}$ John-i [$_{VP}$[$_{NP}$ Mary-lul] [$_{NP}$ papo-lako]mitnunta/sayngkakhanta.]]
J-[PC {Nom}] M-[PC {Acc}]] idiot-Comp believe/think
[RC {Nom}] [RC {Acc}]
[ANI +] [ANI +]

c. ***Mary-ka** **John-i** papo-lako mitnunta/sayngkakhanta.
(Interpreted as 'Mary believes/thinks John to be an idiot.')

---

11) Though we would assume that the examples in (26a) and (26b) have the same structure and the second NP *Mary-* alternates its case between Nom an Acc as in (i), the word order variation of these examples is also easily accounted for.

(i) John-i Mary-ka/lul papo-lako mitnunta/sayngkakhanta.
[PC {Nom}] [PC {Nom, Acc}] idiot-Comp believe/think
[RC {Nom}] [RC {Nom}/{Acc}]
[ANI +] [ANI +]
'John believes/thinks Mary to be an idiot.'

Since the two NPs, *John-* and *Mary-* have different PC values, they are not constrained by LP2. However, since they have the same ANI value, they cannot be switched each other when they have the same RC value by means of LP3. Otherwise, they can be switched, as expected.

d. **Mary-lul John-i** papo-lako mitnunta/sayngkakhanta.
'John believes/thinks Mary to be an idiot.'

The difference in scrambling like this is also accounted for under our approach. In (26a), since the two NPs, *John-* and *Mary-*, have the same RC and ANI values as well as the PC value, they cannot be scrambled by means of both LP2 and LP3. However, in (26b) since the two NPs have different RC values as well as the PC values, they are not subject to any LP constraints and can be switched each other as in (26d).

4.6. NPs without any case marker

So far we have seen that both PC and RC together with ANI information of an NP play crucial roles in deciding scrambling possibility. Now, let us look at somewhat different examples which lack case morphemes. In (27a), where the two NPs bear no case marker, the first NP *John-* is construed as subject and the second NP *Mary-* as object. When the two NPs are switched as in (27b), the first NP *Mary-* is construed as subject and the second NP *John-* as object. However, it is observed that if at least one of the two NPs is marked with a case morpheme, they can be switched each other as in (27c and d).

(27) a. John-φ Mary-φ cohaha-n-ta.
[PC {Nom}] [PC {Acc}] like-Pres-Dec
[ANI +] [ANI +]
b. *Mary-φ John-φ cohaha-n-ta.
[RC { }] [RC { }]
(Interpreted as 'Mary likes John.')
c. Mary-lul John-φ cohaha-n-ta.
[RC {Acc}] [RC { }]
d. Mary-φ John-i cohaha-n-ta.
[RC { }] [RC {Nom}]
'John likes Mary.'

Such difference in scrambling might be a puzzle to the previous RC-based approaches, but under our approach it can be accounted for. In (27a), the two NPs have different PC values, so they are not constrained by LP2. However, since they have the same ANI value, they cannot be switched when they have the same RC value, i.e. { }, in terms of LP3, as in (27b). Note that since the empty set, { }, is a subset of every set, it can be a legitimate RC value, a subset of the PC value. The examples in (27c-d), which have scrambled NPs, are grammatical because LP1 or LP2 does not restrict their scrambling.

5. Conclusion

We have examined that though the Korean language is a relatively free word order language, the canonical word order based on obliqueness is sometimes fixed under some conditions. To identify the conditions, Kuno (1980) and Chung (1998) have couched the so-called COC constraint that can be interpreted as the RC-based constraint under our analysis. Though the RC-based approach can account for some scrambling facts in Korean and Japanese, it is argued that the approach is still insufficient to cover various empirical data presented above.

In this study, we have proposed a new concept of case and scrambling mechanisms to define the conditions. Namely, we have claimed that Case consists of Potential Case (PC) and Realized Case (RC) and both kinds of Case, together with the information on Animacy (ANI) of an NP, play a role in restricting the word order variation in Korean. With this new concept of case system, we could capture the two generalizations from various scrambling data and represent them into the two LP constraints, i.e. the PC Constraint (LP2) and the RC and ANI Constraint (LP3), under the Domain Theory.

The LP Constraints restricting the scrambling possibility among NPs might be interpreted as one of the human processing strategies: When a given sentence is ambiguous, we tend to regard it as an unscrambled one, where NPs are arranged according to the obliqueness hierarchy(cf. Cho, 1996). That is, though NPs are freely scrambled in Korean as long

as they precede their head, the freedom of word order can be deprived of when scrambling may cause ambiguity in interpretation. This tendency is formally expressed within the two LP constraints.

If we are on the right track, it is expected that our approach can provide a uniform analysis for further data without stipulating anything more. Nevertheless, we still have some problems to answer: For example, how should we define the ANI value?; Are there any other conditions restricting the word order variation in Korean? etc. We leave these for future study.

References

Bratt, E. O. (1996). *Argument Composition and the Lexicon: Lexical and Periphrastic Causatives in Korean.* Doctoral dissertation. Stanford Univ.

Chai, M.-H. (1995). Order Domains and Linearization in English: Toward an HPSG Account. *Language Teaching & Research* 24. Chonnam Nat'l Univ.

Chai, M.-H. (2000). VP Complementation of the Periphrastic Causative Verb, *Korean Linguistics* 10: 183-204.

Cho, S.-Y. (1999). Two Types of Double Nominative Constructions in Korean, Presented at the Conference on Korean Linguistics in conjunction with the 1999 Linguistic Institute. UIUC. July.

Cho. S.-Y. (1996). *Non-Constituent Coordination as a Subtype of Constituent Coordination.* Ph. D. dissertation. UIUC.

Cho, S.-Y. & Chai, M.-H. (2002). Case, Animacy, Word Order in Korean. *Proceeding of the WECOL 2000*, Vol 12: 151-163.

Chung, C. (1998). Case, Obliqueness, and Linearization in Korean, *Proceedings of the FHCG-98.* Unpublished manuscript.

Kuno, S. (1980). A Further Note on Tonoike's Intra-Subjectivization Hyothesis, *MIT Working Paper in Linguistics* 2.

Lee, N.-G. & Cho, S.-Y. (1998). Case-Marking Properties of Verbal Nouns and ha-in Korean Causatives, *Selected Papers from the*

11th ICKL: 653-662. University of Hawaii at Manoa.

Mohanan, Tara. (1992). Word Order in Hindi. Paper presented at the Syntax Workshop, Stanford University.

Pollard, C. & Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. CSLI.

Pollard, C. & Sag, I. (1987). *Information-Based Syntax and Semantics: Fundamentals*. CSLI Lecture Notes, No. 13. Chicago: Univ of ChicagoPress.

Reape, M. (1994). Domain Union and Word Order Variation in German, *German in Head-Driven Phrase Structure Grammar:* 151-196. CSLI.

Reape, M. (1996). Getting Things in order. Discontinuous Constituency: 209-253. *Natural Language Processing*, No. 6. Berlin: Mouton de Grayter.

Myong-Hi Chai
Department of Liberal Arts
Chosun College of Science & Technology
290 Seoseok-dong Dong-gu
Gwanjgu, 501-744, Korea
Phone: 82-62-230-8867
Email: mhchai@mail.chosun-c.ac.kr

Sae-Youn Cho
Department of English Language & Literature
Honam University
59-1 Seobong-dong Gwangsan-gu
Gwangju 506-090, Korea
Phone: +82-62-940-5316
E-mail: sycho@honam.ac.kr

Received: 13 Jul, 04
Revised: 4 Aug, 04
Accepted: 18 Aug, 04