“Dual” Selectional Requirements on Complementation[*]

Toru Ishii
Meiji University

1. Introduction

This paper investigates “dual” selectional requirements on complementation, focusing on complementizer stacking in Japanese and Korean exemplified by (1) and (2). In (1, 2), two complementizers, *i.e.* the interrogative complementizer *ka/nya* ‘Q’ and the declarative complementizer *to/ko* ‘that’, are stacked at the right edge of the complement clause:

(1) John-wa Bill-ni [dare-ga kita **ka to**] tazuneta (Japanese)
John-TOP Bill-DAT who-NOM came **Q that** asked
‘John asked Bill who came.’

(2) John-nun Mary-eykey [*pro* kumwuncey-lul phwul-ess **nya ko**] mul-ess-ta (Korean)
John-TOP Mary-DAT that problem-ACC solved **Q that** asked
‘John asked Mary whether she solved the problem.’

I argue that complementizer stacking clauses like (1) and (2) involve “dual” selectional requirements, *i.e.* semantic selection between the matrix predicate ‘ask’ and the interrogative complementizer *ka*/*nya* ‘Q’ and syntactic selection between the matrix predicate ‘ask’ and the declarative complementizer *to*/*ko* ‘that’. Given the sisterhood condition on selection, it remains unaccounted for why the matrix predicate ‘ask’ can semantically select the interrogative complementizer *ka*/*nya* ‘Q’ skipping over the declarative complementizer *to*/*ko* ‘that’. I propose that labeling conflicts allow “relabeling” to apply as part of LF-Transfer, which accounts for the “dual” selectional requirements in complementizer stacking clauses. Since “relabeling” is an operation where labeling applies without Merge, the proposed analysis presents further evidence for the symmetric Merge coupled with labeling algorithms approach (Chomsky 2008, 2013, 2015), which claims that Merge and labeling are independent operations. It also supports the view that labeling is needed not only for interpretations at the interfaces but also for selection as advocated by Chomsky (2008) and Blümel (2017).

[*] This is a revised version of the paper presented at the joint conference of GLOW in Asia XII and SICOGG XXI. I would like to thank the audience at the conference for helpful comments and suggestions. I would also like to thank Željko Boškoviç, Noam Chomsky, Hidehito Hoshi, Yasuyuki Kitao, Shigeru Miyagawa, Norvin Richards, and Mamoru Saito for helpful comments on earlier versions of this paper. Remaining errors and omissions are, of course, the sole responsibility of the author. This work is supported in part by the Japan Society for the Promotion of Science under grant Scientific Research C 18K00666.

The organization of this paper is as follows. Section 2 investigates complementizer stacking clauses in Japanese. It is shown that complementizer stacking clauses involve "dual" selectional requirements. Section 3 presents evidence against a direct quotation analysis of complementizer stacking clauses. Section 4 discusses complementizer stacking clauses in Korean. Section 5 proposes a "relabeling" analysis of complementizer stacking clauses, which gives us an account of "dual" selectional requirements. Section 6 makes concluding remarks.

2. Complementizer Stacking in Japanese

2.1 Semantic Selection

Predicates like *tazuneru* 'ask' semantically select an interrogative clause. They can take a clause headed by the interrogative complementizer *ka* 'Q' as shown in (4b) whereas they cannot take a clause headed by the declarative complementizer *to* 'that' as shown in (4a):

(3) a. * John-wa Bill-ni [Mary-ga kita **to**] tazuneta
John-TOP Bill-DAT Mary-NOM came **that** asked
Lit. 'John asked Bill that Mary came.'
b. John-wa Bill-ni [dare-ga kita **ka**] tazuneta
John-TOP Bill-DAT who-NOM came **Q** asked
'John asked Bill who came.'

2.2 Complementizer Stacking

As pointed out by Fukui (1986), Saito (2010), Hoshi (2011), and Miyagawa (2011), these two complementizers, *i.e.* the interrogative complementizer *ka* 'Q' and the declarative complementizer *to* 'that' can be stacked at the right edge of the complement clause selected by predicates like *tazuneru* 'ask' as shown in (1) (repeated here as (4)):

(4) John-wa Bill-ni [dare-ga kita **ka to**] tazuneta
John-TOP Bill-DAT who-NOM came **Q that** asked
'John asked Bill who came.'

Given that selection is 'local' in the sense that an element can only select its sister, a question arises how the matrix predicate can semantically select the interrogative complementizer *ka* 'Q' skipping over the declarative complementizer *to* 'that' in (4).

2.3 Syntactic Selection

One might argue that if we assume that the declarative complementizer *to* 'that' is transparent for selection in (4), we can account for the "dual" selectional requirements. As pointed out by Saito (2010), however, not all matrix predicates allow complementizer stacking. For example, predicates like *sirigagaru* 'want-to-know', *tyoosasuru* 'investigate', and *hakkensuru* 'discover' semantically select an interrogative clause as shown in (5) whereas they cannot take complementizer stacking clauses as shown in (6):

(5) John-wa [dare-ga kita **ka**] siritagatteiru
John-TOP who-NOM came **Q** want.to.know
Lit. ‘John wants to know who came.’

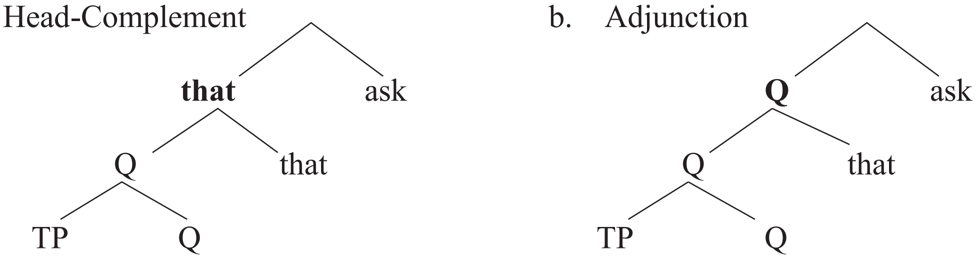
(6) * John-wa [dare-ga kita **ka** **to**] siritagatteiru
John-TOP who-NOM came **Q** **that** want.to.know
Lit. ‘John wants to know who came.’ (Saito 2010: 5)

The contrast between (4) and (6) shows the difference of syntactic selection between predicates like *tazuneru* ‘ask’ and those like *siritagaru* ‘want-to-know’; predicates like *tazuneru* 'ask' can syntactically select a declarative clause headed by *to* ‘that’ whereas predicates like *siritagaru* ‘want-to-know’ in (6) cannot.

2.4 “Dual” Selectional Requirements in Japanese

I argue that “dual” selectional requirements are involved in complementizer stacking clauses like (4); (i) semantic selection between *tazuneru* ‘ask’ and the interrogative complementizer *ka* ‘Q’ at LF, and (ii) syntactic selection between *tazuneru* ‘ask’ and the declarative complementizer *to* 'that' in overt syntax as a driving force of Merge. In the traditional analysis, there are two possible structures of (4), *i.e.* the head complement structure (7a) and the adjunction structure (7b):

(7)



a. Head-Complement

b. Adjunction

Given the sisterhood condition on selection, the “dual” selectional requirements cannot be captured by either head-complement structure or adjunction structure. The head-complement structure (7a) can capture syntactic selection but not sematic selection whereas the adjunction structure (7b) can capture semantic selection but not syntactic selection. What we need is therefore a “dual” structure, where one syntactic object may have more than one structure, *i.e.* both the head-complement structure and the adjunction structure.

One might argue that if we allow the Q-feature of the interrogative complementizer *ka* ‘Q’ to percolate up the higher phrase headed by *to* ‘that’, the matrix predicate *tazuneru* ‘ask’ can satisfy the “dual” selectional requirements. I do not adopt such a partial percolation analysis, however, since the *that*-clause would end up having both an interrogative feature [+Q] and non-interrogative feature [-Q], which is contradictory; this would result in an anomalous interpretation at LF.

3. Evidence against a Direct Quotation Analysis

Before turning to an analysis, I will show that complementizer stacking clauses are not direct quotations but complement clauses.

3.1 Polite Forms

First, as argued by Miyagawa (1987), direct *wh*-questions with *ka* ‘Q’ are deviant if the verb is in the plain form without the polite suffix *-masu*, as shown by the contrast between (8a) and (8b):

(8) a. *Dare-ga **kita** ka (plain form)
who-NOM came Q
‘Who came?’
b. Dare-ga **kimasita** ka (polite form)
who-NOM came Q
‘Who came?’

In the complementizer stacking clause in (4) (repeated here as (9)), *ka* ‘Q’ is used with the plain verb form *kita* ‘came’. This shows that the complementizer stacking clause (9) does not involve a quoted direct *wh*-question but a complementation:

(9) John-wa Bill-ni [dare-ga **kita** ka to] tazuneta (plain form)
John-TOP Bill-DAT who-NOM came Q that asked
Lit. ‘John asked Bill who came.’

3.2 Extraction

Second, direct quotations are opaque to extraction as shown in (10) and (11):

(10)* **What** did Mary say, “I am going to buy ***t***?”

(11)?***Sono situmon-ni** Mary-ga, “Dare-ga ***t*** tadasiku kotaeta no kasira”
that question-DAT Mary-NOM who-NOM correctly answered Q PART
to tazuneta rasii
that asked seem
Lit. ‘**That question**, it seems that Mary asked, “Who answered ***t*** correctly?”’

Extraction out of a complementizer stacking clause, on the other hand, is allowed as shown in (12). This also shows that complementizer stacking clauses are not direct quotations but complement clauses:

(12) **Sono situmon-ni** Mary-ga [dare-ga ***t*** tadasiku kotaeta ka to]
that question-DAT Mary-NOM who-NOM correctly answered Q that
tazuneta rasii
asked seem
Lit. ‘**That question**, it seems that Mary asked who answered ***t*** correctly.’

3.3 Pronominal Binding

Third, direct quotations are opaque to pronominal binding as shown in (13) and (14):

(13) **Mary**$_1$ asked John, “Who cheated **her**$_{*1/2}$?”

(14) **Mary**$_1$-ga John-ni, " Dare-ga **kanozyo**$_{*1/2}$-o damasita no kasira," to
Mary-NOM John-DAT who-NOM **she**-ACC cheated Q PART that
tazuneta rasii
asked seem
'It seems that Mary asked John, "Who cheated her?"'

In (13) and (14), the pronoun within the direct quotation cannot be coreferential with the matrix subject *Mary*. The pronoun *kanozyo* 'she' within the complementizer stacking clause in (15), on the other hand, can be coreferential with *Mary*. Hence, complementizer stacking clauses are not opaque to pronominal binding:

(15) **Mary**$_1$-ga John-ni [dare-ga **kanozyo**$_{1/2}$-o damasita ka to] tazuneta rasii
Mary-NOM John-DAT who-NOM **she**-ACC cheated Q that asked seem
'It seems that Mary asked John who cheated her.'

3.4 *De re* Readings

Fourth, descriptions in direct quotations may not be interpreted as *de re*. In (16a), for instance, *my mother* in the direct quotation can only be interpreted as *de dicto* but not as *de re*. This is in contrast with (16b), where *his mother* in the complement clause can be interpreted as either *de dicto* or *de re*:

(16) a. Oedipus said, "**My mother** is pretty."
De dicto: Oedipus knows she is his mother.
* *De re*: Oedipus doesn't know she is his mother.
b. Oedipus said **his mother** is pretty.
De dicto: Oedipus knows she is his mother.
De re: Oedipus doesn't know she is his mother.

This contrast regarding the *de dicto* and *di re* readings between direct quotations and complement clauses can also be observed in Japanese. In (17a), for example, *sensei* 'teacher' in the direct quotation cannot be interpreted as *de re*; (17a) is deviant in the given context. In (17b), on the other hand, *sensei* 'teacher' in the complement clause can be interpreted as *de re*; (17b) is acceptable:

(17) Context: Mary saw Jack talking with the man who was a stranger to her. She asked me who Jack was talking with. She doesn't know Jack is a teacher. In the classroom, I say to someone else:
a. #Mary-ga, "**Sensei**-wa dare-to hanasiteita no kasira," to boku-ni
Mary-NOM **teacher**-NOM who-with was.talking Q PART that I-DAT
tazuneteita yo
asked PART
'Mary asked me who the teacher was talking with.'

b. Mary-ga [**sensei**-ga dare-to hanasiteita ka] boku-ni
Mary-NOM **teacher**-NOM who-with was.talking Q I-DAT
tazuneteita yo
asked PART
'Mary asked me who the teacher was talking with.'

The complementizer stacking clause (18) is acceptable in the same context as (17), which that *sensei* 'teacher' in the complementizer stacking clause can be interpreted as *de re*. Hence, complementizer stacking clauses are not direct quotations but complements.

(18) In the same context as (17):
Mary-ga [**sensei**-ga dare-to hanasiteita ka to] boku-ni
Mary-NOM **teacher**-NOM who-with was.talking Q that I-DAT
tazuneteita yo
asked PART
'Mary asked me who the teacher was talking with.'

3.5 Temporal Modifiers

Fifth, temporal modifiers within direct quotations are evaluated relative to subjects. In (19), for instance, the temporal modifier *tomorrow* within the direct quotation is evaluated relative to the subject *Mary*; *tomorrow* is interpreted as the day after Mary's saying or asking. In (20), on the other hand, *tomorrow* in the complement clause is evaluated relative to the speaker; *tomorrow* is interpreted as the day after speech act:

(19) a. Mary said, "I will come to the party **tomorrow**."
b. Mary-ga John-ni, "Dare-ga **asita** paatii-ni kuru no kasira,"
Mary-NOM John-DAT who-NOM **tomorrow** party-to come Q PART
to tazuneteita yo
that asked PART
'Mary asked John, "Who comes to the party tomorrow?"'

(20) a. Mary said she would come to the party **tomorrow**.
b. Mary-ga [dare-ga **asita** paatii-ni kuru ka] John-ni
Mary-NOM who-NOM **tomorrow** party-to come Q John-DAT
tazuneteita yo
asked PART
'Mary asked John who would come to the party the next day.'

In the complementizer stacking clause (21), *tomorrow* is interpreted as the day after speech act, which shows that complementizer stacking clauses are complements:

(21) Mary-ga [dare-ga **asita** paatii-ni kuru ka to]
Mary-NOM who-NOM **tomorrow** party-to come Q that
John-ni tazuneteita yo
John-DAT asked PART
'Mary asked who would come to the party the next day.'

3.6 Evaluative Predicates

Sixth, evaluative predicates within direct quotations are evaluated relative to subjects whereas those within complement clauses are evaluated relative to speakers. In (22), for example, *that idiot* in the direct quotation is evaluated relative to the subject *Mary*; it is Mary who thinks that he is an idiot. In (23), on the other hand, *that idiot* in the complement clause is evaluated relative to the speaker; it is the speaker who thinks that he is an idiot:

(22) a. Mary said, “I love **that idiot**.”
b. Mary-wa John-ni, “Dare-ga **ano orokamono**-ni taikin-o
Mary-TOP John-DAT who-NOM **that idiot**-DAT a lot of money-ACC
watasita no kasira,” to tazuneta
gave Q PART that asked
‘Mary asked John, “Who gave a lot of money to that idiot?”’

(23) a. Mary said she loves **that idiot**.
b. Mary-wa John-ni [dare-ga **ano orokamono**-ni taikin-o
Mary-TOP John-DAT who-NOM **that idiot**-DAT a lot of money-ACC
watasita ka] tazuneta
gave Q asked
‘Mary asked John who gave a lot of money to that idiot.’

When *that idiot* appears in the complementizer stacking clause, it is evaluated relative to the speaker as shown in (24). This shows that complementizer stacking clauses are complements:

(24) Mary-wa John-ni [dare-ga **ano orokamono**-ni taikin-o
Mary-TOP John-DAT who-NOM **that idiot**-DAT a lot of money-ACC
watasita ka to] tazuneta
gave Q that asked
‘Mary asked John who gave a lot of money to that idiot.’

3.7 Deictic Terms

Finally, deictic terms within direct quotations are evaluated relative to subjects whereas those in complement clauses are evaluated relative to speakers. In (25), for instance, the deictic term *this* within the direct quotation is evaluated relative to the subject *Mary*; *this picture* is near *Mary*. In (26), on the other hand, the deictic term *this* within the complement clause is evaluated relative to the speaker; *this picture* is near the speaker:

(25) a. Mary said, “I want **this** picture.”
b. Mary-wa John-ni, “Dare-ga **kono** e-o kaita no kasira,”
Mary-TOP John-DAT who-NOM **this** picture-ACC drew Q PART
to tazuneta
that asked
‘Mary asked John, “Who draw this picture?”’

(26) a. Mary said that she wanted **this** picture.
b. Mary-wa John-ni [dare-ga **kono** e-o kaita ka] tazuneta
Mary-TOP John-DAT who-NOM **this** picture-ACC draw Q asked
'Mary asked John who draw this picture.'

The deictic term *this* in the complementizer stacking clause is evaluated relative to the speaker as shown in (27), which indicates that complementizer stacking clauses are complements:

(27) Mary-wa John-ni [dare-ga **kono** e-o kaita ka to] tazuneta
Mary-TOP John-DAT who-NOM **this** picture-ACC draw Q that asked
'Mary asked John who draw this picture.'

4. Complementizer Stacking in Korean

4.1 Semantic Selection

Having investigated complementizer stacking clauses in Japanese, I will then look at complementizer stacking clauses in Korean. In a Korean complementizer stacking clause, a matrix verb semantically selects a mood markers within its complement clause as exemplified by (28-31):

(28) John-nun [Mary-ka ku mwuncey-lul phwul-ess **ta**/*nya/*la/*ca ko]
John-TOP Mary-NOM that problem-ACC solved DECL/Q/IMP/EXH COMP
cwucangha-ess-ta
claimed
'John claimed that Mary solved the problem.'

(29) John-nun Mary-eykey [*pro* kumwuncey-lul phwul-ess *ta/**nya**/*la/*ca ko]
John-TOP Mary-DAT that problem-ACC solved DECL/Q/IMP/EXH COMP
mul-ess-ta
asked
'John asked Mary whether she solved the problem.'

(30) John-nun Mary-eykey [*pro* ku mwuncey-lul phwul * ta/*nya/**la**/*ca
ko]
John-TOP Mary-DAT that problem-ACC solve DECL/Q/IMP/EXH COMP
myengryengha-ess-ta
ordered
'John ordered Mary to solve the problem.'

(31) John-nun Mary-eykey [*pro* ku mwuncey-lul phwul * ta/*nya/*la/**ca** ko]
John-TOP Mary-DAT that problem-ACC solve DECL/Q/IMP/EXH COMP
ceyanha-ess-ta
suggested
'John suggested to Mary to solve the problem together.'

As shown above, the matrix verbs *cwucangha* ‘claim’, *mul* ‘ask’, *myengryengha* ‘order’, and *ceyanha* ‘suggest’ semantically selects the declarative mood marker *ta*, the interrogative mood marker *nya*, the imperative mood marker *la*, and the exhortative mood marker *ca*, respectively. Similarly, a matrix noun semantically selects a mood marker within its complement clause as exemplified by (32-35):

(32) [John-i ku mwuncey-lul phwul-ess **ta**/*nya/*la/*ca nun] cwucang
John-NOM that problem-ACC solved DECL/Q/IMP/EXH that claim
‘the claim that John solved the problem’

(33) [John-i ku mwuncey-lul phwul-ess * ta/**nya**/*la/*ca nun] cilmwun
John-NOM that problem-ACC solved DECL/Q/IMP/EXH that question
‘the question whether John solved the problem’

(34) [*pro* ku mwuncey-lul phwul * ta/*nya/**la**/*ca nun] myenglyeng
that problem-ACC solve DECL/Q/IMP/EXH that order
‘the order to solve the problem’

(35) [pro ku mwuncey-lul phwul * ta/*nya/*la/**ca** nun] ceyan
that problem-ACC solve DECL/Q/IMP/EXH that suggestion
‘the suggestion to solve the problem together’

The matrix nouns *cwucang* ‘claim’, *cilmwun* ‘question’, *myenglyeng* ‘order’, and *ceyan* ‘suggestion semantically select the declarative mood marker *ta*, the interrogative mood marker *nya*, the imperative mood *la*, and the exhortative mood marker *ca*, respectively.

4.2 Syntactic Selection

A question arises how the matrix predicate can semantically select a mood marker skipping over *ko*/*nun* in (28-35). We cannot simply assume that *ko* and *nun* are transparent for selection. As shown by the contrast between (36) and (37), matrix verbs take *ko* but not *nun* whereas matrix nouns take *nun* but not *ko*. This shows that matrix verbs syntactically select *ko* whereas matrix nouns syntactically select *nun*:

(36) a. John-un [Mary-ka ku mwuncey-lul phwul-ess ta **ko/*nun**]
John-TOP Mary-NOM that problem-ACC solved DECL that
cwucangha-ess-ta
claimed
‘John claimed that Mary solved the problem.’

b. John-un Mary-eykey [*pro* ku mwuncey-lul phwul-ess nya **ko/*nun**]
John-TOP Mary-DAT that problem-ACC solved Q that
mul-ess-ta
asked
‘John asked Mary whether she solved the problem.’

c. John-un Mary-eykey [*pro* ku mwuncey-lul phwul la **ko/*nun**]
John-TOP Mary-DAT that problem-ACC solve IMP that
myengryengha-ess-ta
ordered
'John ordered Mary to solve the problem.'

d. John-un Mary-eykey [*pro* ku mwuncey-lul phwul ca **ko/*nun**]
John-TOP Mary-DAT that problem-ACC solve EXH that
ceyanha-ess-ta
suggested
'John suggested to Mary to solve the problem together.'

(37) a. [John-i ku mwuncey-lul phwul-ess ta ***ko/nun**] cwucang
John-NOM that problem-ACC solved DECL that claim
'the claim that John solved the problem'

b. [John-i ku mwuncey-lul phwul-ess nya ***ko/nun**] cilmwun
John-NOM that problem-ACC solved Q that question
'the question whether John solved the problem'

c. [*pro* ku mwuncey-lul phwul la ***ko/nun**] myenglyeng
that problem-ACC solv IMP that order
'the order to solve the problem'

d. [*pro* ku mwuncey-lul phwul ca *** ko/nun**] ceyan
that problem-ACC solve EXH that suggestion
'the suggestion to solve the problem together'

4.3 "Dual" Selectional Requirements in Korean

I argue that "dual" selectional requirements are also involved in Korean complementizer stacking clauses; (i) semantic selection between a matrix predicate and a mood marker at LF, and (ii) syntactic selection between a matrix verb/noun and *ka/nun* 'that' in overt syntax as a driving force of Merge.

5. A Proposal

This section proposes a "relabeling" analysis of "dual" selectional requirements in Japanese and Korean complementizer stacking clauses. Before coming on to that, I will briefly explicate Merge, labeling algorithms, and labeling conflicts that are crucial in the proposed analysis.

5.1 Merge and Labeling Algorithms

Merge is a uniform operation that constructs a structure. In Chomsky's (1995) formulation of Merge (38), labeling is part of Merge, which makes Merge asymmetric:

(38) Asymmetric Merge
Merge $(\alpha, \beta) =_{\mathrm{def}} \{\gamma, \{\alpha, \beta\}\}$, where $\gamma \in \{\alpha, \beta\}$

In the asymmetric formulation of Merge (38), every node must have a label, but this is a residue of phrase structure grammar. It is therefore better to separate the labeling part from Merge. Under this view, Chomsky (2008, 2013, 2015) formulates Merge as a symmetric operation as formulated in (39)

(39) Symmetric Merge
Merge (α, β) $=_{\text{def}}$ {α, β}

In (39), labels are not created by Merge, but rather determined by labeling algorithms. I adopt Chomsky's (2008) version of labeling algorithms (40):

(40) Labeling Algorithms (Chomsky 2008: 145)
a. In {H, α}, H an LI, H is the label.
b. If α is internally merged to β, forming {α, β}, then the label of β is the label of {α, β}.

According to (40a), it is always a head that projects. (40b), which is concerned with Internal Merge, requires that the target of Internal Merge should always project. I also assume with Chomsky (2008) and Blümel (2017) that labeling is needed for selection as stated in (41):

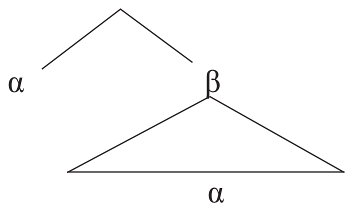
(41) Each SO [syntactic object] generated enters into further computations. Some information about SO is relevant to these computations. In the best case, a single designated element should contain all the relevant information: the *label* (the item “projected” in X'-theories; the *locus* of the label-free system of Collins 2002). The label selects and is selected in EM [External Merge], and ...
(Chomsky 2008: 141; see also Blümel 2017; supplements and underlines T.I.)

Since Merge and labeling are independent operations under the symmetric Merge coupled with labeling algorithm approach, we should expect that labeling can apply without Merge. I argue that labeling without Merge applies in complementizer stacking. More specifically, I propose that when a labeling conflict arises, “relabeling,” *i.e.* labeling without Merge, may apply as part of LF-Transfer, which accounts for the “dual” selectional requirements in complementizer stacking clauses.

5.2 Labeling Conflicts

The notion of labeling conflict has been proposed by Donati (2006), Chomsky (2008), and Cecchetto and Donati (2010; 2011). Let us consider (42) as an illustration:

(42)



Suppose that α is a lexical item, *i.e.* a head, and β is not a head. α undergoes Internal Merge with β. The labeling algorithms (40) make conflicting predictions. According to (40a), α, which is a head, should be the label. According to (40b), on the other hand, β, which is the target of Internal Merge, should be the label. They claim that a labeling conflict makes two different labels available, which creates an ambiguous structure.

As a concrete example, they argue that a labeling conflict explains free relatives and indirect questions exemplified by (43). When *what* moves to the initial position, a labeling conflict arises. (40a) requires that *what*, which is a head, should become the label whereas (40b) requires that C, which is the target of Internal Merge, should become the label. If *what* becomes the label, it creates a free relative as in (43a). If C becomes the label, it creates an indirect question as in (43b):

(43) [**what** [C [you read **what**]]]
a. I read [DP what you read].
b. I wonder [CP what your read].

5.3 A "Relabeling" Analysis of Complementizer Stacking Clauses

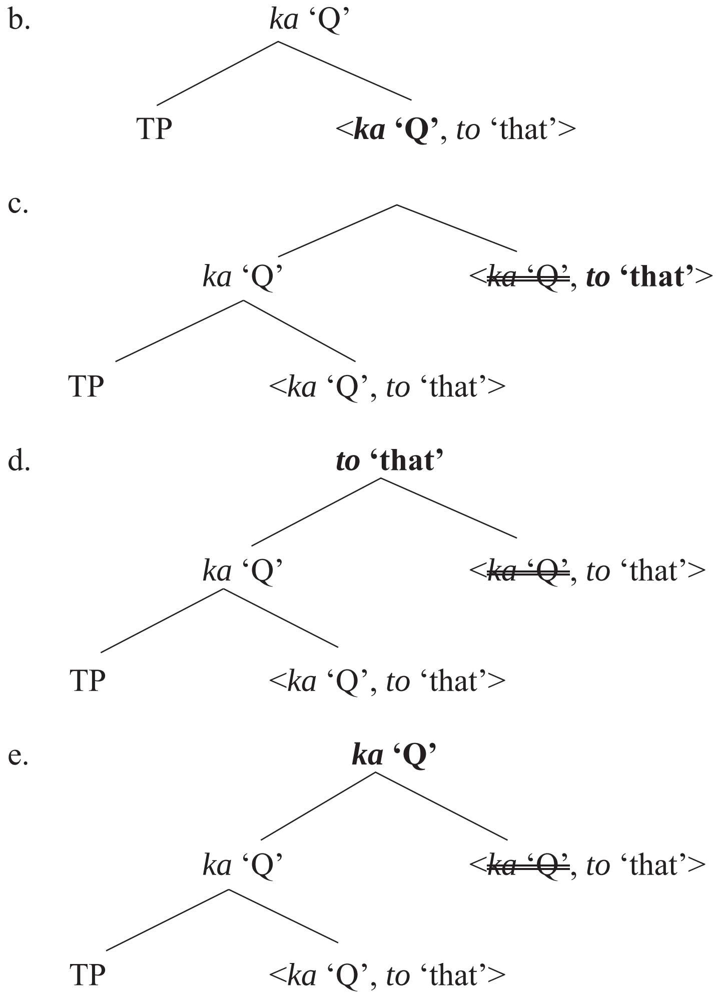
Extending their labeling conflict analysis, I argue that labeling conflicts create not only ambiguous structures but also "dual" structures because of "relabeling" as part of LF-Transfer. What I mean by "dual" structures is that one syntactic object has different structures in overt syntax and LF. I assume with Shlonsky (2006) that cartographic structure is created by self-attachment of C as stated in (44):

(44) a. The initially merged C is associated with an ordered set of LIs (bundles of features if C is null) <C_1, ... C_n>, which corresponds to Rizz's (1997) Fin, Foc, Top, etc.
b. The computational system accesses and activates these LIs (bundles of features) one by one in terms of Merge (Internal or External Merge); each time C is merged, the leftmost LI (or the leftmost bundle of features) in the set is activated.
c. Once an LI (a bundle of features) is activated, it is no longer accessible to the computational system later in the derivation.

Let us look at how a "relabeling" analysis can account for "dual" selectional requirements on complementizer stacking clauses, taking (4) (repeated here as (45)) as an example. The derivation of (45) is represented in (46):

(45) John-wa Bill-ni [dare-ga kita **ka to**] tazuneta
John-TOP Bill-DAT who-NOM came Q **that** asked
'John asked Bill who came.'

(46) a. C: <*ka* 'Q', *to* 'that'>



The initially merged C consists of the ordered set <*ka* 'Q', *to* 'that'> (46a). As represented in (46b), *ka* 'Q' is accessed and activated by the initial merger of C, *i.e.* External Merge of C with TP. According to the labeling algorithms (40), *ka* 'Q', which is a head, becomes the label. At the next stage (46c), *to* 'that' is accessed and activated by the next merger of C, *i.e.* Internal Merge (self-attachment) of C. It should be noted that *ka* 'Q', which was activated in the previous Merge, is no longer accessible to the computational system at this stage. A labeling conflict arises; the labeling algorithm (40a) requires that *to* 'that', which is a head, should become the label, whereas the labeling algorithm (40b) requires that *ka* 'Q', which is the target of Internal Merge, should become the label. This labeling conflict allows us to have two labeling options; either *to* 'that' becomes the label or *ka* 'Q' becomes the label. In this case, we take the former option; *to* 'that' becomes the label in accordance with (40a) in overt syntax, as represented in (46d). This labeling drives External Merge with the matrix predicate *tazuneru* 'ask', thereby satisfying the syntactic selection of *tazuneru* 'ask'. When we come to stage (46e) where Transfer applies, "relabeling" applies as part of LF-Transfer; *ka* 'Q' becomes the label in accordance with (40b). This labeling satisfies the semantic selection of the matrix predicate *tazuneru* 'ask' at LF. Hence, we can account for the "dual" selectional requirements on complementizer stacking clauses in terms of "dual" structures through "relabeling" due to a labeling conflict. Korean complementizer stacking can be accounted for in the same way.

Let us next consider predicates like *omou* 'think'. These predicates *syntactically* and *semantically* select a complement clause headed by *to* 'that' as shown in (47):

(47) a. John-wa [Mary-ga kita **to**] omotta
John-TOP who-NOM came that thought
'John thought that Mary came.'
b. *John-wa [dare-ga kita **ka**] omotta
John-TOP who-NOM came Q thought
Lit. 'John thought who came.'

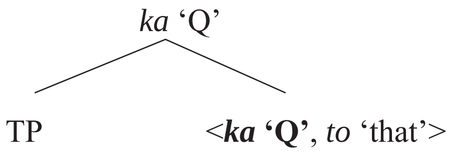
These predicates can also take a complementizer stacking clause as shown below:

(48) John-wa [dare-ga kita **ka to**] omotta
John-TOP who-NOM came **Q that** thought
Lit. 'John thought who came.'

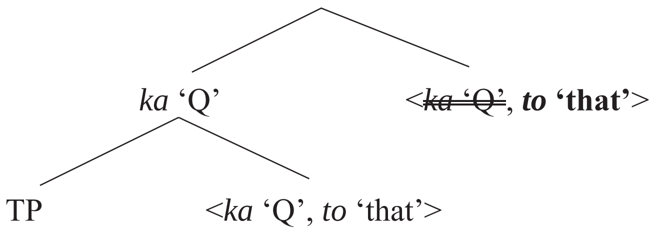
Under our analysis, the derivation of (48) proceeds as represented in (49):

(49) a. C: <*ka* 'Q', *to* 'that'>

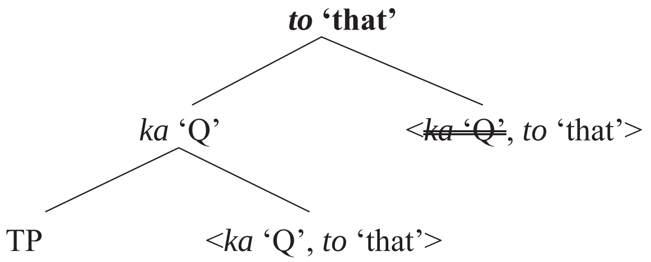
b.



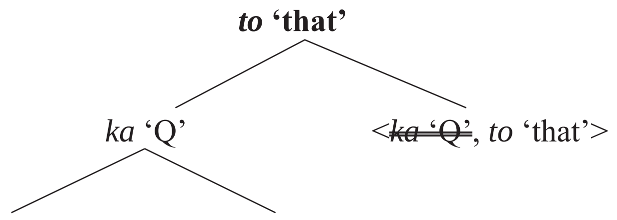
c.



d.



e.



TP <*ka* ‘Q’, *to* ‘that’>

The initially merged C consists of the ordered set <*ka* ‘Q’*, to* ‘that’>. As represented in (47b), *ka* ‘Q’ is accessed and activated by the initial merger of C, *i.e.* External Merge of C with TP. According to the labeling algorithms (40), *ka* ‘Q’, which is a head, becomes the label. At the next stage (47c), *to* ‘that’ is accessed and activated by the next merger of C, *i.e.* Internal Merge (self-attachment) of C. A labeling conflict arises; the labeling algorithm (40a) requires that *to* ‘that’ should become the label, whereas the labeling algorithm (40b) requires that *ka* ‘Q’ should become the label. Among the two labeling options this labeling conflict makes possible, we take the option where *to* ‘that’ becomes the label as represented in (47d). This labeling drives External Merge with the matrix predicate *omou* ‘think’, thereby satisfying the syntactic selection of *omou* ‘think’. Since “relabeling” as part of LF Transfer is optional, we do not apply “relabeling” in this case; *to* ‘that’ remains as the label at LF. This satisfies the semantic selection of the matrix predicate *omou* 'think' at LF. Hence, complementizer staking clauses with predicates like *omou* 'think' can be accounted for.

5.4 Complementizer Stacking in Slovene

In “dual” selectional requirements on Japanese and Korean complementizer stacking clauses discussed above, matrix predicates semantically select inner complementizer and syntactically select outer complementizer. In (45) (repeated here as (50)), for instance, the matrix predicate *tazuneru* semantically selects the inner complementizer *ka* ‘Q’ and syntactically selects the outer complementizer *to* ‘that’:

(50) John-wa Bill-ni [dare-ga kita **ka to**] tazuneta
John-TOP Bill-DAT who-NOM came Q **that** asked
‘John asked Bill who came.’

Our “relabeling” analysis of complementizer stacking clauses would predict that there should also be cases where matrix predicates semantically select outer complementizer and syntactically select inner complementizer. I argue that such cases can be found in Slovene. Unlike in languages like English, the fronted *wh*-phrase can appear with an overt complementizer in Slovene as shown below:

(51) a. ?Rad bi vedel [koga **ali** je Peter videl]
I.like would know who whether be Peter saw
Lit. ‘I would like to know who Peter saw.’
b. Sprašujem se [koga **ali** Špela lujbi]
I.wonder myself who whether Špela love
Lit. ‘I wonder who Špela loves.’
(Marvin 1997: 50; Tatjana Marvin p.c.)

In (51), the fronted *wh*-phrase *koga* ‘who’ appears with the interrogative overt complementizer *ali* ‘whether’, which shows that the matrix predicates *vedel* ‘know’ and *sprašujem* ‘wonder’ can semantically select an interrogative complement. When the fronted *wh*-phrase appears with the declarative overt complementizer *da* ‘that’, *vedel* ‘know’ can appear as the matrix predicate, but not *sprašujem* ‘wonder’ as shown by the contrast between (52a) and (52b):

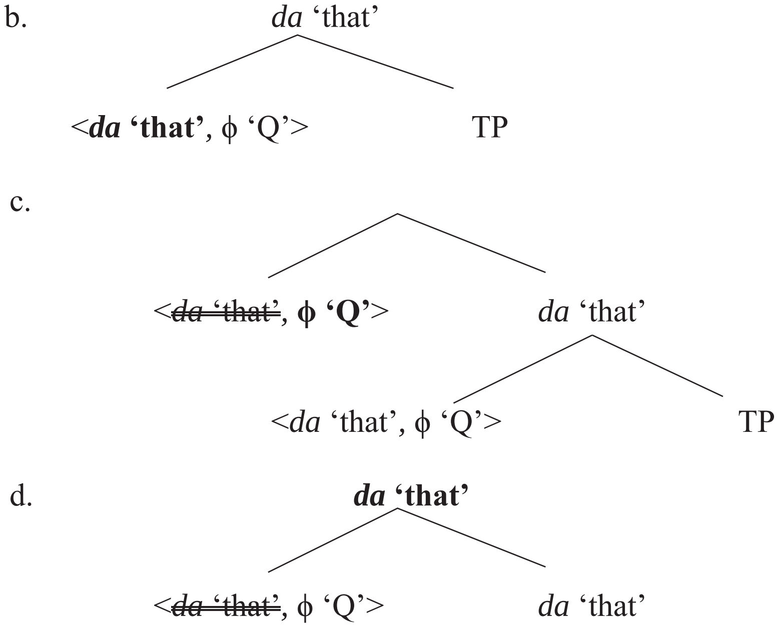
(52) a. Rad bi vedel [koga **da** je Peter videl]
I.like would know who that be Peter saw
Lit. ‘I would like to know who Peter saw.’
b. *Sprašujem se [koga **da** Špela lujbi]
I.wonder myself who that Špela love
Lit. ‘I wonder who Špela loves.’
(Marvin 1997: 50; Tatjana Marvin p.c.)

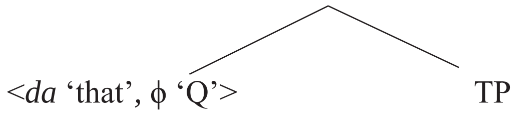
Since the *wh*-phrase *koga* ‘who’ cannot be in the Spec of the declarative complementizer *da* ‘that’, I claim that there should be a null interrogative complementizer whose specifier position is occupied by the *wh*-phrase *koga* ‘who’ as represented in (53):

(53) [koga [C[+Q] [da [...
who that

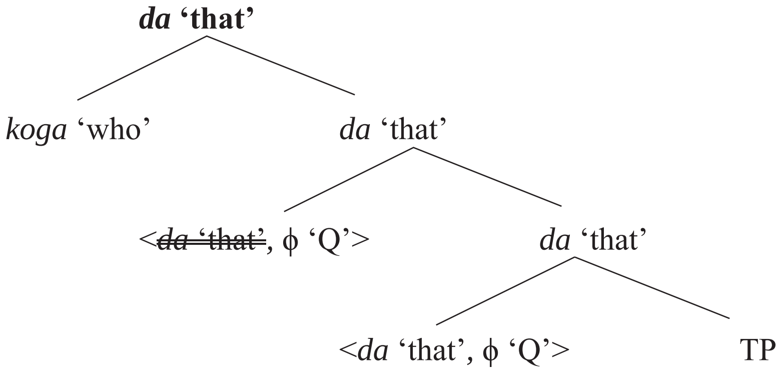
Under this analysis, although both predicates like *vedel* ‘know’ and those like *sprašujem* ‘wonder’ *semantically* select interrogative clauses, predicates like *vedel* ‘know’, but not those like *sprašujem* ‘wonder’, *syntactically* select declarative clauses. In other words, predicates like *vedel* ‘know’ syntactically select the declarative complementizer *da* ‘that’ in the inner complementizer position skipping over the interrogative null complementizer C[+Q] in the outer complementizer position. Our “relabeling” analysis can account for such “dual” selectional requirements in Slovene complementizer stacking clauses like (52a). Under our analysis, the derivation of (52a) proceeds as represented in (54):

(54) a. C: <*da* ‘that’, ϕ ‘Q’>

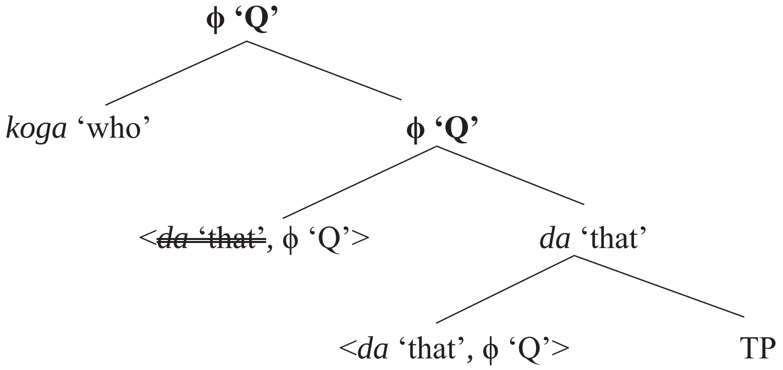




e.



f.



The initially merged C consists of the ordered set <*da* 'that', ϕ 'Q'>, consisting of the declarative complementizer *da* 'that' and the null interrogative complementizer ϕ 'Q'. As represented in (54b), *da* 'that' is accessed and activated by the initial merger of C, *i.e.* External Merge of C with TP. According to the labeling algorithms (40), *da* 'that', which is a head, becomes the label. At the next stage (47c), ϕ 'Q' is accessed and activated by the next merger of C, *i.e.* Internal Merge (self-attachment) of C. A labeling conflict arises; the labeling algorithm (40a) requires that ϕ 'Q', which is a head, should become the label, whereas the labeling algorithm (40b) requires that *da* 'that', which is the target of Internal Merge, should become the label. This labeling conflict makes it possible for us to have two labeling options; either ϕ 'Q' becomes the label or *da* 'that' becomes the label. In this case, we take the latter option; *da* 'that' becomes the label in accordance with (40b) in overt syntax, as represented in (47d). At the next stage (47e), the *wh*-phrase *koga* 'who' undergoes Internal Merge to the Spec of C; *da* 'that' becomes the label in accordance with (40b). This labeling drives External Merge with the matrix predicate *vedel* 'know', thereby satisfying the syntactic selection of *vedel* 'know'. "Relabeling" then applies as part of LF Transfer; at LF, ϕ 'Q' becomes the label of the node which is formed by self-attachment of C. ϕ 'Q' also becomes the label of the whole structure (47f) at LF concomitantly. This satisfies the semantic selection of the matrix predicate *vedel* 'know' at LF. Hence, complementizer staking clauses in Slovene also follow from our “relabeling” analysis.

6. Conclusion

This paper has first investigates complementizer staking clauses in Japanese and Korean. It was shown that complementizer stacking clauses involve the “dual” selectional requirements, which cannot be accounted for by either the traditional head-complement or adjunction structure. I have then proposed “relabeling” as part of LF Transfer due to labeling conflicts, which gives us a principled account of the “dual” selectional requirements in complementizer stacking clauses.

References

Blümel, Andrea. 2017. Exocentric root declaratives: Evidence from V2. In Leah Bauke and Andrea Blümel eds., *Labels and roots*, pp. 263-289, Berlin: De Gruyter.

Boeckx, Cedric. 2008. *Bare syntax*. Oxford: Oxford University Press.

Cecchetto, Carlo and Caterina Donati. 2010. On labeling: Principle C and head movement. *Syntax* 13: 241-278.

Cecchetto, Carlo and Caterina Donati. 2011. Relabeling heads: A unified account for relativization structures. *Linguistic Inquiry* 42: 519-560.

Chomsky, Noam. 1995. *The minimalist program*. Cambridge, MA: MIT Press.

Chomsky, Noam. 2004. Beyond explanatory adequacy. In Adriana Belletti ed., *Structures and beyond*, pp. 104-131. Oxford: Oxford University Press.

Chosmky, Noam. 2008. On phases. In Robert Friedin, Carlos P. Otero, and Maria Luisa Zubizarreta eds., *Foundational issues in linguistic theory: Essays in honor of Jean-Roger Vergnaud*, pp. 133-166. Cambridge, MA: MIT Press.

Chomsky, Noam. 2013. Problems of projection. *Lingua* 130: 33-49.

Chomsky, Noam. 2015. Problems of projection: Extensions. In Elisa Di Domenico, Cornelia Hamann, and Simona Matteini eds., *Structures and beyond - studies in honour of Adriana Belletti*, pp. 3-16. Amsterdam: John Benjamins.

Collins, Chris. 2002. Eliminating labels. In Sam Epstein and Daniel Seely eds., *Derivation and explanation in the minimalist program*, pp. 42-64. Oxford: Blackwell.

Donati, Caterina. 2006. On wh-head movement. In Lisa Lai-Shen Cheng and Norbert Corver eds., *Wh-movement: Moving on*, pp. 21-46. Cambridge, MA: MIT Press.

Fukui, Naoki. 1986. *A theory of category projection and its application*. Doctoral dissertation, MIT.

Fukui, Naoki. 2011. Merge and bare phrase structure. In Cedric Boeckx ed., *The oxford handbook of linguistic minimalism*, pp. 73-95, Oxford: Oxford University Press.

Hornstein, Norbert. 2009. *A theory of syntax: Minimal operations and universal grammar*. Cambridge: Cambridge University Press.

Hoshi, Hidehito. 2011. Merge, transfer, and labels. Ms. Doshisha University.

Marvin, Tatjana. 1997. *Wh-movement in the generative theory with special reference to Slovene*. Diploma thesis, University of Ljubljana.

Miyagawa, Shigeru. 1987. LF affix raising in Japanese. *Linguistic Inquiry* 18, 362-367.

Miyagawa, Shigeru. 2011. Agreements that occur mainly in the main clause. Ms., MIT.

Plann, Susan. 1982. Indirect questions in Spanish. *Linguistic Inquiry* 13, 297-312.

Rizz, Luigi. 1997. The fine structure of the left periphery. In L. Haegeman ed., *Elements of* *grammar*, 281-337, Dordrecht: Kluwer.

Saito, Mamoru. 2010. Sentence types and the Japanese right periphery. Ms., Nanzan University.

Shlonsky, Ur. 2006. Extended projection and CP cartography. Ms., University of Geneva.

School of Arts and Letters
Meiji University
1-1 Kandasurugadai, Chiyoda-ku
Tokyo, 101-8301 JAPAN
tishii@meiji.ac.jp