Head-internal Relatives in Japanese as Rich Context-Setters

Tohru Seraku
St. Catherine's College
Manor Road, Oxford, UK
OX1 3UJ
tohru.seraku@scatz.ox.ac.uk

Abstract

Head-Internal Relatives (HIRs) in Japanese are regarded as rich context-setters within Dynamic Syntax (DS): the propositional tree of the HIR clause is mapped onto a 'partial' tree, which establishes a rich context for the embedding clause to be parsed. This partial tree contains a situation node decorated with the Relevancy restriction and a node for an internal head. This account handles some new data and makes a novel prediction. Further, it is shown that the past DS analysis of HIRs in fact models change relatives (but not HIRs).

1 Introduction

Japanese displays so-called HIRs (Head-Internal Relatives), where the relative clause lacks a gap, the head is found inside the relative clause, and the relative clause ends with the particle *no*.

(1) [*Ringo-ga* *tsukue-no-ue-ni*
[apple-NOM table-GEN-top-at
oite-atta *no*]-*o* *Kiki-ga* *tabeta.*
place-existed NO]-ACC K-NOM ate
'An apple was on a table and Kiki ate it.'

This paper addresses Japanese HIRs in Dynamic Syntax (DS; Cann et al., 2005; Kempson et al., 2001). Sect. 2 surveys previous studies. Sect. 3 introduces DS. Sect. 4 argues that the past DS account of *no* (Cann et al., 2005) fails to capture the non-nominality of HIRs. Sect. 5 presents an alternative DS account. Sect. 6 argues that the past DS account of *no* models change relatives (but not HIRs). Sect. 7 concludes the paper.

2 Previous Studies

Several papers collected in Kuroda (1992) as a point of departure, the Japanese HIR has been extensively explored (Kitagawa, 2005; Kuroda, 2005; see references therein). Two approaches stand out. First, some scholars note parallelisms between HIRs and E-type anaphora and make use of the E-type mechanism for HIRs (Hoshi, 1995; Kim, 2007, 2008a/b, 2009; Matsuda, 2002; Shimoyama, 1999, 2001). The most advanced work in this camp is Kim's analysis. Second, others postulate the null functional head ChR (Choose Role) as a sister to VP, and assume that ChR picks out the internal head by choosing a salient thematic role in the eventuality denoted by VP (Grosu, 2010; Grosu & Landman, 2012).

Kim's E-type analysis and the ChR analysis are the two most influential accounts of HIRs in the literature, but they seem need revisions. First, it is widely held that the head in the HIR denotes a maximal set of individuals that satisfy the HIR clause description (Hoshi, 1995). For instance, for (1) to be felicitous, the situation must be the one where Kiki ate **all** of the apples on the table. But maximality effects are shown to be derived **pragmatically**. Thus, for (2) to be felicitous, a situation must be the one where each passenger puts no more than one ticket in the checker, even though he has multiple tickets, provided our world knowledge that the insertion of multiple tickles may cause malfunction of the checker (Kubota & Smith, 2007: 154).

(2) *Dono-zyookyaku*$_i$*-mo* [e$_i$ *saifu-ni*
every-passenger-too [wallet-in
kaisuuken-ga *haitteita* *no*]-*o*
coupon.ticket-NOM was.present NO]-ACC
toridashite *kaisatsu-ni* *ireta*.
pick.up ticket.checker-to put
'Every passenger picked up a coupon ticket that she/he had in (her/his) wallet and put it in the ticket checker.'

In Kim's account, maximality effects obtain due to the feature [+*definite*] of the head D, and in the ChR account, they emerge due to the feature

Copyright 2013 by Tohru Seraku
27th Pacific Asia Conference on Language, Information, and Computation pages 47－56

[MAX] of the head C. Thus, both accounts do not predict the context-dependency of maximality.

Second, HIRs are not sensitive to islands. For instance, Mihara (1994: 239) shows that the HIR (3) is not sensitive to the complex NP island.[1]

(3) [*Taro-ga* [*Hanako-ga subarashii ronbun-o*
[T-NOM [H-NOM excellent paper-ACC
kaita toiu uwasa]*-o kiiteita*
wrote TOIU rumour]-ACC has.heard
no]*-ga tsuini syuppansareta*.
NO]-NOM finally was.published
'Taro has heard a rumour that Hanako wrote an excellent paper, and the paper was finally published.'

Kim's account cannot model island-insensitivity of HIRs because it concerns only the eventuality denoted by the **highest** clause in the HIR clause (cf., Grosu (2010: 250)). In the ChR account, a null operator at Spec, ChRP undergoes cyclic A'-movement and this predicts island-sensitivity of HIRs. This prediction is said to be borne-out by considering data in Watanabe (2003), but without taking into account the examples such as (3).

Finally, it has been widely believed that the HIR clause cannot license negation (Hoshi, 1995; Grosu & Landman, 2012). The present paper, however, observes that negation is licensed if the existence of the individual denoted by the head is inferable. For instance, negation is licensed in the HIR (4) because it is inferable that there was a wallet somewhere other than a safe.

(4) *Dorobo-wa* [*saifu-ga kinko-ni*
thief-TOP [wallet-NOM safe-at
haittei-naka-tta no]*-o*
put.inside-NEG-PAST NO]-ACC
manmato nusumi-dashita.
successfully steal-took.away
'A wallet was not inside a safe (but outside the safe), and a thief successfully stole it.'

In the ChR account, they might argue that *saifu* moves over NegP at LF so that it out-scopes the negator. But this remedy is untenable since ChR cannot select NegP, anyway. This is because it is assumed that (i) VP denotes an open proposition with an event slot; (ii) ChR selects such an open proposition; but (iii) NegP closes the proposition over the event slot before it is selected by ChR (Grosu & Landman, 2012: 176). Kim's account, on the other hand, seems to correctly treat (4). In her analysis, the head denotes the maximal set of individuals that satisfy a salient property and a salient thematic role in the state denoted by the HIR clause. In (4), the property is identified with *saifu*' and the role is identified with Theme. So, the head *saifu* is correctly detected. As illustrated in (5), however, the negation data display long-distance dependency. Given that Kim's account concerns only the state denoted by the highest clause in the HIR (cf., discussion around (3)), it cannot detect the head *hoseki* in (5).

(5) *Dorobo-wa* [*aru-yumeijin-ga*
thief-TOP [certain-celebrity-NOM
[*ie-de-wa hoseki-o kinko-ni*
[house-at-TOP jewellery-ACC safe-at
irete-nai to] *TV-de itteita*
put.inside-NEG COMP] TV-at said
no]*-o manmato nusumi-dashita*.
NO]-ACC successfully steal-took.away
'A celebrity said in a TV programme that she did not put her jewellery in a safe, and the thief successfully stole it.'

These data undermine the recent works on the HIR. In this paper, I shall propose an alternative account within Dynamic Syntax.

3 Dynamic Syntax (DS)

Dynamic Syntax (DS) is a formalism that models 'knowledge of language,' construed as a set of procedures to build up an interpretation on the basis of word-by-word parsing in real time (Cann et al., 2005; Kempson et al., 2001). DS assumes semantic representation **without** a separate level of syntactic representation. So, a string is directly mapped onto a semantic structure as it is parsed left-to-right online.

3.1 A Sketch of the Formalism

DS models gradual updates of an interpretation as progressive growth of a semantic tree. The initial state is specified by the Axiom:

(6) Axiom
?t, ◊

The Axiom sets out a node decorated with ?t, a requirement that this node will be of type-t. A pointer ◊ indicates a node under development. A parser updates this initial tree state by executing general, lexical, and pragmatic actions. Every

[1] Kuroda (2005) suggests that the Complex NP Constraint may be at work. At the same time, however, he notes that the HIR involving the complex NP is not totally degraded.

time a node is created, it comes with a set of requirements, and every tree update is driven by some form of requirements. A DS tree is said to be well-formed iff no outstanding requirements remain. A string is said to be grammatical iff there is a tree update that leads to a well-formed tree. For instance, if a parser processes (7), it gradually updates the initial state (6) by running general, lexical, or pragmatic actions until the well-formed tree (8) emerges, where there are no outstanding requirements. (Throughout this paper, tense is set aside; see Cann (2011).)

(7) *Kiki-ga hashi-tta.*
K-NOM run-PST
'Kiki ran.'

(8)
    hashi'(*Kiki*')(*SIT*) : t, ◊
        SIT : e_S
        hashi'(*Kiki*') : e_S→t
            Kiki' : e
            hashi' : e→(e_S→t)

DS trees are binary-branching, an argument being on the left and a functor on the right. Each node is decorated with a pair α : β, where α is a semantic content and β is a set of labels that show various properties of the content such as logical type. In (8), *hashi* (= 'run') takes not only the subject term *Kiki*' but also the situation term *SIT*. DS assumes that all verbs select a situation term of type-e (cf., Davidson (1967)). The type of situation term is notated as e_S.

The backbone of DS trees is LOFT (Logic Of Finite Trees; Blackburn & Meyer-Viol (1994)). LOFT is a language to talk about node relations. Two operators are of particular relevance to this paper. $<\downarrow_0>$ refers to an argument daughter and $<\downarrow_1>$ refers to a functor daughter, together with their inverses: $<\uparrow_0>$ and $<\uparrow_1>$. These operators may be used in conjunction with labels. Thus, $<\downarrow_0>(e_S)$ states that the argument daughter is of type-e_S. This holds at the top node in the tree (8).

As stated above, a set of requirements drives the application of general, lexical, or pragmatic actions to update a tree state. An action package is in the following conditional format:

(9) IF (input condition)
THEN (action; if the condition is met)
ELSE (action; if it is not met)

The IF-block is a condition on the node marked by the pointer ◊. The THEN-block specifies an action to be run if the condition is met whereas the ELSE-block specifies an action to be run if the condition is not met. Let us consider an action package that is encoded in a verb. Since Japanese is pro-drop, it is assumed that all verbs project a propositional template. For instance, the verb *hashi* (= 'run') generates the tree (10).

(10) Parsing *hashi* (= 'run')
    ?t
        U : e_S
        ?(e_S→t)
            V : e
            hashi' : e→(e_S→t), ◊

Each argument node is annotated with a meta-variable, a place-holding device to be saturated with a term such as *Kiki*'. The action package to generate the tree (10) is formulated as follows:

(11) Entry of *hashi* (= 'run')
IF ?t
THEN make/go($<\downarrow_0>$); put(U : e_S); go($<\uparrow_0>$)
make/go($<\downarrow_1>$); put(?(e_S→t));
make/go($<\downarrow_0>$); put(V : e); go($<\uparrow_0>$)
make/go($<\downarrow_1>$); put(*hashi*' : e_S→(e→t))
ELSE ABORT

The IF-block declares that a parser performs the actions in the THEN-block iff a current node is a type-t-requiring node. (If this is not met, ABORT applies; the tree update is quitted.) The THEN-block consists of primitive actions. make/go(α) is an action to create a node α and move a pointer ◊ to the node. Since $<\downarrow_0>$ refers to an argument daughter, make/go($<\downarrow_0>$) is an action to create an argument daughter and moves a pointer ◊ to the node. put(α) is an action to decorate a current node with α. So, put(?(e_S→t)) decorates a current node with ?(e_S→t). These atomic actions build the tree (10).

DS adopts the epsilon calculus for modelling quantification. The epsilon calculus, proposed by David Hilbert, is the logic of arbitrary names in natural deduction in Predicate Logic (Kempson et al., 2001). All quantified NPs are mapped onto an epsilon term, a type-e term defined as a triple: a binder, a variable, and a restrictor. For instance, *neko* (= 'a cat')[2] is mapped onto (ε, x, *neko*'(x)), where ε is an epsilon binder (analogous to ∃), x a variable, and *neko*'(x) a restrictor. A situation term is notated as *SIT* in (8) but it is precisely

[2] Japanese lacks determiners. Thus, the quantificational force of bare NPs is contextually determined.

expressed as an epsilon term such as (ε, s, *S*(s)). (For the situation predicate *S*, see Cann (2011).)

Once a proposition emerges, each epsilon term is evaluated for scope. This process, Quantifier-Evaluation (Q-Evaluation), explicates the scope dependencies; the restrictor of a term is enriched with the other predicates in the proposition. For instance, the proposition (12) contains two terms. Suppose that the situation term (ε, s, *S*(s)) out-scopes the subject term (ε, x, *neko*'(x)).

(12) *hashi*'(ε, x, *neko*'(x))(ε, s, *S*(s))

A term having a narrow scope is Q-Evaluated first. So, (ε, x, *neko*'(x)) is evaluated first, to the effect that (12) is updated to (13). The evaluated epsilon term, abbreviated as a, reflects not only the original predicate *neko*' but also the predicate *hashi*' into the restrictor, with the connective & for existential quantification.

(13) *neko*'(a)&*hashi*'(a)(ε, s, *S*(s))
where a = (ε, x, *neko*'(x)&*hashi*'(x)(ε, s, *S*(s)))

The same procedure then applies to the situation term, and (13) is updated into (14).

(14) *S*(b)&[*neko*'(a_b)&*hashi*'(a_b)(b)]
where b = (ε, s, *S*(s)&[*neko*'(a_s)&*hashi*'(a_s)(s)])
a_b = (ε, x, *neko*'(x)&*hashi*'(x)(b))
a_s = (ε, x, *neko*'(x)&*hashi*'(x)(s))

The technical detail here is unimportant. What is essential is that (i) Q-Evaluation algorithmically applies to a term in the reverse-order of the scope relation, (ii) each evaluated term reflects the full content of the proposition into the restrictor, and (iii) the output such as (14) explicates the full scope dependency.

In closing this DS exegesis, the LINK device needs to be mentioned. So far, only individual trees have been considered, but two discrete trees may be built up in tandem and paired in virtue of a shared term. This formal tree pairing is called 'LINK.' The LOFT operator <L> refers to the LINKed node from the perspective of a current node. The inverse is defined as <L^{-1}>. For details, see Sect. 4 and, especially, Sect. 5.1.

3.2 A Sample Tree Update

Progressive growth of a DS tree vis-à-vis left-to-right parsing is illustrated with the string (15). The initial state is the Axiom (16), and a parser incrementally updates this initial tree by running general, lexical, or pragmatic actions.

(15) *Neko-ga* *hashi-tta.*
cat-NOM run-PST
'A cat ran.'

(16) Axiom
?t, ◊

First, the actions encoded in *neko* and *ga* induce a subject node decorated with the content of *neko* and the logical type e.[3]

(17) Parsing *Neko-ga*
?t, ◊
?(e_S→t)
(ε, x, *neko*'(x)) : e

Next, *hashi* (= 'run') projects a propositional schema, where a situation and a subject node is decorated with a meta-variable (cf., (10)). Note that a subject node is already present in (17). This pre-existing node harmlessly collapses with the subject node created by *hashi*.

(18) Parsing *Neko-ga hashi-tta* (ignoring tense)
?t, ◊
U : e_S ?(e_S→t)
(ε, x, *neko*'(x)) : e *hashi*' : e→(e_S→t)

Two daughter nodes at the bottom are specified for content and type. Thus, functional application and type deduction compute the content and type of the mother node. This process, formulated as the general action Elimination, also applies to the intermediate argument-functor pair, yielding the decoration at the top node.

(19) Elimination (twice)
hashi'(ε, x, *neko*'(x))(ε, s, *S*(s)) : t, ◊
(ε, s, *S*(s)) : e_S *hashi*'(ε, x, *neko*'(x)) : e_S→t
(ε, x, *neko*'(x)) : e *hashi*' : e→(e_S→t)

[3] Formally, the general action Local *Adjunction induces an unfixed node, to be decorated by *neko* and to be fixed as a subject node by the nominative case particle *ga*.

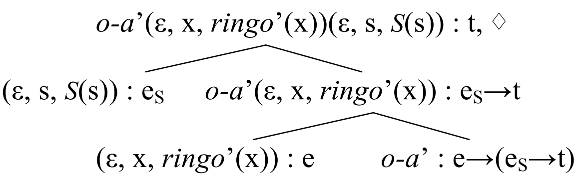
This is a well-formed final state in that it has no outstanding requirements. The proposition at the top node is Q-Evaluated; see (14) for the output.

4 A Previous DS Account

Building on Kurosawa (2003), Cann et al. (2005) and Kempson & Kurosawa (2009) propose that *no* in HIRs is a LINK-inducing nominaliser. For instance, consider (20). The parse of (20) up to *oite-atta* yields the tree (21). The proposition at the top node in (21) is Q-evaluated as in (22):

(20) [*Ringo-ga oite-atta no*]-*o*
[apple-NOM place-existed NO]-ACC
Kiki-ga tabe-ta.
K-NOM eat-PST
'There was an apple and Kiki ate it.'

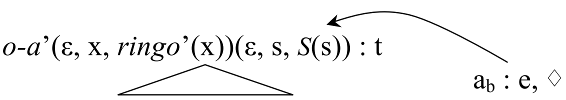
(21) Parsing the string (20) up to *oite-atta*



(22) Evaluating the proposition in (21)
$S(b)\&[ringo'(a_b)\&o\text{-}a'(a_b)(b)]$
where $b = (\varepsilon, s, S(s)\&[ringo'(a_s)\&o\text{-}a'(a_s)(s)])$
$a_b = (\varepsilon, x, ringo'(x)\&o\text{-}a'(x)(b))$
$a_s = (\varepsilon, x, ringo'(x)\&o\text{-}a'(x)(s))$

Now, *no* (i) initiates a LINK relation to a type-e-requiring node and (ii) decorates the node with a term in the evaluated proposition (in this case, a_b in (22)). In the tree display (23), a LINK relation is expressed by a curved arrow:

(23) Parsing the string (20) up to *no*



where a_b is as defined in (22).

The rest of the process is as usual. Especially, the node decorated with a_b is identified as an object node by the accusative case particle *o*, and *tabe*-projects a propositional schema, where the object node collapses with the pre-existing object node.

The heart of this analysis is that *no* is regarded as a **nominaliser**: it maps a proposition onto a term denoting an entity reflecting the proposition. Seraku (in prep.) demonstrates that this entry of *no* models FRs (Free Relatives), where *no* is seen to have the nominal status (Tonosaki, 1998).

Unlike FRs, however, HIRs possess a number of **non-nominal** characteristics. First, when the nominaliser *no* denotes a human, it has a (mostly, derogatory) connotation (cf., Kuroda (1992)). So, *no* in the FR (24) may have such connotation but *no* in the HIR (25) does not. This suggests that individuals are not denoted in HIRs.

(24) [*Naita no*]-*o Kiki-ga nagusameta.*
[cried NO]-ACC K-NOM consoled
'Kiki consoled a person who cried.'

(25) [*Tomodachi-ga naita no*]-*o*
[friend-NOM cried NO]-ACC
Kiki-ga nagusameta.
K-NOM consoled
'A friend cried and Kiki consoled him.'

Second, the relative clause is modifiable by demonstratives in FRs but not in HIRs (Tonosaki, 1998). Given that only individual-denoting items may be modified, it seems that an individual is denoted in the FR (26) but not in the HIR (27).

(26) *Sono* [*Kiki-ga katta no*]-*o*
that [K-NOM bought NO]-ACC
Jiji-ga tabeta.
J-NOM ate
'Jiji ate that thing which Kiki bought.'

(27) **Sono* [*Kiki-ga ringo-o katta*
that [K-NOM apple-ACC bought
no]-*o Jiji-ga tabeta.*
NO]-ACC J-NOM ate
'Kiki ate that apple and Jiji ate it.'

Third, FRs but not HIRs may offer an answer to *wh*-questions asking about an individual (cf., Matsuda (2002)). For instance, the *wh*-question *Who did Kiki console?* may be answered by the FR (28) but not by the HIR (29).

(28) [*Naita no*]-*o nagusameta.*
[cried NO]-ACC consoled
'Kiki consoled a person who cried.'

(29) ♯[*Tombo-ga naita no*]-*o nagusameta.*
[T-NOM cried NO]-ACC consoled
Int. 'Kiki consoled Tombo, who cried.'

Finally, a focus position in clefts is occupied by FRs, but not HIRs. Given that only a nominal item is focussed in Japanese clefts (Seraku, in prep.), it follows that FRs, but not HIRs, denote a nominal entity.

(30) [*Kiki-ga* *tabeta* *no*]-*wa* [*Osono-ga*
[K-NOM ate NO]-TOP [O-NOM
yaita *no*] *da.*
baked NO] COP
'It is [the thing that Osono baked] that Kiki ate.'

(31) *[*Kiki-ga* *tabeta* *no*]-*wa* [*Osono-ga*
[K-NOM ate NO]-TOP [O-NOM
pan-o *yaita* *no*] *da.*
bread-ACC baked NO] COP
'It is Osono's baked bread that Kiki ate.'

To sum up, it seems reasonable to assume that HIRs do not denote individuals; see also Seraku (in prep.) for further sets of data that point to the same conclusion. Thus, while the entry of *no* in Cann et al. (2005) deal with nominalisation data appropriately (Seraku, in prep.), it cannot predict the **non-nominal** status of HIRs.

Further, the entry of *no* in Cann et al. (2005) fails to account for why **only** HIRs (but not other types of relatives) are subject to the Relevancy Condition (Kuroda, 1992). The detail is still a controversy (Kim, 2007) but it requires that the event described by the HIR clause should be a relevant sub-event of the event described by the embedding clause. One construal of relevancy is 'temporal contiguity'; for instance, the HIR (25) cannot be interpreted as: 'A friend cried 1 year ago and Kiki consoled him today.' By contrast, this reading is possible in the FR (24). So, if *no* in Cann et al. (2005) applies to both HIRs and FRs, the Relevancy Condition asymmetry is left as a mystery.

5 A New DS Account

5.1 Proposal

I now propose an alternative DS account of HIRs. The last section has argued for the non-nominal status of HIRs. What remains unclear is why the HIR clause is case-marked, though case particles are usually attached to nominal items.

This apparent conflict is solved if HIRs are regarded as **rich context-setters**: the proposition of the HIR clause is mapped onto a propositional structure that is **partially articulated** when it is introduced. The embedding clause will be parsed with this partial tree as **context**. The partial tree contains two nodes. First, a situation node comes with the requirement that the situation term in this main tree will be in a 'Relevancy' relation to the situation node of the HIR clause. Second, a node for an individual term is present and it is decorated with the content of a head. This makes sure that the head, though internal to the relative clause, is selected by the embedding verb. The position of the node is guided by the case particle. For instance, in the sequence *no-ga*, where *ga* is a nominative-case particle, the node of the head is identified as a subject node. I shall propose that this tree update is lexically triggered by the sequence '*no* + case particle.'[4]

(32) Proposal (see (40) below for formal details)
The unit '*no* + case particle' maps the tree of the HIR clause onto a **partial** tree which involves (i) a situation node decorated with the 'Relevancy' requirement and (ii) a node for an internal head. The node position of the head is signalled by the case particle.

To illustrate (32), consider the HIR (33). The parse of (33) up to *oite-atta* yields the tree (34) (cf., (21)). The proposition at the top node is then Q-Evaluated as in (35) (cf., (22)).

(33) [*Ringo-ga* *oite-atta* *no*]-*o*
[apple-NOM place-existed NO]-ACC
Kiki-ga *tabe-ta.*
K-NOM eat-PST
'There was an apple and Kiki ate it.'

(34) Parsing the string (33) up to *oite-atta*

o-a'(ε, x, *ringo*'(x))(ε, s, *S*(s)) : t, ◊

(ε, s, *S*(s)) : e_S *o-a*'(ε, x, *ringo*'(x)) : e_S→t

(ε, x, *ringo*'(x)) : e *o-a*' : e→(e_S→t)

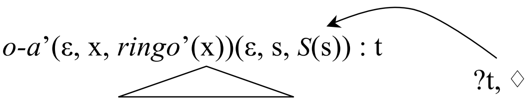
(35) Evaluating the proposition in (34)

S(b)&[*ringo*'(a_b)&*o-a*'(a_b)(b)]
where b = (ε, s, *S*(s)&[*ringo*'(a_s)&*o-a*'(a_s)(s)])
a_b = (ε, x, *ringo*'(x)&*o-a*'(x)(b))
a_s = (ε, x, *ringo*'(x)&*o-a*'(x)(s))

Now, *no-o* drives lexical actions. First, it LINKs the type-t node onto the type-t-requiring node.

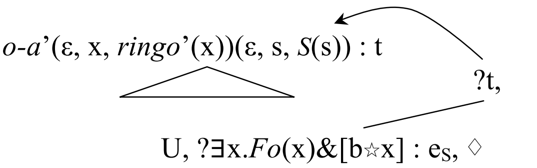
[4] Seraku (in prep.) argues that the sequence '*no* + the topic particle *wa*' models clefts. Like HIRs, a propositional tree is mapped onto another propositional tree. In this view, clefts are regarded as **context-setters**: the pre-*no-wa* part sets a context for the focus item to be parsed. But unlike HIRs, the mapped tree in clefts **lacks** internal structure (i.e., it is not partially articulated when it is induced.) Hence, clefts as context-setters, and HIRs as rich context-setters.

(36) <u>Parsing (33) up to *no-o*: the part (i)</u>



Second, a parser creates a situation node with the requirement that the term will contain as a sub-term a situation term in the previous proposition, in the present case, the situation term b in (35). This is expressed as ?∃x.*Fo*(x)&[b☆x]. *Fo* is a formula predicate (Kempson et al., 2001) and ☆ stands for whatever relation holds between the events denoted by the HIR and the matrix clauses, as governed by the Relevancy Condition.

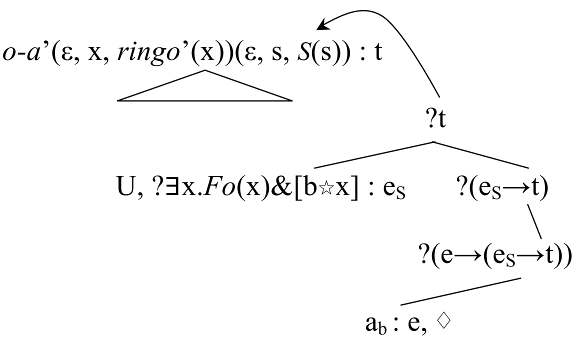
(37) <u>Parsing (33) up to *no-o*: the part (ii)</u>



where b is as defined in (35)

Finally, a parser creates a node for a head. In the present case, this is decorated with a_b in (35).[5] The node position is guided by the case particle; in (33), the accusative case particle signals that the term a_b is at an **object** node.

(38) <u>Parsing (33) up to *no-o*: the part (iii)</u>

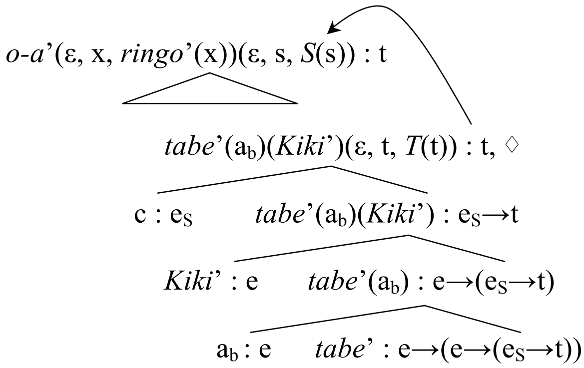


where a_b and b are as defined in (35)

This partial tree is a rich context against which the matrix clause is subsequently parsed. Within this partial tree, (i) *Kiki-ga* introduces a subject node; (ii) the matrix verb *tabe* (= 'eat') projects a propositional schema; (iii) each argument node collapses with the pre-existing nodes. The final tree state is given in (39).

[5] The selection of a term is pragmatically determined. This models the indeterminacy of HIR heads (Kuroda, 1992).

(39) <u>Parsing the whole string (33): final state</u>



where c = *Fo*(ε, t, *T*(t))&[b☆(ε, t, *T*(t))]
a_b and b are as defined in (35)

The entry of '*no* + case particle' is formally presented as follows:

(40) <u>Entry of the unit '*no* + case particle'</u>

IF t
THEN IF φ[(α : e_S), (β : e)]
  THEN make/go(<L^{-1}>); put(?t);
    make/go(<$\downarrow_0$>);
    put(U, ?∃x.*Fo*(x)&[α☆x] : e_S);
    go(<$\uparrow_0$>); make/go(<μ>);
    put(β : e)
  ELSE ABORT
ELSE ABORT

where $\mu \in \{\downarrow_1\downarrow_0, \downarrow_1\downarrow_1\downarrow_0, \downarrow_1\downarrow_1\downarrow_1\downarrow_0, \ldots\}$

φ stands for an evaluated proposition of the HIR clause. α is a situation term occurring in φ and β a non-situation term occurring in φ. μ stands for some LOFT-relation and its value is fixed by a case particle: the nominative case particle selects $\downarrow_1\downarrow_0$ (i.e., subject), the accusative case particle $\downarrow_1\downarrow_1\downarrow_0$ (i.e., object), and the dative case particle $\downarrow_1\downarrow_1\downarrow_1\downarrow_0$ (i.e., indirect object). I shall assume only these three case specifications here, but the set could be enriched (Seraku, in prep.).

One may object that (40) is a stipulation, but Seraku (in prep.) shows that (40) is defined based on the entries of the nominaliser *no* and the cleft marker *no-wa*; see Seraku (in prep.) Further, the fusion of *no* and a case particle is diachronically plausible; these fusions yielded many sentential connectives such as *no-ni* (= 'though'). Kuroda (2005: 230, fn 37) suggests that such connectives may have developed from the sequence '*no* + case particle' through the use of HIRs.

5.2 Non-nominal Nature of HIRs

The entry (40) models the non-nominal features of HIRs in Sect. 4. First, *no* in HIRs is no longer regarded as a nominaliser as conceived in FRs. Thus, the lack of connotation in the HIR (25), repeated here as (41), is anticipated.

(41) [*Tomodachi-ga naita no*]-*o*
[friend-NOM cried NO]-ACC
Kiki-ga nagusameta.
K-NOM consoled
'A friend cried and Kiki consoled him.'

Second, in our analysis, the tree of the HIR clause is mapped onto a **type-t**-requiring node. This is contrasted with FRs, where *no* maps the tree of the relative clause onto a **type-e**-requiring node. Provided that demonstratives only modify a type-e item, it is thus expected that they cannot modify HIRs. Consider (27), re-cited here as (42).

(42) **Sono* [*Kiki-ga ringo-o katta*
that [K-NOM apple-ACC bought
no]-*o Jiji-ga tabeta.*
NO]-ACC J-NOM ate
'Kiki ate that apple and Jiji ate it.'

Third, since the mapped tree is of **type-t**, it is also expected that HIRs cannot offer an answer to *wh*-questions asking about individuals. This is why the HIR (29), repeated here as (43), cannot answer to the question *Who did Kiki console?*

(43) ♯[*Tombo-ga naita no*]-*o nagusameta.*
[T-NOM cried NO]-ACC consoled
Int. 'Kiki consoled Tombo, who cried.'

For the same reason, the HIR (31), reproduced here as (44), cannot be at a type-e focus position.

(44) *[*Kiki-ga tabeta no*]-*wa* [*Osono-ga*
[K-NOM ate NO]-TOP [O-NOM
pan-o yaita no] *da.*
bread-ACC baked NO] COP
'It is Osono's baked bread that Kiki ate.'

In the literature, there is some indication that HIRs exhibit a nominal property (Hoshi, 1995; Kuroda, 2005). In the HIR (45), the *no*-part looks as though it stands as a nominal that licenses the numeral quantifier *san-mai.*[6]

(45) *Kiki-wa* [*pan-ga teiburu-ni*
K-TOP [bread-NOM table-on
oiteatta no]-*o san-mai tabeta.*
place.existed NO]-ACC 3-CL ate
'Kiki ate 3 slices of bread on a table.'

But (45) does not show the nominality of HIRs. In our analysis, the unit *no-o* introduces an object node and decorates it with the evaluated content of the head *pan*. It is this **content** that licenses the numeral quantifier *san-mai.* In fact, as shown in (46), *san-mai* may be licensed even if there is no overt host NP as long as there is a proper content that denotes a salient object, say, bread. (In DS terms, the object meta-variable posited by *tabe* (= 'eat') is pragmatically substituted with a content denoting a salient object such as bread.)

(46) *Kiki-wa san-mai tabeta.*
K-TOP 3-CL ate
'Kiki ate 3 slices of something (e.g., bread).'

5.3 Maximality, Islands, and Negation

Another benefit of the entry (40) is that the data in Sect. 2 also follow. First, (40) says nothing about maximality effects. For instance, the term of the internal head in (35), namely a_b, as re-cited here as (47), only involves the epsilon binder ε, which is analogous to the existential operator ∃.

(47) a_b = (ε, x, *ringo*'(x)&*o-a*'(x)(b))
b = (ε, s, *S*(s)&[*ringo*'(a_s)&*o-a*'(a_s)(s)])
a_s = (ε, x, *ringo*'(x)&*o-a*'(x)(s))

So, the term a_b itself does not encode maximality. This models the context-dependent nature of the maximality effect as illustrated in (2).

Second, in the entry (40), β is a term of the internal head. Importantly, (40) does not impose any structural restriction on where β is detected within the evaluated proposition. This captures island-insensitivity of HIRs (3).

Third, negation data are also handled. DS has not explored negation but it is reasonable to hold that the negator interacts with quantifiers to fix the scope. In (4), Q-Evaluation may give rise to a proposition where the term of *saifu* (= 'a wallet') out-scopes the negator. A parser makes a copy of this term and puts it at an object node built by the sequence *no-o*.

5.4 The Relevancy Condition

The Relevancy predicate ☆, though it does not spell out the Relevancy Condition, offers a basis for modelling that only HIRs are subject to the condition. A research avenue is to substantiate ☆

[6] One may claim that *san-mai* is licensed by the internal head *pan* (= 'bread') and it is then floated out of the HIR clause. But this analysis is not plausible because quantifier float is clause-bounded; see Hoshi (1995: 36-50).

by representing aspects and tense within situation terms (cf., (Cann, 2011)).

Still, the entry (40) at its present form makes a novel prediction: the condition holds between the HIR clause and its **immediate** embedding clause. Consider (48). The HIR clause has to be relevant to the intermediate clause *Kiki-ga tabeta* but not to the matrix clause *Jiji-ga itta*. Thus, (48) may have the reading: 'There was an apple and Kiki ate it. Then, 3 years later, Jiji said about it.' This restriction is predicted by the entry (40) since ☆ is put at a situation node in the structure of the **immediately** embedding clause.

(48) [[*Ringo-ga* *oite-atta* *no*]-*o*
[[apple-NOM place-existed NO]-ACC
Kiki-ga *tabeta* *to*] *Jiji-ga* *itta.*
T-NOM ate COMP] J-NOM said
'Jiji said that [there was an apple and Kiki ate it].'

Is this generalisation expressible in previous works? In Kim's E-type analysis, the HIR clause moves and adjoins to a higher AspP. So, it must be assumed that it does not move over the AspP for *Kiki-ga tabeta*. In the ChR account, the null OP at Spec of ChRP may undergo successive cyclic A'-movement. Thus, it must be assumed that the null OP does not move up to Spec of CP within the matrix clause. These assumptions may be justified in terms of computational economy, but no such justification is as yet provided.

6 Change Relatives (CRs)

It is argued that Cann et al.'s (2005) entry of *no* is not applicable to HIRs. Then, is this entry to be eliminated? The answer is negative. First, it treats *no*-nominalisation data (Seraku, in prep.). Second, as will be argued below, it also accounts for CRs (Change Relatives), a much less studied type of Japanese relatives.

CRs denote the 'state of change,' as illustrated in (49) (Tonosaki, 1998: 144).

(49) [*Otamajyakushi-ga* *kaeru-ni* *natta*
[tadpole-NOM frog-COP became
no]-*ga* *niwa-o* *haneteiru.*
NO]-NOM garden-in is.hopping
'A frog which is the result of changing from a tadpole is hopping in the garden.'

CRs are quite similar to HIRs at a surface level: the head is inside the relative clause without a gap and the relative clause ends with *no*. Yet, Tonosaki (1998) claims that CRs behave more like FRs than HIRs.[7] A convincing set of data concerns modifiability: like FRs and unlike HIRs, *sono* may be put in CRs as exemplified in (50).

(50) *Sono* [*otamajyakushi-ga* *kaeru-ni*
that [tadpole-NOM frog-COP
natta *no*]-*ga* *niwa-o* *haneteiru.*
became NO]-NOM garden-in is.hopping
'That frog which is the result of changing from a tadpole is hopping in the garden.'

I shall provide additional pieces of data. First, like FRs and unlike HIRs, CRs may be used to answer *wh*-questions asking about individuals. For instance, the *wh*-question *What is hopping in the garden?* may be properly answered by (51).

(51) [*Otamajyakushi-ga* *kaeru-ni* *natta*
[tadpole-NOM frog-COP became
no]-*ga* *haneteiru.*
NO]-NOM is.hopping
'A frog which is the result of changing from a tadpole is hopping in the garden.'

Second, like FRs but unlike HIRs, CRs may be at a focus position in clefts.

(52) [*Haneteiru* *no*]-*wa* [*otamajyakushi-ga*
[is.hopping NO]-TOP [tadpole-NOM
kaeru-ni *natta* *no*] *da.*
frog-COP became NO] COP
'It is [a frog which is the result of changing from a tadpole] that is hopping.'

Finally, like FRs but unlike HIRs, the Relevancy Condition is inert in CRs. For instance, (49) may be interpreted as: 'A tadpole became a frog 2 years ago and it is now hopping in the garden.'

These additional data corroborate Tonosaki's claim that CRs are more like FRs than HIRs. Given that the entry of *no* in Cann et al. (2005) models FRs (Seraku, in prep.), it is reasonable to assume that this entry of *no* applies to CRs (but not HIRs). More specifically, the parse of (49) up to *natta* yields a propositional content and the nominaliser *no* then picks out a term within the evaluated proposition and annotates a new type-e node with the term. This node is reflected into the propositional tree constructed by the matrix verb *haneteiru*. For details, see Seraku (in prep.).

[7] Contrary to our expectation, CRs do not have connotation when they denote humans (Tonosaki, 1998). In this respect, CRs behave more like HIRs. This is a residual problem.

7 Conclusion

This paper views Japanese HIRs as rich context-setters: the unit '*no* + case particle' encodes the procedures to map the tree of the HIR clause onto a partially-articulated tree. This partial tree is a 'rich' context against which the immediately embedding clause is processed. The partial tree contains two nodes:

- First, there is a situation node annotated with the relational predicate ☆. This provides a basis for modelling that only HIRs are subject to the Relevancy Condition.
- Second, there is an individual term decorated with the content of a head. This ensures that the head, though internal to the HIR clause, is licensed by the embedding verb.

This account predicts a range of HIR properties, including the data that would pose a problem for recent analyses of HIRs (e.g., maximality, island-insensitivity, negation, the locality restriction on the Relevancy Condition). It has also been shown that the nominaliser *no* (Cann et al., 2005) does not model HIRs but CRs. For additional sets of predictions, see Seraku (in prep.).

Acknowledgements

I am grateful to anonymous PACLIC reviewers for their helpful comments on the earlier version of the present paper. I would also like to thank Ruth Kempson and Jieun Kiaer for constructive exchange. All inadequacies are solely due to the author. This work is supported by the Sasakawa Fund Scholarship.

References

Blackburn, P. & Meyer-Viol, W. 1994. Linguistics, logic and finite trees. Bulletin of the IGPL 2: 3-31.

Cann, R. 2011. Towards an Account of the English Auxiliary System, In Kempson, R. et al. (Eds.) The Dynamics of Lexical Interfaces. CSLI, Stanford.

Cann, R., Kempson, R., and Marten, L. 2005. The Dynamics of Language. Elsevier, Oxford.

Davidson, D. 1967. The logical form of action sentences. In Rescher, N. (Ed.) The Logic of Decision and Actions. UPS, Pittsburgh.

Grosu, A. 2010. The status of the internally-headed relatives of Japanese/Korean within the typology of *definite* relatives. JEAL, 19: 231-274.

Grosu, A. & Landman, F. 2012. A quantificational disclosure approach to Japanese and Korean internally headed relatives. JEAL, 21: 159-196.

Hoshi, K. 1995. Structural and Interpretive Aspects of Head-Internal and Head-External Relative Clauses. Ph.D. dissertation, University of Rochester.

Kempson, R. & Kurosawa, A. 2009. At the syntax-pragmatics interface. In Hoshi, H. (Ed.) The Dynamics and Mechanism of Language. Kuroshio, Tokyo.

Kempson, R., Meyer-Viol, W., and Gabbay, D. 2001. Dynamic Syntax. Blackwell, Oxford.

Kim, M. J. 2007. Formal linking in internally headed relatives. NALS, 15: 279-315.

Kim, M. J. 2008a. Relevance of grammar and pragmatics to the Relevancy Condition. Language Research, 44: 95-120.

Kim, M. J. 2008b. Event Structure and Internally-headed Relative Clauses. VDN Verlag Dr. Mueller, Saarbrucken.

Kim, M. J. 2009. E-type anaphora and three types of *kes*-construction in Korean. NLLT, 27: 345-77.

Kitagawa, C. 2005. Typological variations of head-internal relatives in Japanese. Lingua, 115: 1243-76.

Kubota, Y. & Smith, E. A. 2007. The Japanese internally headed relative clause is not an E-type pronoun. In Miyamoto, Y. & Ochi, M. (Eds.) MITWPL 55. MIT Press, MA, Cambridge.

Kuroda, S.-Y. 1992. Japanese Syntax and Semantics. Kluwer, Dordrecht.

Kuroda, S.-Y. 2005. Nihongo-kara Mita Seisei Bunpo. (Generative Grammar from the Perspective of Japanese) Iwanami, Tokyo.

Kurosawa, A. 2003. On the Interaction of Syntax and Pragmatics. Ph.D. thesis, King's College London.

Matsuda, Y. 2002. Event sensitivity of head-internal relatives in Japanese. In Akatsuka, N. et al. (Eds.) Japanese/Korean Linguistics 10. CSLI, Stanford.

Mihara, K. 1994. Nihongo-no Togo Kozo. (Syntactic Structure of Japanese) Syohakusya, Tokyo.

Seraku, T. in prep. Clefts, Relatives, and Language Dynamics. D.Phil. thesis, University of Oxford.

Shimoyama, J. 1999. Internally headed relative clauses in Japanese and E-type anaphora. JEAL, 8: 147-82.

Shimoyama, J. 2001. *Wh*-Constructions in Japanese. Ph.D. dissertation, University of Massachusetts, Amherst.

Tonosaki, S. 1998. Change-relatives in Japanese. Journal of Japanese Linguistics, 16: 143-60.

Watanabe, A. 2003. *Wh* and operator constructions in Japanese. Lingua, 113: 519-58.