Computation with doubling constituents: Pronouns and antecedents in Phase Theory*

Sandiway Fong Jason Ginsburg
University of Arizona University of Aizu

Abstract

In the biolinguistics program, syntax has access to only limited and minimal operations such as Merge (and Move) plus local (to Phase) probe-goal, Chomsky (2000, 2001). Under this blueprint, the challenge for implementation is twofold: (1) to show that such a system operating locally can systematically derive complex facts, and (2) that computation, i.e. the selection of the appropriate operation is locally (and perhaps globally) efficient. To this end, available mechanisms may be prioritized to create an unambiguous instruction stream, e.g. Merge over Move. In this paper, we develop a computationally verified implementation of classic Binding Theory facts, including pronoun and anaphor asymmetries for mono- and multi-clausal sentences, possessive DPs, and picture DPs. We take as our starting point Chomsky's probe-goal system plus Kayne's (2002) doubling constituent proposal for pronoun-antecedent coreference relations. Within this framework, maintaining Phase locality between probe and goal forces independent licensing of an r-expression from its pronominal component via an operation of Last Resort theta Merge. We maintain that this Last Resort variant of Merge maintains the computational efficiency of the probe-goal system in that it operates precisely at the limit of probe-goal search domain and it does not introduce any additional choice points into the instruction stream.

1 Introduction

As part of the biolinguistics perspective, the Minimalist Program (MP) (Chomsky 1995) focuses on simple and optimal solutions to the problem of the nature of human language. It is expected that considerations of efficient computation (within the constraints of the biological system) should contribute to and help explain the shape or restrictions on the space of possible human languages (cf. Chomsky 2005). An uncontroversial property of

* Portions of this work were presented at the 2010 Western Conference on Linguistics at California State University, Fresno; the Linguistic Society of America 2011 Annual Meeting in Pittsburgh, Pennsylvania and at the 2011 Second Joint ASU/U of A Linguistic Symposium at Arizona State University. We thank the audiences for their helpful comments. We also would like to thank the two anonymous reviewers of this paper for their helpful comments.

this computational system is that it should be recursive, i.e., in principle allow the unbounded combination of a limited number of lexical primitives to form an infinite variety of specific structures that encode the variety of language phenomena. Within the strictures of the MP, it is proposed that simple recursive merge[1] (external and internal, i.e., displacement) suffices to generate an arbitrary number of structures, and the agree relation that obtains between probes and goals serves to restrict the possible instances (Chomsky 2001). Given that recursive merge can generate arbitrarily complex structures, efficient computation dictates that probe search be restricted in scope to cyclic domains known as phases. In this paper, we show that this computationally-motivated limit on search also serves to explain a variety of Binding Theory (Chomsky 1981, 1986) facts.

Much research has focused on the various restrictions that language imposes on coreference relations. These restrictions have traditionally been accounted for by Binding Theory, exemplified by the famous Conditions A-C (1a-c).

(1) (a) An anaphor must be bound in a local domain.
(b) A pronoun must be free in a local domain.
(c) An r-expression must be free. (Chomsky 1995:96)

These Binding Conditions accurately describe a wide variety of Binding phenomena. However, these Conditions have, since their formulation, been recognized to be inadequate to account for the variety of coreference phenomena found in language (e.g., see Reuland & Everaert 2001 and references cited within, among many others). In addition, while going a long way towards describing coreference phenomena, they do not explain why the phenomena are the way they are (e.g., why does Condition A hold?). Within the MP, in order to achieve a more adequate account of pronoun-antecedent relations and explain away the Binding Conditions, there have been some attempts to formulate Binding Theory in terms of movement (cf. Hornstein 2001, Kayne 2002, Zwart 2002, Heinat 2003). This paper builds on these analyses, particularly on the work by Kayne (2002). In this work, we describe a computer model that relies on independently motivated elements of Phase Theory (Chomsky 1999, 2000, 2001, 2004, 2006) to account for a variety of pronoun-antecedent relations in a computationally efficient manner.

[1] Henceforth we use “merge” to refer to Chomsky’s “Merge”.

The organization of this paper is as follows. Section 2 presents an overview of Kayne (2002), which forms the foundations of our analysis. Section 3 explains our proposals. Section 4 describes the computer model that we used to implement these proposals. Section 5 demonstrates how our model accounts for a variety of coreference facts, focusing on mono-clausal and multi-clausal constructions, possessive DPs and Picture DPs. Section 6 presents further evidence for our proposed doubling constituent structures. In section 7, we give our conclusions.

2 Doubling constituents

This paper builds on work by Kayne (2002) that assumes certain coreference relations result from movement of an r-expression out of a doubling constituent. Kayne, building on movement analyses of control phenomena (O'Neil 1995, 1997, Hornstein 1999, 2001) and Hornstein's (2001) movement analysis of certain pronoun antecedent relations, develops a doubling constituent analysis of pronoun antecedent relations. Kayne proposes that a pronominal element and an antecedent originate within a doubling constituent of the form [Spec Head], such as '[John he]', where the Spec is the antecedent and the head is the pronominal. The Spec can move out of a doubling constituent, but the head cannot. The head of the doubling constituent is licensed in its final surface position and thus has no need to move, whereas the Spec needs to move to obtain a theta-role and case. A crucial component of Kayne's analysis is that the Spec can only move out of a doubling constituent if the doubling constituent has undergone movement. In addition, a reflexive has a structure in which a doubling constituent moves, as in (2).

(2) [$_{DP}$ [John him] ~~[John him]~~ self]

These proposals derive some typical Condition A-C effects.

The notion that only a Spec can move out of a doubling constituent accounts for Condition C effects. An r-expression cannot be c-commanded by a pronominal element because the pronominal is the head of the doubling constituent. The head is licensed (gets a theta-role and case) in its surface position, whereas the Spec needs to move to get equivalently licensed. For example, the ill-formed (3a) cannot be derived, assuming the base structure in (3b). (3a) is ill-formed because the head of the doubling constituent 'he', not the Spec 'John', has been extracted. 'He' cannot move because then it would get a second theta-role.

(3) (a) *He$_i$ thinks John$_i$ is smart. (Kayne 2002:137)

(b) thinks [John he] is smart.

Certain Condition B effects are derived as follows. In (4a), with the derivation in (4b), the doubling constituent moves from its base position to the embedded [Spec, TP]. This frees the Spec 'John' for movement to theta-position. Note that 'he' remains free in the lower clause, thus satisfying Condition B.

(4) (a) John$_i$ thinks he$_i$ is smart. (Kayne 2002:146)

(b) [John thinks [$_{TP}$ [~~John~~ he] is smart ~~[John he]~~]]

On the other hand, in (5a), assuming the base structure in (5b), the doubling constituent has nowhere to move before the subject theta-position can be filled. Thus, the Spec cannot move out of the doubling constituent and the subject theta-role is not assigned.

(5) (a) *John$_i$ praises him$_i$. (Kayne 2002:146)

(b) [praises [John him]]

This analysis also accounts for why an anaphor can be local to its antecedent,[2] in accord with Condition A. In the derivation of (6a), as shown in (6b), the doubling constituent moves within the anaphor, since Kayne assumes that a reflexive has a position within it to which a doubling constituent moves (see (2) above). This movement somehow frees the Spec 'John' for movement to theta-position. Thus, the anaphor is bound locally.

(6) (a) John$_i$ praises himself$_i$.

(b) [John praises [[~~John~~ him] ~~[John him]~~ self]]

Kayne's system, however, faces some problems. First of all, a crucial component of Kayne's analysis is the requirement that a doubling constituent move in order for the Spec to be extracted. This accounts for the Condition A-C effects in examples (3-6). However, it is not clear why the possibility of extraction of the Spec of the doubling constituent is dependent on movement of the doubling constituent.

In addition, there are some basic data that are problematic for Kayne's analysis. Kayne's analysis appears to predict the opposite grammaticality judgments for (7) and (8). Example (7a) is well-formed, indicating that the Spec 'John' has moved out of the doubling constituent. However, as shown in (7b), there does not appear to be any

---

2 Note that Kayne's analysis does not account for why an anaphor must generally be local to its antecedent. See example (8).

position that the doubling constituent can move to so that the Spec 'John' can be extracted.

(7) (a) $John_i$ thinks that Mary likes him_i.

(b) [$_{v*P}$ …v* thinks [$_{CP}$ that Mary T [$_{v*P}$ ~~Mary~~ likes [John him]]]]

Consider (8). Kayne's analysis predicts that it should be possible for 'John' to move out of the doubling constituent and into theta-position, since the doubling constituent moves within the anaphor, as shown in (8b). But the ill-formedness suggests that this is not the case.

(8) (a) *$John_i$ thinks that Mary likes $himself_i$.

(b) [$_{v*P}$ John v* thinks [$_{CP}$ that Mary T [$_{v*P}$ ~~Mary~~ likes [$_{DP}$ [~~John~~ him] ~~[John him]~~ self]]]]

A possible explanation for the ill-formedness of (8) is that *Mary* blocks movement of John, but if that were the case, then it is not clear why *Mary* would not block movement of John in (7). Note that Kayne's analysis thus fails to account for why an anaphor must generally be local to its antecedent.

Next, consider the ECM construction (9a). In the partial derivation (9b), the doubling constituent moves within the lower clause and then to the higher clause. This movement should free the Spec 'John' for movement to theta-position, thus predicting, contrary to fact, that (9a) should be well-formed.

(9) (a) *$John_i$ considers him_i to be intelligent. (Kayne 2002:146)

(b) [$_{v*P}$ John considers [~~John~~ him] [$_{TP}$ ~~[John him]~~ to be [intelligent ~~[John him]~~]]

Kayne (2002:146) explains the ill-formedness of (9a) as follows:

> … "raising to object" must apply first and [. . .] once it does [John him] is too high in the structure for there to be any available intermediate position above it, yet below the subject theta-position of *consider*.

However, it is not clear why raising to the object position of an ECM verb, as well as raising to the subject position of the embedded clause, should not count as movement that frees the Spec of a doubling constituent.

Another problem for Kayne's analysis can be found with possessive DPs. In the well-formed possessive (10a), it is not clear where the doubling constituent can move to in order to enable extraction of the subject 'John'.

(10) (a) $John_i$ likes his_i dog.

(b) likes [John he]'s dog

On the other hand, Kayne's analysis predicts that the ill-formed (11a) should be well-formed, since the doubling constituent can move within the anaphor, thereby allowing 'John' to move to subject theta-position.

(11) (a) *John$_i$ likes himself$_i$'s dog.

(b) John likes [$_{DP}$ [~~John~~ him] ~~[John him]~~ self] 's dog

In this paper, we present a new proposal of these pronoun-antecedent facts that adopts a version of Kayne's doubling constituent proposal. Also see Zwart (2002)[3] and Heinat (2003)[4] for related analyses. Our proposal, however, predicts data that Kayne's

---

[3] Zwart (2002) assumes (following Kayne 2002), that coreference is the result of merge of an antecedent and a variable referential element. Zwart (2002:274) writes, "A pronoun α is coreferential with β iff α is merged with β." Unlike in Kayne's analysis, though, the pronominal element that originates in a doubling constituent must be an anaphor, and all other forms of pronoun-antecedent coreference are considered to be accidental. On the one hand, this analysis does away with the troubling requirement that a doubling constituent move in order for the Spec to be extracted, but on the other hand, it does not provide a principled account of pronoun coreference, which Zwart suggests is subject to certain pragmatic constraints. For example, it is not clear how to account for traditional Condition C effects. Zwart points out that (ia) is ill-formed because if 'John' and 'him' originate in the doubling constituent '[John him]', then it must be Spelled out as shown in (ib).

(i) (a) John$_i$ loves him$_i$.

(b) John$_i$ loves himself$_i$.

But then it is not clear why (i)(a) cannot result from accidental coreference, as in (ii).

(ii) John$_i$ thinks that Mary loves him$_i$.

[4] Heinat (2003) presents a revised version of the doubling constituent proposal in which he too assumes base generation of a doubling constituent and he does not follow Kayne's requirement that a doubling constituent move. Rather, the form of a pronominal (whether it is pronounced as a pronoun or anaphor) depends on whether or not it is sent to Spell-Out at the same time as its antecedent. Heinat's proposal relies on the Phase Theory view that when the edge of a phase is reached, the complement of the head of the phase is sent to Spell-Out (Chomsky 2000, 2001). When the antecedent and the pronominal, which Heinat refers to as a PRONOUN, are sent to Spell-Out simultaneously, the PRONOUN is pronounced as an anaphor. When the PRONOUN is sent to Spell-Out before the antecedent, the PRONOUN is pronounced as a pronoun. One potential problem with this proposal is that it requires there to be object

analysis can account for, as well as data that are problematic for Kayne's analysis, without the stipulation that a doubling constituent move, and in a computationally efficient manner. We next turn to our proposals.

3 Proposals

We follow Kayne (2002) in assuming that a pronoun and antecedent originate as a doubling constituent, and we also follow Kayne's idea that constraints on movement out of a doubling constituent account for pronoun-antecedent facts. However, we differ from Kayne with respect to the structure of a doubling constituent and with respect to how movement occurs out of a doubling constituent.

We assume a view of Phase Theory whereby a derivation is broken up into phases that are formed via selection and merge of Lexical Items (LIs) from a numeration. Strong phase heads are C (complementizer), v* (little *v*) and certain D (determiner) heads (*self* or possessive *'s*). We propose a revised view of the Phase Impenetrability

---

shift in an English construction with an anaphor object. If (ia) has the underlying structure in (ib), then the PRONOUN is sent to Spell-Out when the v*P edge is reached, and 'Mary' will be sent to Spell-Out later - thus the PRONOUN should be pronounced as 'her' contrary to fact.

(i) (a) Mary likes herself. (Heinat 2003)

(b)[$_{CP}$ Mary[$_{v*P}$ Mary likes [Mary PRONOUN]]]

To get around this problem, Heinat takes the position that the doubling constituent undergoes object shift to the v*P edge, as in (ii) below.

(ii) [$_{CP}$ Mary likes [$_{vP}$[[Mary$_{copy}$] PRONOUN] [$_{vP}$… (adapted from Heinat 2003)

But this means that the verb must also move out of the vP or else the object 'herself' would precede 'likes'. Yet, it is not clear where the verb would move to. Furthermore, it is not clear that there is object shift in this construction. Heinat presents Chomsky's (2001) claim that object shift can occur in English, as in (iii).

(iii) (guess) what$_{Obj}$ [John$_{Subj}$ T [vP t$_{Obj}$ [t$_{Subj}$ read t$_{Obj}$ Obj]]] (Chomsky 2001:26)

However, Chomsky claims that when there is object shift in languages like English "the object must move on beyond the position of OS [Object Shift] (Chomsky 2001:26)." In (iii), Chomsky claims that the object moves to [Spec, CP]. Furthermore, even if there is object shift in a relative clause such as (iii), it is not clear that there is object shift in a simple transitive clause such as (i).

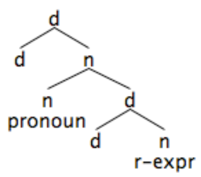
Condition (PIC). In the view of the PIC in Chomsky (2004), when a phase head is merged, the complement of a lower phase head is sent to Spell-Out. In our view, when a phase head is merged, an entire lower phase (not just the complement of the phase head), if present, is closed off, where "closing off" refers to a domain that becomes unaccessible. In (12), when the phase head C is merged, the lower v*P phase is closed-off and becomes unaccessible to further syntactic operations.

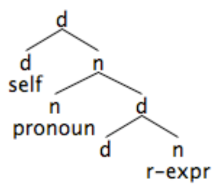
(12) [$_{CP}$ C T [$_{vP}$ v* V …]]

This closing off results from the need to restrict the domain of probe search. Beyond two phases, memory limitations make search inefficient. We demonstrate that, for the constructions discussed in this paper, there is no need to access phase edges.

We propose the structures in (13a-b), where (13a) is a doubling constituent consisting of a pronoun and antecedent r-expression ('r-expr') and (13b) is an anaphor and antecedent r-expression.

(13) (a) pronoun and r-expresssion (b) anaphor and r-expression





In (13a), the pronoun (e.g., 'he/him') is a noun with an r-expression DP complement (e.g., 'John'). We propose that a DP anaphor, (13b), is a (strong) phase DP with the D phase head 'self'. Morphological affixation between 'self' and a pronoun results in a reflexive; for example, 'self' combined with 'him' results in 'himself' at Spell-Out. The DP in (13a) lacks 'self' and is not a (strong) phase.[5]

Kayne proposes that a doubling constituent moves within an anaphor, as shown in (2), repeated below, but it is not clear where the doubling constituent moves to or why it moves within an anaphor.

(14) [$_{DP}$ [John him] ~~[John him]~~ self]

---

[5] Chomsky (2006), following work by Svenonious (2004) and Hiraiwa (2005), writes that DPs might also be phases.

Our structure of an anaphor (13b) is clearer. There is no movement within the anaphor doubling constituent, nor is there any requirement that movement occur in order for an r-expression to be extracted.

In addition, we incorporate the idea that the grammar makes available an operation of Last Resort. Last Resort (Chomsky 1995) was originally formulated as an operation that requires movement to involve a feature checking operation. Our version of Last Resort also involves feature checking. Crucially, though, we propose that Last Resort arises when an LI that needs to undergo a feature checking relation is in danger of falling outside the scope of a probe - this condition arises when the LI is contained within a phase that is about to be closed off to higher operations. Last Resort (LR) is defined as follows.

(15) Last Resort (LR): an LI with an unvalued feature that is in imminent danger of falling outside of a probe-goal scope relation can:

(a) preferentially undergo internal merge into an available theta-position

(b) or, go into a buffer for later external merge,

(c) external merge to a theta-position has priority over Last Resort

If an unlicensed LI remains in a closed off domain, then a derivation will crash. To avoid this non-optimal outcome, the human grammar module makes available LR as an escape mechanism. The LI, if possible, is theta-merged (assume that the LI is a DP) with the head of a tree (15a). This is the most optimal operation, since the LI is reused immediately. If that is not possible, which is the case if the head of the tree does not select for a theta-position, then the LI will be set aside for further merge. The grammar has a special buffer that an LI can be placed in (15b), as a Last Resort when merge into a theta-position is not available. From the buffer, the LI can later be selected and merged into a derivation. There is a catch though - this LR operation is blocked by external theta-merge (15c); if external merge into theta-position of a phrase from the numeration is possible, then LR is blocked. This arises out of the need for efficient computation - external merge is more economical than the LR movement process.

The phasehood distinction between the two types of doubling constituents, the non-phase pronoun-antecedent and phase anaphor-antecedent (13a-b) plays an important role in our analysis. Phases are a consequence of minimal computation. Chomsky (2006:143) writes:

> For minimal computation, as soon as the information is transferred it will be forgotten, not accessed in subsequent stages of derivation: the computation will not have to look back at earlier phases as it proceeds, and cyclicity is preserved in a very strong sense.

Long distance, i.e., unbounded, dependencies borne via displacement from within doubling constituents, therefore require some form of "escape hatch" from phase domains. In Chomsky's proposals, (repeated) displacement to the edge of a phase is used for this purpose. By contrast, in our system, the LR operation functions as an escape hatch, and, via the buffer, it allows long distance dependencies.

Consider (16a-b), in which the LR process, combined with our view that a phase is closed-off when a higher phase head is merged, accounts for the possibility of remerge. '[F:]' represents an unvalued feature of an r-expression; assume that the r-expression lacks case and/or a theta-role.

(16) (a) $[_{v*}$ v*…$[_{d}$ self …r-expr$_{[F:_]}$ …]]

(b) $[_{c}$ C …$[_{v*}$ v* …r-expr$_{[F:_]}$ ...]]

In (16a), 'self' and v* are phase heads. When v* is merged, the lower DP phase will be closed off. Since the r-expression contains an unvalued feature, it is subject to the LR process and it will be immediately remerged with v*; that is, if external theta-merge is not an option (i.e., there is no other DP available within the numeration). In (16b), both C and v* are phase heads. When C is merged, the lower v*P will be closed off. In this case, the r-expression thus is subject to the LR process. Note, though, that C does not select for an argument; theta-merge with C is not possible. Therefore, the r-expression will go directly into a buffer, from which it can later be remerged into the derivation.

The computational system has a choice when it comes to selecting between theta merge operations: (a) external merge, (b) the LR operation of immediate remerge, and (c) the LR operation of insertion into a buffer. Our proposals, namely options (b) and (c), do not compromise the optimality of Chomsky's system: in other words, theta merge options are managed such that these options are in complementary distribution, and extra choice points (that weaken the optimality of derivations) are never introduced. Furthermore, the order of preference between these options is directly tied by computational efficiency concerns. The first option (a), external merge directly into theta position, is a primitive of Chomsky's system, and thus the simplest possible operation. It is therefore preferred (when available) over the two remaining possibilities.[6] The second option, (b), a direct theta merge operation after search to the limit of the probe mechanism, is computationally more complex than option (a), and thus is dispreferred. The last option, (c), namely storage into a special buffer for delayed theta merge, still achieves theta merge but is more complex still. It involves additional "hardware" (i.e., the buffer) and multiple copy operations (i.e., copy into and out of the buffer).

The LR process, in addition, has the advantage of increasing computational efficiency because, at least with respect to the examples discussed in this paper, it eliminates the need to (a) move an element to a phase edge, and (b) move an element through an intermediate position. Consider (17).

(17) (a) [$_{c}$ C …[$_{v*}$ v*…r-expr$_{[F:_]}$ ...]]
(b) LR insertion of r-expr$_{[F:_]}$ into buffer
(c) [$_{v*}$ r-expr$_{[F:_]}$ v*...[$_{c}$ C … [$_{v*}$ v*…~~r-expr$_{[F:_]}$~~...]]]

When C is merged (17a), assume that the lower v*P is sent to Spell-Out. Since the r-expression is not yet licensed, as it lacks case and a theta-role, the LR process is triggered. Since the r-expression cannot be immediately remerged, as it is not selected for by C, it is inserted into the buffer (17b). Then when the matrix v* is merged, the r-expression is selected and remerged in matrix subject position (17c), after which its unvalued features can be checked. Crucially, the r-expression never moves to the edge of any phase. Nor is it ever merged in an intermediate position - it does not adjoin to the embedded CP edge. Under the standard Phase Theory account, it would have to move through each intervening phase edge. However, it is more computationally efficient to avoid these extra remerge operations and simply remerge the r-expression directly in subject theta-position. Our proposals enable this more optimal solution. If our analysis is on the right track, it suggests that there may not be any need to use phase edges as escape hatches, a view which could lead to a simplification of Phase Theory. We leave for further analysis investigation of whether or not there is a need to access phase edges in derivations in general. We demonstrate in this paper that under the appropriate circumstances, LR allows a derivation to converge. Under other circumstances, an LI with an unvalued feature is stranded in a closed-off phase and a derivation crashes.

---

6 See (35a) and (44) below for examples where direct theta merge blocking LR is crucial to our analysis.

4 The Computer Model

We implemented our proposals via a computer model. Computer modeling of generative grammar has many useful aspects. One is verification: a working computer model that faithfully replicates a theory can help to demonstrate its coverage, consistency and coherence. Another is formalization: if pinned down in sufficient detail, this may permit the mathematical modeling of core theory properties, such as formal complexity and descriptive power. Stabler (1997) is an example of the latter; the Minimalist Grammar (MG) framework allows one to build simple yet powerful systems using just categorial selection and ±feature matching.[7] The computer model implemented for this paper is an example of the former aspect, verification. Our computer model faithfully replicates all aspects of the linguistic theory described. In terms of coverage, we attest that all examples described in the following sections of this paper have been verified. However, we have not attempted to mathematically model the computational properties of our theory.

Since linguistic theories are not complete down to every last detail, gaps will exist between the theoretical "blueprint" and a realized computer implementation. It is important to draw a line between theoretical commitments and the necessary pinning down of irrelevant details for the sake of a working computer program. In other words, there will always be features of the computer model to which no theoretical import should be attached. Moreover, algorithmically speaking, there are always several ways to concretize an abstract theory. With these caveats in mind, what follows is a brief inventory of the implementation-based assumptions that underpin our computer model.

Basic phrase structure is determined by categorial selection. As is typically assumed in the Minimalist literature, we stipulate that complementizer (c) selects for

---

[7] We note here that the MG framework can model mildly context-sensitive grammars. It also permits the direct encoding of the feature matching portion of Chomsky's MP (Chomsky 1995), but does not appear to contain the apparatus necessary to encode further developments needed for the theory described in this paper: e.g. probe-goal search and computational reflexes when search limits are reached. In the case of Binding Theory, reduction of these mechanisms to pure feature matching would be of architectural interest but requires empirical demonstration; however, we are unaware of any such efforts in the MG framework.

tense (t). Tense selects for little v (v/v*). Little v selects for verb V (or perhaps another predicate such as an adjectival phrase). V may select for a variety of possible objects if the verb permits an object, e.g. determiner phrase (d) or propositional complement (c or t). In this implementation, selection is encoded using a Definite Clause Grammar (DCG) of the (simplified) form shown in (18).

(18)

c --> c, t.	d --> d, n.
t --> d, t.	n --> n, d.
t --> t, {search(g), copy(g)}.	c --> [that, empty complementizer].
t --> t, v, {search(g), agree(t,g)}.	t --> [t-complete, t-defective].
v --> v, V {search(g), agree(v,g)}.	n --> […lexical nouns…].
v --> V.	d --> [the, a, self].
V --> V, d.	V --> [think, like, …].

The DCG permits a simple (top-down, left-to-right) recursive descent execution strategy. However, the implementation is configured to construct bare phrase structure in a bottom-up (left-to-right) order.[8] Since a DCG is employed, the numeration is effectively constrained to be an ordered multiset. In other words, we assume heads are streamed to the DCG in the (correct) underlying left-to-right order.

In Chomsky's MP, displacement does not create new features (e.g. indices) or distinct phrases (e.g. traces). Our Binding model also does not introduce or rely on any external "marks" to indicate coreferentiality; instead, pronoun-antecedent identification is "baked" into the doubling constituent model. Therefore, implementation of displacement involves just a pure copy operation. For example, the specifier of tense may either be filled by a (pleonastic) noun directly from the stream or from a (phase-limited) search for a noun (phrase) in its complement domain (e.g. a sentential subject in specifier of v*).[9] In the case where search (search(g), g a goal) has returned a suitable candidate, the implementation simply copies it into the specifier of tense. By

8 This bottom-up order is reported in the screenshots shown in this paper.

9 The first option is encoded by the t --> d, t. rule. The second by t --> t, {search(g), copy(g)}.

the no-tampering condition (Chomsky 2006), the complement domain remains unmodified.

Agree is implemented at the DCG rules for tense and little v. The probe-goal search operation (search(g)) obtains a goal (g) and the operation agree(head,g) implements uninterpretable/interpretable feature agreement between the head (of tense or little v) and the matched goal (g). If no suitable goal is found, the rule fails to complete and other choices may be explored.

With respect to non-determinism, we should distinguish theory choice points imposed by the linguistic theory and computational choice points introduced as artifacts of the implementation. Architecturally, an ideal implementation will introduce no computational choices of its own. More precisely, given an instruction stream of heads, assembly into a complete phrase should proceed deterministically without backtracking or multiple threading. Our choice of an ordered multiset does not fully implement Chomsky's original (non-deterministic) numeration; e.g., *John likes Mary* and *Mary likes John* are both possible outcomes of the same numeration in Chomsky's model; in our model, these two sentences would be generated by different orderings of the same numeration. Similarly, our implementation does not allow for a free choice of antecedents for a pronoun from a given numeration; the assembly of the doubling constituent is fixed by stream order. In terms of computational choice points, a recursive descent interpretation of the DCG is intrinsically non-deterministic.[10] Different models could be employed to reduce local non-determinism introduced by rule choice points.[11]

This computer model incorporates the LR process, repeated below, in the following manner.

---

10 The execution model permits multiple rules for a given non-terminal. The order in which they are tried is simply in the order in which they are written.

11 This discussion could extend way beyond the scope of this paper. Many implementation strategies from the formal language and compiler theory literature could be employed. For example, a bottom-up, shift-reduce grammar rule model could replace the recursive descent model. Traditional ambiguity-reducing optimizations such as left-corner or lookahead sets are also possible choice point reduction strategies.

(19) Last Resort (LR): an LI with an unvalued feature that is in imminent danger of falling outside of a probe-goal scope relation can:

(a) preferentially undergo internal merge into an available theta-position, or

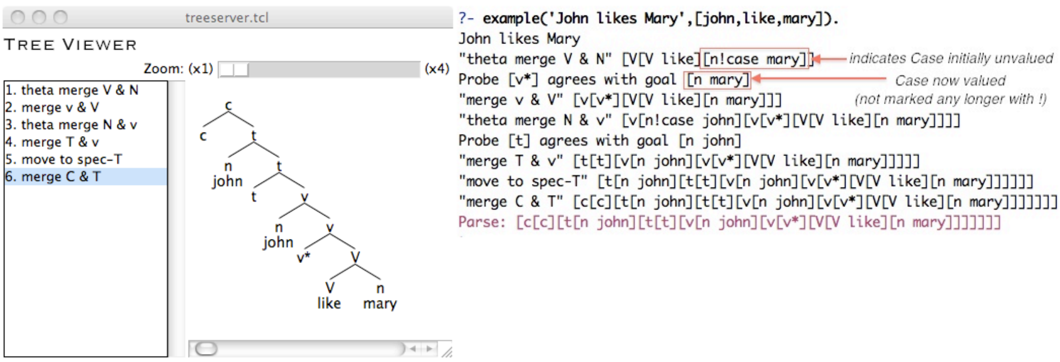
(b) go into a buffer for later external merge,

(c) external merge to a theta-position has priority over Last Resort

Algorithmically speaking, when a phase edge is reached, there is a Last Resort search, skipping past one phase, into a lower phase. This component models the PIC - there is a search into the next lower phase because that entire phase will be closed off. If no lower phase is present, the Last Resort search process fails. If a lower phase is present, and there is an LI in that phase that has an unvalued feature, that LI is selected. Then, if theta-Merge with the label of the tree is possible, the LI is immediately remerged into the derivation. Otherwise, the LI is inserted directly into a buffer, where it remains until it can be remerged, if possible.

Our model automatically constructs, as described above, the derivation of a sentence from a numeration that it is fed. Example (20) is a screenshot of the derivation of the simple sentence 'John likes Mary'. The left side of the Tree Viewer lists each step of the derivation. The middle of the tree viewer displays tree diagrams of each step of the derivation. The right side shows each step of the derivation in bracket format. Initially, case on 'Mary' and 'John' is unvalued, as represented with 'n!case'. When case becomes valued, the 'n!case' disappears - it is valued and eliminated. V merges with 'Mary'. Then v* merges and assigns 'Mary' case. This is followed by theta merge of 'John', etc.

(20)



We next demonstrate how this model constructs the derivations of sentences with pronoun- antecedent relations.

5 Derivations

In this section, we demonstrate how the LR process accounts for a variety of pronoun-antecedent data. We examine mono-clausal constructions, bi-clausal constructions, possessive DPs and picture DPs.

5.1 Pronoun-antecedent relations in mono-clausal constructions

First of all, we account for the complementary distribution of pronouns and anaphors in local relations, as in (21a-b), originally presented as (5a) and (6a) above.

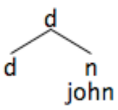
(21) (a) *John$_i$ praises him$_i$.

(b) John$_i$ praises himself$_i$.

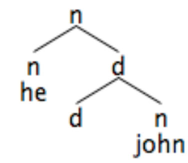
The distinction between these examples results from the (im)possibility of the r-expression ‘John’ to undergo the LR process, which depends on whether or not the r-expression is base-generated within a DP phase.

The derivation of the ill-formed (21a) proceeds as show in (22a-e), which are screenshots of the derivation as it is constructed automatically by our model.

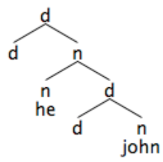
(22) (a)



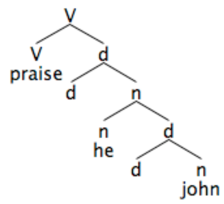
(b)



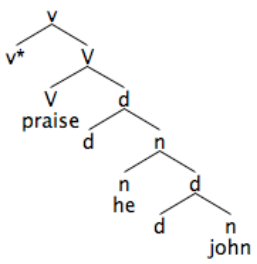
(c)



(d)



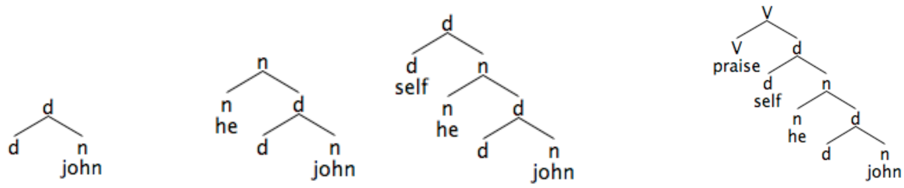
(e)



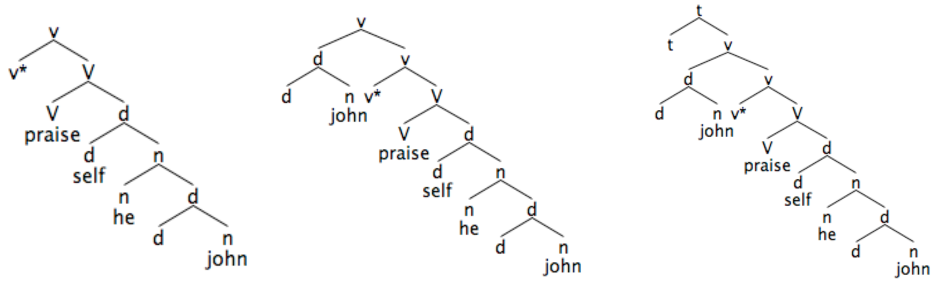
Initially, D and ‘John’ are merged (22a). This is followed by merge of the pronoun ‘he’ with the DP (22b), and then merge of a higher D, to form the complete doubling constituent (22c). Next, the V ‘praise’ is merged (22d), followed by v* (22e). When v* is merged, there is no lower phase present and thus no phase is closed off. At this point, the DP ‘John’ has unvalued features, since it lacks case and a theta role (it is not licensed). However, since it is not contained within a phase that is about to be closed of, it cannot undergo the LR process. As a result, there is no DP available for merge with v* and the derivation crashes. Thus, we are able to account for the inability of a pronoun to be bound locally - a Condition B effect.

The derivation of the well-formed (21b) proceeds as in (23a-i). Crucially, the doubling constituent in this example corresponds to an anaphor and antecedent.

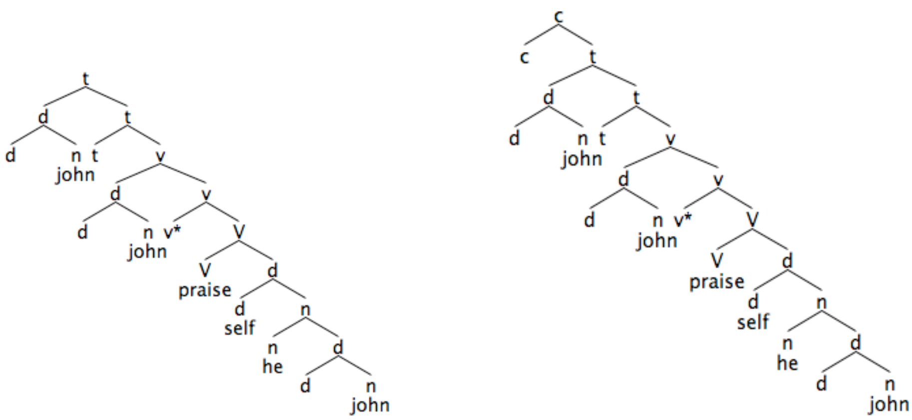
(23) (a) (b) (c) (d)



(e) (f) (g)



(h) (i)



The doubling constituent is formed in steps (23a-c), and forms a DP phase after merge of the D ‘self’. The V ‘praise’ is merged (23d), followed by merge of v* (23e). Crucially, the DP ‘John’ lacks case and a theta-role. Thus, when v* is merged, since ‘John’ is contained within a lower DP phase that is about to be closed off, and ‘John’ contains unvalued features, it undergoes the LR process. Since v* selects for a subject DP and there is no LI available for external merge, 'John' is immediately remerged with v*, thereby landing in subject theta-position (23f). This is followed by merge of T (23g). Then 'John' undergoes EPP driven movement to [Spec, T] (23h). Lastly, C is merged (23i) and the derivation converges successfully. We thus account for a Condition A effect, since the anaphor is bound locally.

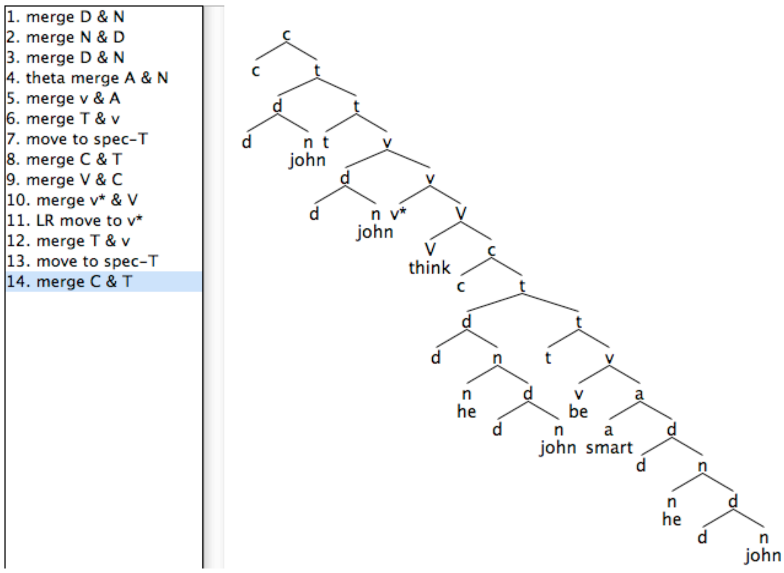
5.2 Pronoun-antecedent relations in multi-clausal constructions

We next turn to pronoun-antecedent relations in multi-clausal constructions. Our analysis accounts for why a pronoun and antecedent can co-occur if they are in separate clauses. Consider (4a), repeated below.

(24) John$_i$ thinks he$_i$ is smart.

The derivation for this example is shown in (25). The left side of the screenshot lists each step of the derivation.

(25)



The doubling constituent '[he John]' is base generated within the adjectival predicate. Then it undergoes EPP-driven movement to subject position of the embedded clause. When the matrix v* is merged, crucially, the lower CP phase must be closed-off. Thus, 'John', which contains unvalued features (it lacks a theta-role and case), undergoes the LR reinsertion process and it is immediately remerged in subject-theta position in the matrix [Spec, v*]; v* selects for DP and there is no other DP available for external merge.

Our analysis also straightforwardly accounts for the impossibility of a pronoun binding an r-expression as in (3a), repeated below.

(26) *He$_i$ thinks John$_i$ is smart.

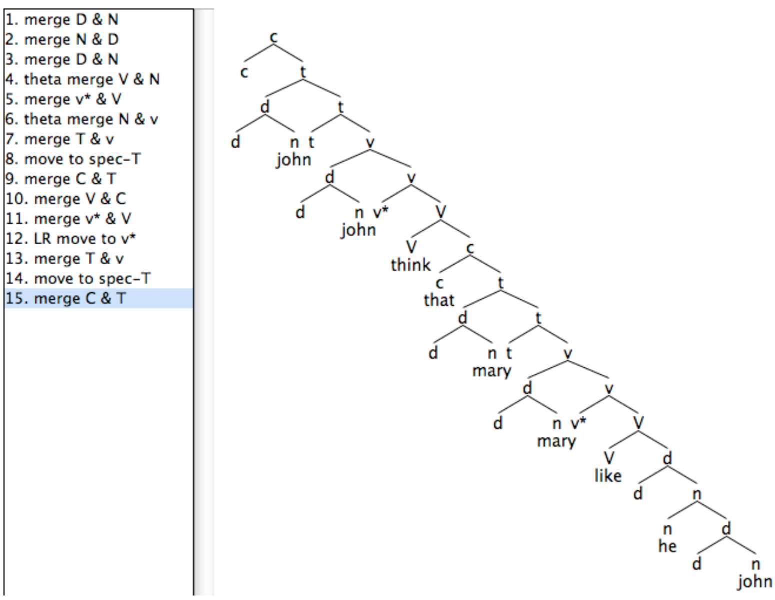
Following Kayne's proposal that the head of a doubling constituent must be a pronoun (see our proposed structure in (13a)), (26) is underiveable. The doubling constituent must have the form (simplified) '[he John]', with the head 'he', and thus, it can only result in the LR movement of 'John', thereby producing example (24) above. Thus, the impossibility of binding an r-expression in this example, a Condition C effect, is accounted for.

In addition, we account for the possibility of long-distance co-reference relations (Condition B effects). Consider (7a), repeated below, which as noted in section 2, is problematic for Kayne's analysis.

(27) John$_i$ thinks that Mary likes him$_i$.

The derivation is shown in (28).

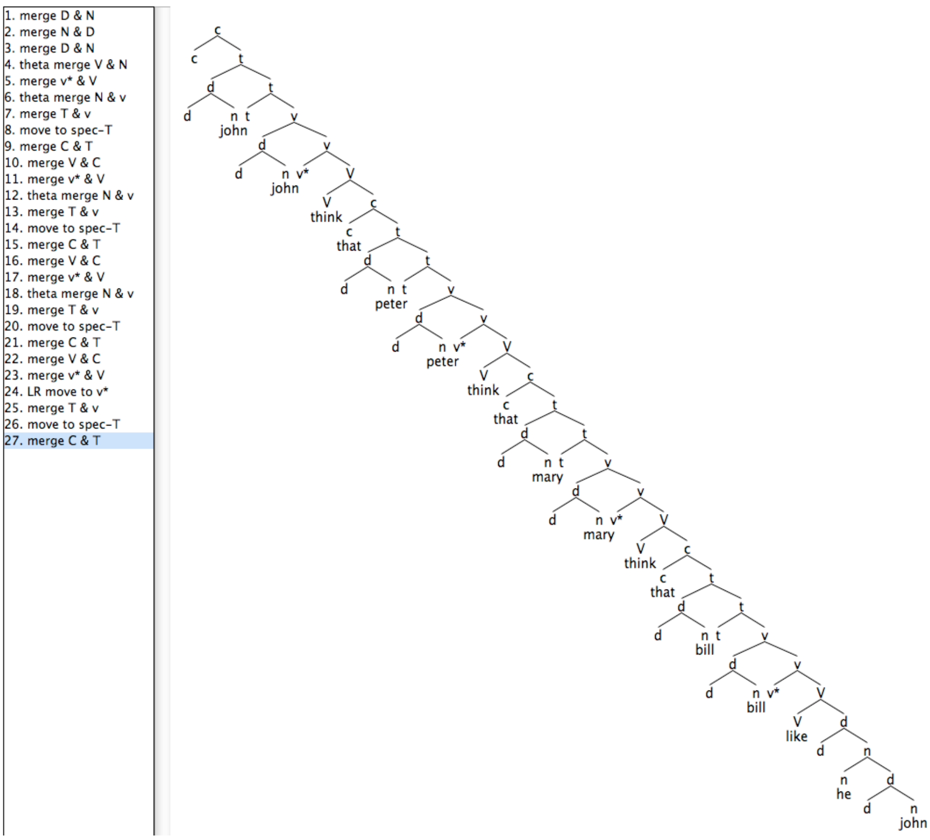
(28)



When the lower v* is merged, there is no lower phase present, and thus 'John' cannot undergo the LR process. When the embedded C is merged, the lower v*P will be closed off. Thus, at this point, 'John', which contains unvalued features, is subject to the LR process. However, the head of the tree at this point, C, does not permit theta-merge. As a result, 'John' cannot be used immediately and be remerged into the derivation. Therefore, 'John' is inserted into a buffer. When the matrix v* is merged, 'John' is selected and merged in subject theta-position, and the derivation converges successfully. Long distance coreference, as in (29), is accounted for in a similar manner, as shown in (30).

(29) John$_i$ thinks that Peter thinks that Mary thinks that Bill likes him$_i$.

(30)



In this construction, when the most embedded C is merged, the lower v*P will be closed off. Thus, 'John' undergoes the LR process. Again, since C does not permit theta-merge, 'John' goes into the buffer and then is remerged in matrix subject position. This raises the issue of whether or not an r-expression in a buffer must be merged in matrix subject position.

We suggest that the following holds, with respect to an LI in a buffer.

(31) An LI in a buffer is merged in the highest possible position.

Once a DP is inserted into a buffer, it can potentially be inserted into any theta-position. However, there is a preference for merge in the matrix subject position. (32), in which the r-expression 'Peter', which is coindexed with the pronoun, does not appear in the matrix subject position is marginal.

(32) (?*)John thinks that $Peter_i$ thinks that Mary thinks that Bill likes him_i.

We suggest that this preference for a matrix subject is motivated by parsing considerations. The generator, "aware" of parsing considerations, prefers the highest subject. Parsing and generation have different computational starting points (a surface sequence vs. an unordered lexical array) and therefore must rely on different mechanisms (e.g. gap-filling vs. upwards movement). We propose that parsing and generation are not completely independently-derived mechanisms: in fact, there must be some "co-awareness" between them in terms of computational strategy for efficiency. As numerous studies in the psycholinguistics literature have shown, top-down prediction and left-to-right expectation are properties of parsing that have demonstrable impact; e.g., preferences for subject vs. object gap-filling (e.g., see King & Kutas 1995). Top-down prediction logically implies that the matrix subject position will be proposed and filled first from the surface input. That matrix subject must be "remembered" or carried along as parsing proceeds from higher to lower phrases, possibly into lower clauses, until its "gap" or originating theta position is found.[12] (Assuming the framework proposed in this paper, that gap may also occur within a DP doubling constituent headed by a pronoun.) This situation requires working memory. Let us call this working memory a "buffer". We must also deal with the possibility of predicting multiple gaps as parsing proceeds to fill in other open positions from the input, as an additional computational burden, but the matrix subject is the first hypothesized "passenger" of the parsing process.

We propose that generation may also make use of this working memory; in other words, co-opt this buffer resource and respect its preferences. Hence, there is a matrix subject preference when items are placed into the buffer. However, if the LI in the buffer cannot be merged in matrix subject position (e.g., there is a gender mismatch) then it will be merged in a lower position, if possible.

As pointed out by an anonymous reviewer, (33) is fine.

(33) Lucy said that $Peter_i$ thinks that Mary likes him_i.

[12] We put aside here the case of direct injection of a pleonastic, as in *It rains*.

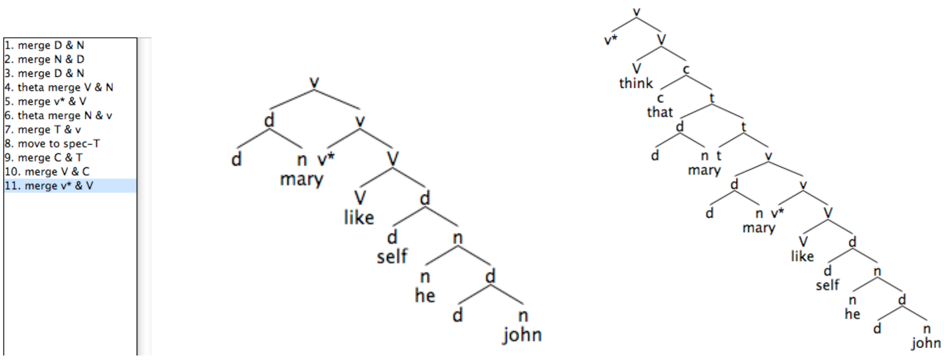
Crucially, due to the gender mismatch between the matrix and higher embedded subject, this does not create a problem for parsing. Since it does not create a problem for parsing, it does not create a problem for generation. In this manner, merge of a subject from the buffer into a non-matrix position is possible if it does not cause a problem for parsing.

We also are able to account for the impossibility of long distance binding of an anaphor, as in (8a), repeated below, which we noted is problematic for Kayne's analysis.

(34) *John$_i$ thinks that Mary likes himself$_i$.

The relevant parts of the derivation are shown in (35a-b).

(35) (a) (b)



The doubling constituent crucially originates within a DP phase. When v* is merged, the lower DP will be closed off. The DP 'John' contains unvalued features. However, in this case, the direct external merge of the subject 'Mary' blocks the LR process, since external merge is a simpler and more efficient operation then remerge of 'John' in a theta-position or insertion of 'John' into a buffer.[13] As a result, 'John' remains in-situ, and 'Mary' is merged in subject position (35a). Then, when the matrix v* is merged (35b), there is no subject available and the derivation crashes. The impossibility of local binding of an anaphor in this construction, a Condition A effect, thus falls out of our analysis.

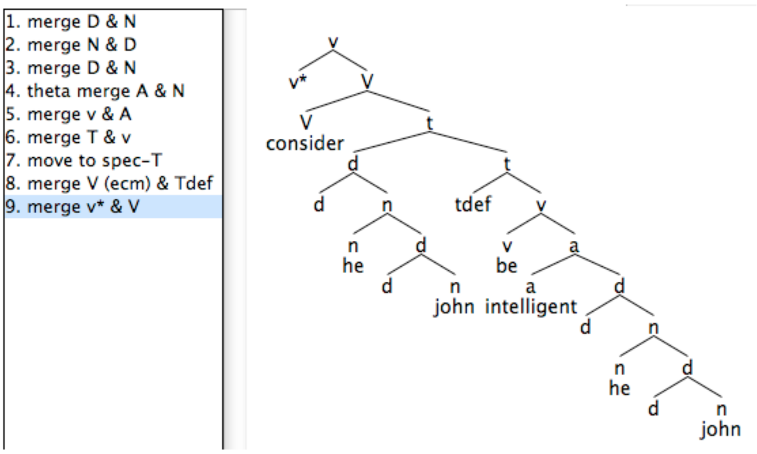
ECM constructions, which are problematic for Kayne's analysis, also are accounted for. Consider (9a), repeated below.

(36) *John $_i$ considers him $_i$ to be intelligent.

13 This blocking action is crucial. For example, if the buffer is available to 'John', the illicit derivation will go through.

The relevant part of the derivation is shown in (37). The doubling constituent raises from within the adjectival predicate to [Spec, T] of the embedded clause. Assuming that an ECM lacks a CP (Chomsky 1981), when the matrix v* is merged, the lower TP, which contains the doubling constituent, is not a phase, and thus will not be closed off. As a result, 'John' is unable to undergo the LR process, leaving no matrix subject available, and causing the derivation to crash.

(37)

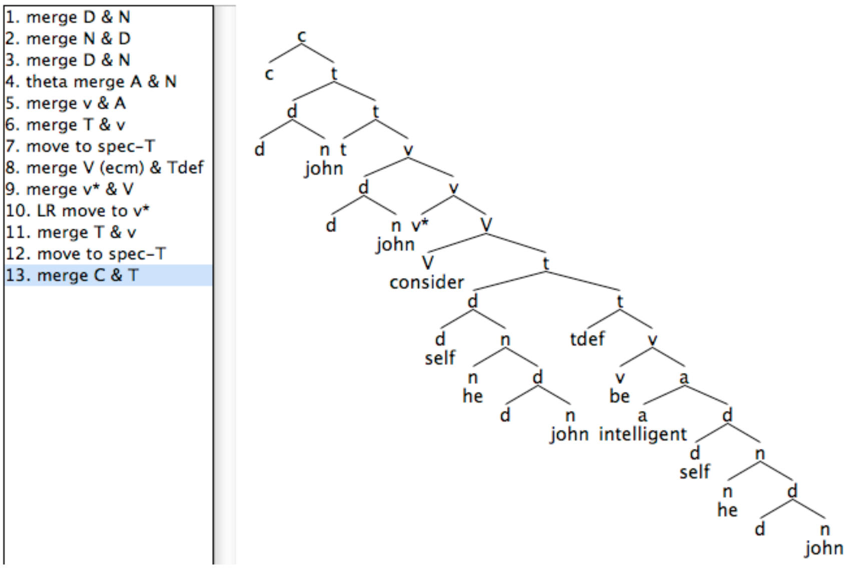


We also predict the possibility of an anaphor in subject position of an ECM clause, as in (38).

(38) $John_i$ considers $himself_i$ to be intelligent.

Unlike in the derivation of (36) above, 'John' is contained within a DP phase doubling constituent. As shown in (39), the DP doubling constituent raises to the embedded [Spec, TP]. When the matrix v* is merged, the DP phase will be closed-off. As a result, 'John' undergoes the LR process. Since v* selects for a subject and there is no other subject available for theta-merge, 'John' is reused immediately and remerged in subject theta-position.

(39)

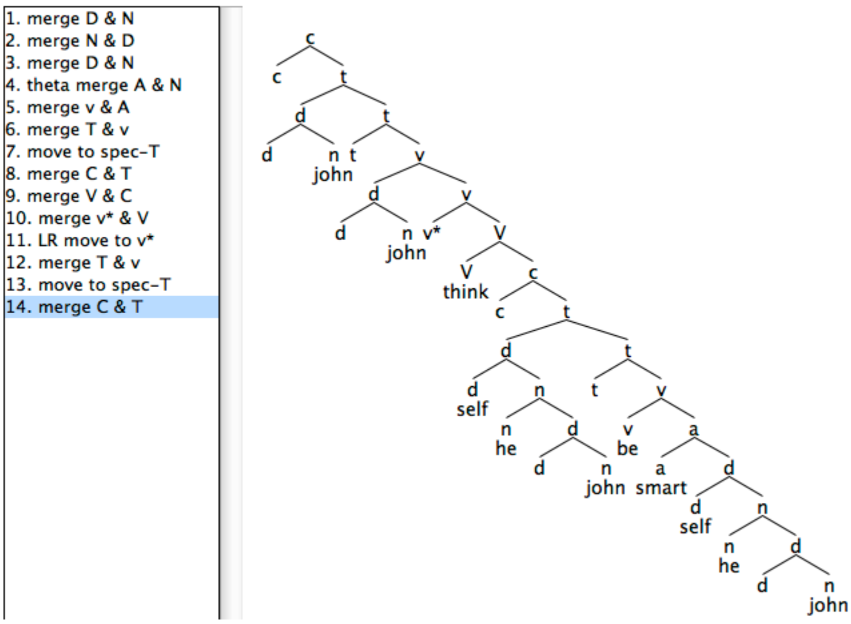


Next, consider the following example, in which 'heself' occurs in subject position of an embedded clause.

(40) *John$_i$ thinks heself$_i$ is smart.

As shown in (41), when the embedded C is merged, 'John' will undergo the LR process and be inserted into a buffer (since theta-merge is not possible), after which it can be merged in subject theta-position.

(41)



We attribute the ill-formedness of this example to an independently motivated ban on nominative reflexives (e.g., there is no 'heself' in English). Rizzi (1990) argues that the impossibility of a nominative anaphor results from an "anaphor agreement effect" - the impossibility of an anaphor to undergo agreement. A subject anaphor in a finite clause in English undergoes agreement with the local verb, thus resulting in ill-formedness. We leave investigation of this anaphor agreement effect for further work, but see Rizzi (1990) and Woolford (1999) for further discussion.[14]

We have applied our analysis to a variety of pronoun-antecedent facts. We have accounted for the same data that Kayne's analysis can account for (3-6). In addition, we have demonstrated how our analysis accounts for data that are problematic for Kayne's analysis (7-9).

5.3 Pronoun-antecedent relations in possessive DPs

We next turn to possessive DPs. As discussed in section 2, the possessive DPs in (10a) and (11a), repeated below, are a problem for Kayne's analysis. Our analysis, however, predicts these judgments.

(42) (a) John$_i$ likes his$_i$ dog.

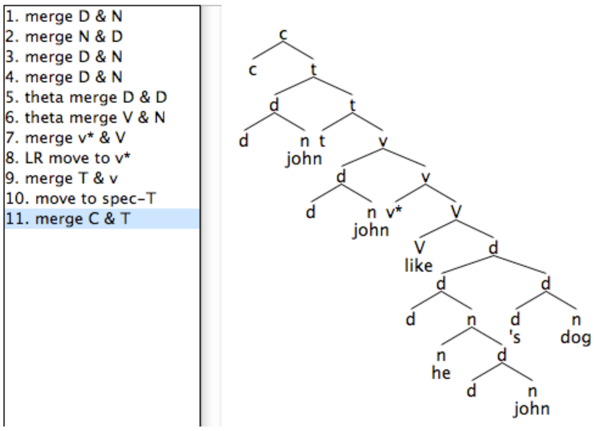
---

[14] Rizzi (1990) presents two possible explanations, which rely on the notion that agreement is a pronominal element, for the anaphor agreement effect. One proposal is that a referential autonomy hierarchy, of the form "R-expressions > pronouns > anaphors (Woolford 1999:278)" is at play. An anaphor in subject position of a finite clause is an argument and the agreement of the anaphor is a nonargument pronominal. The anaphor and its agreement form a chain. This agreement, being a pronoun, is higher on the referential autonomy hierarchy than the anaphor. The agreement-anaphor chain is ruled out by a ban against a "non argument in the chain which is higher in the referential autonomy hierarchy than the argument (Rizzi 1990:37)." Rizzi's other proposal is that when an anaphor undergoes agreement, the result is a single binding domain that contains an anaphor, subject to Condition A, and a coindexed pronominal (the agreement) which is subject to Condition B. Since both Condition A and Condition B cannot be met within the same binding domain, the result is ungrammatical. See Woolford (1999) for further discussion of Rizzi's proposals, as well as evidence that anaphors can undergo agreement, but only if this is a special type of anaphoric agreement. We leave for future work the issue of how exactly this anaphoric agreement effect can be incorporated into our doubling constituent analysis.

(b) *John$_i$ likes himself$_i$'s dog.

Example (42a) is interesting because the pronoun 'he', assuming that 'he' + ''s' = 'his', appears to be bound locally - a Condition B violation. Our analysis accounts for the well-formedness of this construction as resulting from the possessive DP being a phase. The derivation of (42a) is shown in (43).

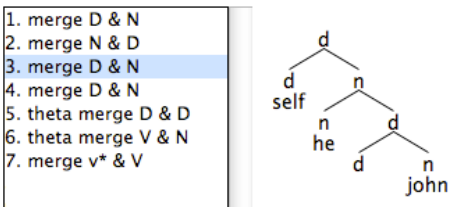
(43)



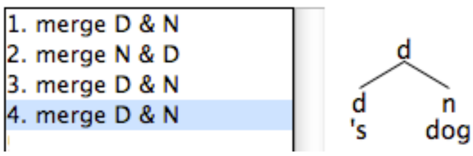
When v* is merged, the possessive DP phase will be closed off. Therefore, the r-expression 'John' undergoes LR and is remerged immediately with v* - a subject theta-role is assigned and the derivation converges.

The relevant parts of the derivation of the ill-formed (42b) are shown in (44a-d).

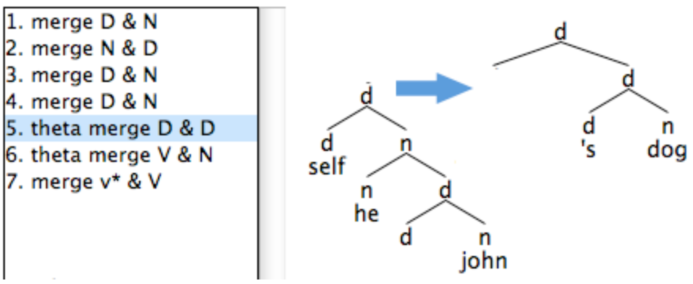
(44) (a)



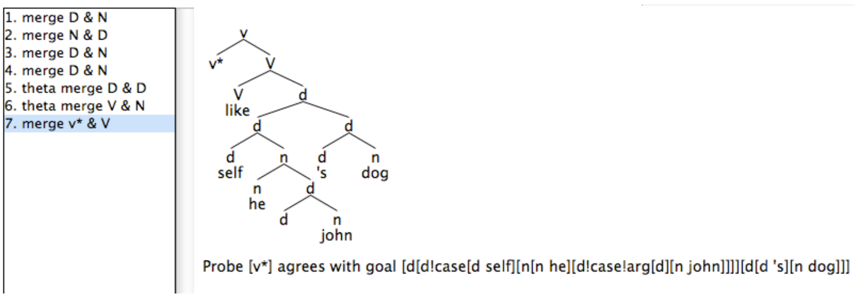
(b)



(c)



(d)



In (42b), we need to explain what prevents 'John' from undergoing internal merge to specifier of v*, the sentential subject position. In fact, we argue that 'John' is blocked from being moved at all from its originating position inside the doubling constituent. First, the anaphor doubling constituent (44a) and the possessive (44b), both phases, are constructed separately. Since both DPs are phases, it appears that it should be possible *prima facie* to raise 'John' to the specifier of possessive ''s' by applying LR. However, internal merge can only apply to already merged syntactic objects; LR itself being a special case of internal merge. If (44a) and (44b) are to be merged first to satisfy this constraint, as illustrated in (44c), the possibility of LR disappears altogether since the target position, namely the specifier of possessive ''s', will be filled by (44a). In other words, merge of the anaphor doubling constituent essentially blocks LR extraction of 'John' from within the doubling constituent. Continuing on, when v* is merged as shown in (44d), 'John' is unavailable for theta-merge to specifier of v* because, being beyond two phase boundaries down, it is inaccessible and LR cannot save it. Without an available subject for v*, the derivation can progress no further, and therefore crashes.[15]

5.4 Pronoun-antecedent relations in picture DPs

We next demonstrate how our model can be extended to account for certain picture DP constructions. An experimental study of native speaker judgments of picture DPs by Keller & Asudeh (2001) found that pronouns and anaphors are in complementary

[15] Note, in principle the derivation must crash even if a suitable subject is made available for direct merge from the numeration, as in **Mary likes [self he John]'s dog*. In this case, the doubling constituent merges with the possessive DP. After merge of v*, the subject 'Mary' is merged. The derivation crashes because the r-expression 'John' cannot receive a theta role in its (original) doubling constituent position.

distribution in some cases (45a-b) and (45c-d), but not in others (45e-f).[16,17] Examples (45e-f) are interesting in that subjects found both the pronoun and anaphor to be generally acceptable.[18]

(45) (a) ?*Hannah$_i$ found a picture of her$_i$.

(b) Hannah$_i$ found a picture of herself$_i$.

(c) ?*Hannah found Peter$_i$'s picture of him$_i$.

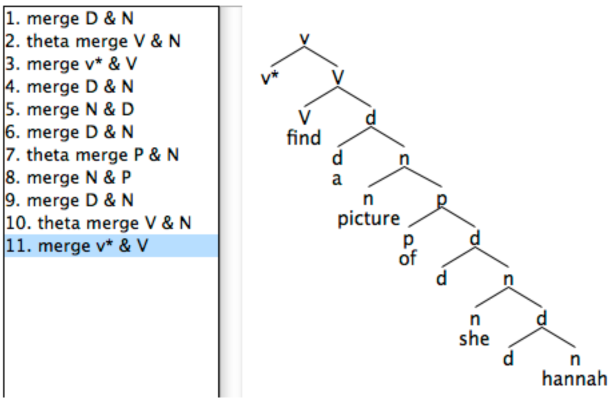
(d) Hannah found Peter$_i$'s picture of himself$_i$.

(e) Hannah$_i$ found Peter's picture of her$_i$.

(f) Hannah$_i$ found Peter's picture of herself$_i$.

The complementary distribution between (45a-b) depends on the availability of LR movement, as determined by whether or not the r-expression is base generated in a DP phase. In (45a), as shown in (46), when v* is merged, the r-expression 'Hannah' is not contained within a phase that is about to be closed-off. Therefore, it cannot undergo the LR process and no subject theta-role is assigned, causing the derivation to crash.

(46)



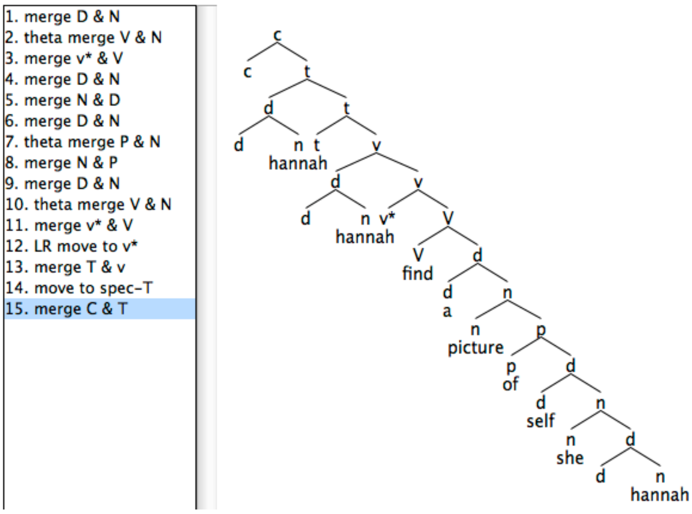
16 The judgments that we give reflect those presented by Keller & Asudeh (2001), as resulting from their study.

17 Phrases containing "picture", generally referred to as picture NPs or DPs, have been problematic for Binding Theory since its inception (e.g. see Chomsky 1981, 1986).

18 We note that (45f) does not seem to be completely acceptable to us - see the discussion of (53). Also, the traditional Binding Theory predicts that this example should be ill-formed (cf. Keller & Asudeh 2001).

In (45b), the r-expression originates within a DP phase. Thus, when v* is merged, since the DP doubling constituent will be closed off, 'Hannah' undergoes the LR process. In this case, it is remerged immediately in subject theta-position and the derivation converges, as shown in (47).

(47)



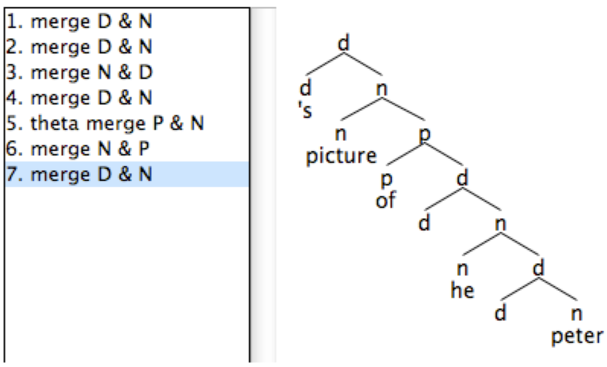
The complementary distribution between a pronoun and anaphor in possessive picture-DPs, as in (45c-d), repeated below, is also accounted for. Crucially, a possessive D ''s' is a phase head.

(48) (a) ?*Hannah found Peter$_i$'s picture of him$_i$.

(b) Hannah found Peter$_i$'s picture of himself$_i$.

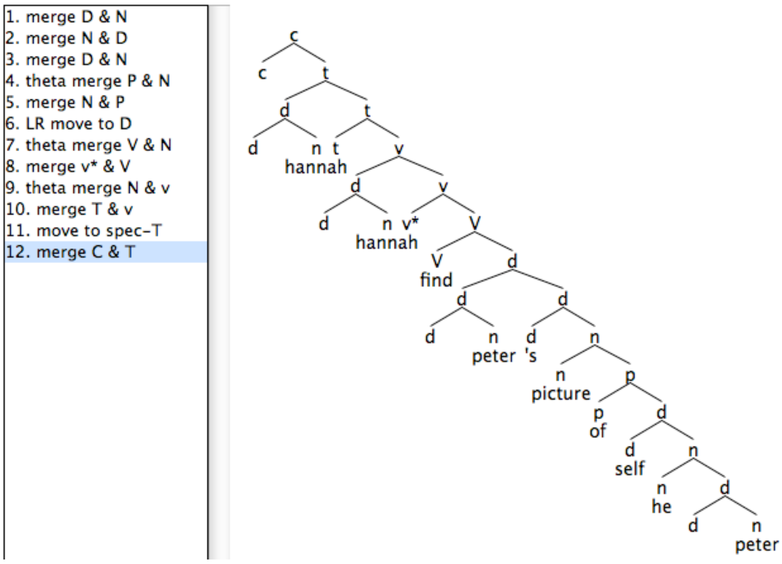
In the ill-formed (48a), when the possessive D ''s' is merged, the r-expression, not being contained within a lower phase, is unable to undergo the LR process and a possessive subject theta-role is not assigned, causing the derivation to crash.

(49)



In the well-formed (48b), 'Peter' is base generated within a DP phase doubling constituent. When the possessive D ''s' phase head is merged, 'Peter' undergoes LR and is remerged in subject theta-position of the possessive DP since ''s' selects for a subject. The derivation converges, as shown in (50).

(50)

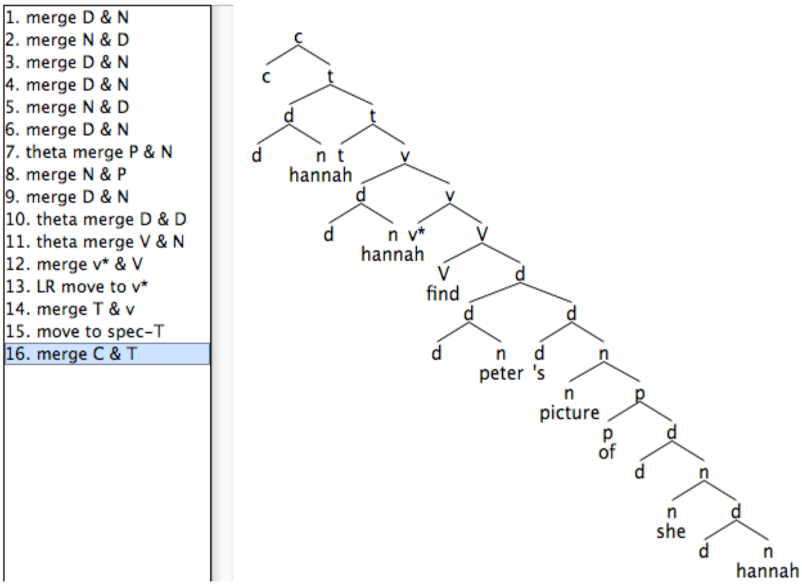


Example (45e), repeated below, is also accounted for.

(51) Hannah$_i$ found Peter's picture of her$_i$.

As shown in (52), the doubling constituent is not a DP phase. However, it is contained within a possessive DP phase. Thus, when the matrix v* is merged, 'Hannah' undergoes the LR process and is remerged in subject theta-position.

(52)

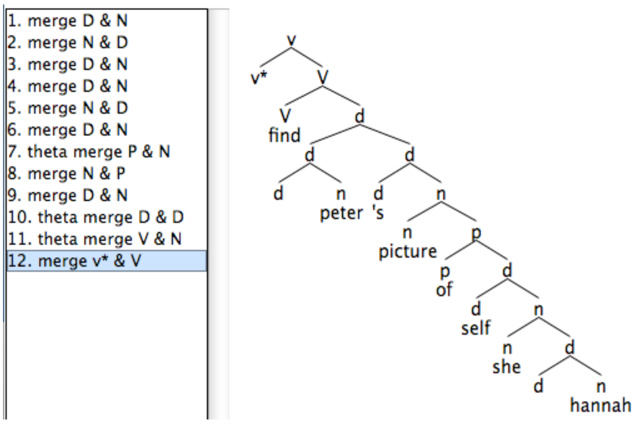


Lastly we turn to (45f), repeated below.

(53) Hannah$_i$ found Peter's picture of herself$_i$.

Our model produces the following derivation, which crashes.

(54)

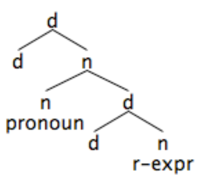


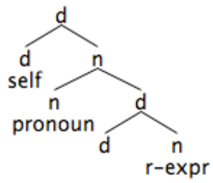
When the possessive D phase head ''s' is merged, the lower DP phase containing 'Hannah' will be closed off. 'Hannah', which contains unvalued features, is blocked from undergoing the LR process because of the possibility of external theta-merge of 'Peter' to subject position of the possessive DP. Therefore, when v* is merged, there is no subject available and the derivation crashes. This is the derivation at work for people who find this example ill-formed; our intuition is that this example is not perfect. However, Keller & Asudeh (2001) found that subjects considered this example to generally be well-formed. For those who find this well-formed, we suggest the following possibility; inherent case assignment may void a phase boundary in order to save a parse. In this construction, 'of' assigns inherent genitive case to its DP doubling constituent complement. This case assignment may, for some people, void the DP phase boundary when there is an LI within it that needs raising, resulting in the DP doubling constituent not functioning as a phase. When the matrix v* is merged, the lower possessive picture DP will be closed off, thus forcing LR to apply to 'Hannah'. Crucially, this phase boundary voiding only works to save a parse; otherwise, LR would not be triggered for (45b,d). This proposal, however, is speculative and requires further investigation.

6 The doubling constituent structure

We next return to our proposed structures for pronouns and antecedents, (13a-b), repeated below, which crucially rely on a difference in phasehood between a pronoun and antecedent doubling constituent and an anaphor and antecedent doubling constituent, the latter being a phase.

(55) (a) pronoun and r-expresssion (b) anaphor and r-expression





There are two basic properties of the structures that are crucial: (a) phrases with 'self' are phases, and (b) the pronoun selects for the r-expression and doesn't move. There may be further structural details that can be elaborated on in future work; e.g., the internal structure of the doubling constituent may be more elaborate than what we propose, but this is not crucial to our analysis. We have demonstrated, in the previous sections, that these structures, combined with the LR process, account for a variety of coreference data.

We next turn to a further piece of evidence that supports the proposal that 'self' is the head of a DP phase; example (56).

(56) John$_i$ is self$_i$-praising.

This example is accounted for if the r-expression and 'self' originate in the following structure, where D*P indicates a phase DP.

(57) [$_{D*P}$ self John]

In (56), when v* is merged, the D*P anaphor phase is closed-off (58a). LR applies (58b) and 'John' is remerged in subject position. We assume that 'self' affixes onto the verb via a process of morphological incorporation, the nature of which we leave aside.

(58) (a) [$_{v*P}$ v* praising [$_{D*P}$ self John]]

(b) LR: Remerge 'John' with v*

(c) [$_{v*P}$ John praising [$_{D*P}$ self ~~John~~]]

Thus, (56) is accounted for straightforwardly if 'self' is a phase head, even though it originates in a doubling constituent that lacks a pronoun (57). This example supports our proposal that an anaphor is base generated in a DP with the phase head 'self'.

In addition, Hicks (2009) independently proposes a structure for certain DPs in tough-movement constructions that is very similar to our proposed structure. Hicks proposes that in a tough-movement construction such as (59a), there is a DP with the structure in (59b). There is a null Operator N head with a DP complement 'everyone'. The null operator is licensed in its base position and the DP complement undergoes movement in order to become licensed - to obtain case.

(59) (a) Everyone is tough for us to please. (Hicks 2009:547)

(b) [$_{DP}$ D [$_{NP}$ [$_{N}$ Op] [$_{DP}$ everyone]]]

See Hicks (2009) for the details of this analysis. The key point that is important for us is that in this proposed structure, there is an N head with a DP complement and the DP complement undergoes movement; i.e., Hicks' proposed structure of a complex-DP in a tough-movement construction, if correct, provides independent evidence for our proposed doubling constituent structures of a pronominal-antecedent construction in which there is a pronominal N that has a DP r-expression complement.

7 Conclusion

Computationally speaking, a system that exhibits unbounded discrete infinity can still exhibit operational efficiency if its primitive operations have bounded or limited scope and there are no unnecessary choice or computational branching points in the system. The merge/agree system, as proposed in Chomsky (2001), exhibits both these properties. The probe-goal mechanism that underlies the agree relation is efficient in the (first) sense since goal search is phase bounded. Our introduction of Last Resort displacement directly to an edge theta position is similarly a local (and thus) efficient operation since it operates just within the same probe-goal search domain. Structure building in Chomsky's system is also efficient in the (latter) sense since selecting whether to externally merge or displace is fixed by the choice of LIs and disambiguating maxims such as Maximize Matching Effects or Merge Over Move. Our proposals continue to preserve this efficiency. Last Resort creates no new computational choice points since by definition it is available only when no other operations apply. Our novel buffer mechanism facilitates long distance displacement of antecedent r-expressions to a target theta position in a single paired operation without recourse to iterated movement. It also does not introduce any extra choice points into the system since the competing operation of external theta merge takes precedence over and blocks its application.

In conclusion, this analysis accounts for more data than Kayne (2002) with fewer stipulations. A wide variety of coreference facts result from base generation of a pronoun and antecedent within a DP, and the possibility of Last Resort movement (that can carry an r-expression into theta- position) as determined within the limits of Phase Theory.

References

Chomsky, Noam. 1981. *Lectures on Goverment and Binding*. Dordrecht: Foris.

——. 1986. *Knowledge of language: its nature, origin, and use*. New York: Praeger.

——. 1995. *The Minimalist Program*. Cambridge, MA: MIT Press.

——. 1999. *Derivation by phase*. MIT Working Papers in Linguistics: MIT Occasional Papers in Linguistics Number 18.

——. 2000. Minimalist inquiries: The framework. In Step by step, ed. by Roger Martin, David Michaels, & Juan Uriagereka, 89–155. Cambridge, MA: MIT Press.

——. 2001. Derivation by phase. In K*en Hale: A life in language*, Michael Kenstowicz (ed), 1–52. Cambridge, MA: MIT Press.

——. 2004. Beyond explanatory adequacy. In *Structures and beyond: The cartography of syntactic structures, volume 3*, Adriana Belletti (ed), 104–131. Oxford, UK: Oxford University Press.

——. 2005. Three factors in language design. *Linguistic Inquiry* 36: 1–22.

——. 2006. On phases. In *Foundational issues in linguistic theory; essays in honor of Jean-Roger Vergnaud*, Robert Freidin, Carlos Otero & Maria-Luisa Zubizaretta (eds), 133–166. Cambridge, MA: MIT Press.

Heinat, Frederik. 2003. Pronouns in a derivational syntax. In *The Department of English: Working papers in English linguistics, vol. III*, Henrik Gyllstad & Frederik Heinat (eds). Lund University, Centre for Languages and Literature.

Hicks, Glyn. 2009. Tough-constructions and their derivation. *Linguistic Inquiry* 40: 535–566.

Hiraiwa, Ken. 2005. *Dimensions of symmetry in syntax: Agreement and clausal architecture*. Massachusetts Institute of Technology dissertation.

Hornstein, Norbert. 1999. Movement and Control. *Linguistic Inquiry* 30: 69–96.

——. 2001. *Move! A minimalist theory of construal*. Malden, MA: Blackwell.

Kayne, Richard. 2002. Pronouns and their antecedents. In *Derivation and explanation in the Minimalist Program*, Samuel David Epstein & T. Daniel Seely (eds), 133–166. Malden, MA: Blackwell Publishing.

Keller, Frank & Asudeh, Ash. 2001. Constraints on linguistic coreference: Structural vs. pragmatic factors. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*, Johanna D. Moore & Keith Stenning (eds), 483–488. Mahawah, NJ: Lawrence Erlbaum.

King, Jonathan W., & Kutas, Marta. 1995. Who did what and when? Using word- and clause-level ERPs to monitor working memory usage in reading. *Journal of Cognitive Neuroscience* 7: 376–395.

O'Neil, John. 1995. Out of control. In *Proceedings of NELS 25*, Jill Beckman (ed), 361–371. Amherst, MA: GLSA.

——. 1997. *Means of control: Deriving the properties of PRO in the Minimalist Program*. Harvard University dissertation.

Reuland, Eric & Everaert, Martin. 2001. Deconstructing Binding. In *The handbook of contemporary syntactic theory*, Mark Baltin & Chris Collins (eds), 634–669. Malden, MA: Blackwell Publishing.

Rizzi, Luigi. 1990. On the anaphor-agreement effect. *Rivista di Linguistica* 2: 27–42.

Stabler, Edward P. 1997. Derivational minimalism. In *Logical aspects of computational linguistics*, C. Retoré (ed), 68–95. New York: Springer.

Svenonious, Peter. 2004. On the edge. In *Peripheries: Syntactic edges and their effects*, David Adger, Cécile de Cat & George Tsoulas (eds), 259–287. Dordrecht: Kluwer.

Woolford, Ellen. 1999. More on the anaphor agreement effect. *Linguistic Inquiry* 30: 257–287.

Zwart, Jan-Wouter. 2002. Issues related to a derivational theory of binding. In *Derivation and explanation in the Minimalist Program*, Samuel David Epstein & T. Daniel Seely (eds), 269–304. Malden, MA: Blackwell Publishing.