STiL – Studies in Linguistics
CISCL Working Papers
Vol. 2, 2008

Phases, Strong Islands, and Computational Nesting*

VALENTINA BIANCHI,
University of Siena
bianchi10@unisi.it

Cristiano Chesi
University of Siena
chesi@media.unisi.it

This paper is an attempt to recast the connectedness condition (Kayne 1983) in derivational terms: we will show that a Top-Down derivation is adequate to describe strong island conditions (as in Huang's original proposal), without loosing the ability to discriminate among distinct phenomena (preverbal subject islandhood, complex-NP constraint, special properties of the first argument in double object constructions, intermediate status w.r.t. extraction/ gapping, for right adjuncts), and predicting, moreover, the grammatical distribution of parasitic/licensed gaps in the derived structure.

1. Left branch islands and the connectedness effect

In this paper we reconsider the connectedness effect discussed by Kayne (1983), and illustrated in the examples (1)-(3). Kayne observed that in VO languages, left branch constituents are strong islands for extraction;[1] however, an illegitimate gap inside a left branch island can be rescued by another gap embedded in a lower right branch constituent. The examples in (1)-(2) illustrate preverbal subject islands, and (3) a small clause subject island: while in the (a) examples, extraction from the left-branch subject is impossible, in the (b) examples, the illegitimate gap is followed by a legitimate gap on a right branch and this creates a grammatical configuration.

* This paper is a expanded and revised version of the talk presented at the 28th Glow Colloquium (Geneva, March-April 2005). We wish to thank Adriana Belletti, Cedric Boekcx, Alec Marantz, Andrew Nevins, Luigi Rizzi, Ur Shlonsky and Michal Starke for discussion of the material presented in this paper, as well as the audiences at the 29th Penn Linguistic Colloquium (February 2005), the Harvard Grammatical Locality reading group and at the 28th Glow Colloquium (Geneva, March-April 2005).

[1] Throughout the paper, by "strong islands" we mean *nonselective* islands, which do not give rise to argument/adjunct asymmetries in extraction, as opposed to weak (Relativized Minimality) islands, which selectively affect the extraction of certain constituents: see Rizzi (1990, 1994, 2002). See Starke (2001) for an interesting attempt at unification of the islands phenomena which however does not account for connectedness effects.

(1) a. *[Which famous playwright]$_i$ did [close friends of e_i] become famous?
b. $^?$[Which famous playwright]$_i$ did [close friends of e_i] admire e_i ?

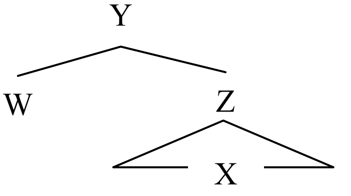
(2) a. *Who$_i$ did [my talking to e_i] bother Hilary?
b. $^{\surd}$Who$_i$ did [my talking to e_i] bother e_i?

(3) a. *Who$_i$ did you consider [friends of e_i] angry at Sandy?
b. $^{\surd}$Who$_i$ did you consider [friends of e_i] angry at e_i ?

Kayne (1983) proposed an essentially representational constraint to account for these data, the Connectedness Condition. The central notion is that of a g-projection, which is defined in (4)-(5). In a VO language like English, every right branch is in a canonical government configuration, by definition (4); the recursive definition in (5) ensures that all the maximal projections dominating a structural governor X and lying on a right branch are g-projections of X.

(4) W and Z (Z a maximal projection, and W and Z immediately dominated by some Y) are in a canonical government configuration iff
a. V governs NP to its right in the grammar of the language and W precedes Z
b. V governs NP to its left in the grammar of the language and Z precedes W

(5) Y is a g-projection of X iff
i. Y is an (X') projection of X or of a g-projection of X, or
ii. X is a structural governor and Y immediately dominates W and Z, where Z is a maximal projection of a g-projection of X, and W and Z are in a canonical government configuration



Thus, the g-projections of X can extend upward as long as any dominating maximal projection is on a right branch.

The Connectedness Condition (henceforth CC) requires that the set of the g-projections of (the governor(s) of) the empty category(ies) bound by a given binder and the binder itself form a connected subtree:

(6) The g-projection set G_β of a category β is defined as follows (γ governs β):
a. $\forall\pi$, π = a g-projection of $\gamma \rightarrow \pi \in G_\beta$
b. $\beta \in G_\beta$ and
b'. δ dominates β and δ does not dominate $\gamma \rightarrow \delta \in G_\beta$[2]

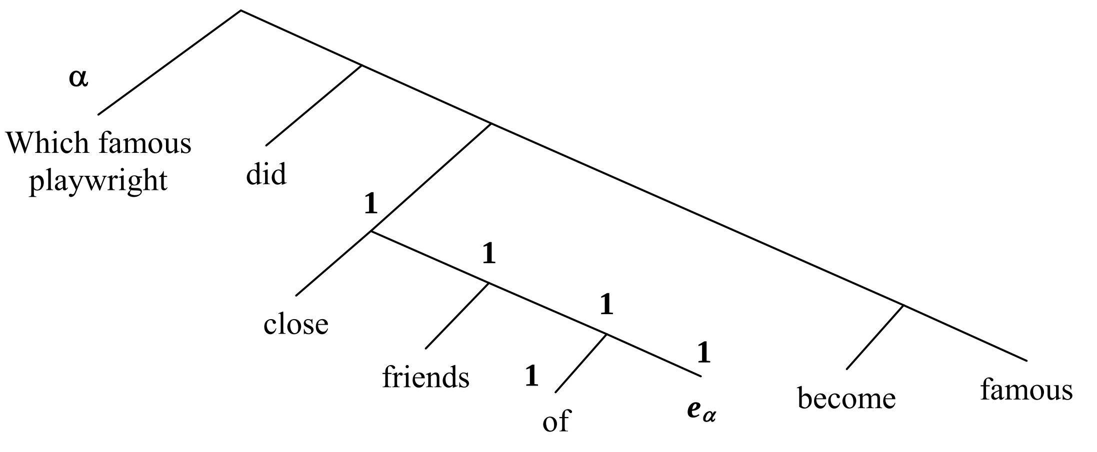
(7) Connectedness Condition[3]
Let $\beta_1 \ldots \beta_j, \beta_{j+1} \ldots \beta_n$ be a maximal set of empty categories in a tree T such that $\exists\alpha\forall j$, β_j is locally bound by α. Then $\{\alpha\} \cup (\bigcup_{1=j=n} G_{\beta j})$ must constitute a subtree of T.

In case of a single gap, the CC requires that all the maximal projections in the path between the gap and its binder be on a right branch. Consider for instance the

[2] Clause b' takes care of government across a clausal boundary δ.
[3] For our present purposes, we ignore the modification of the CC proposed by Kayne to account also for multiple wh structures and for negative elements licensed by clausal negation in Italian.

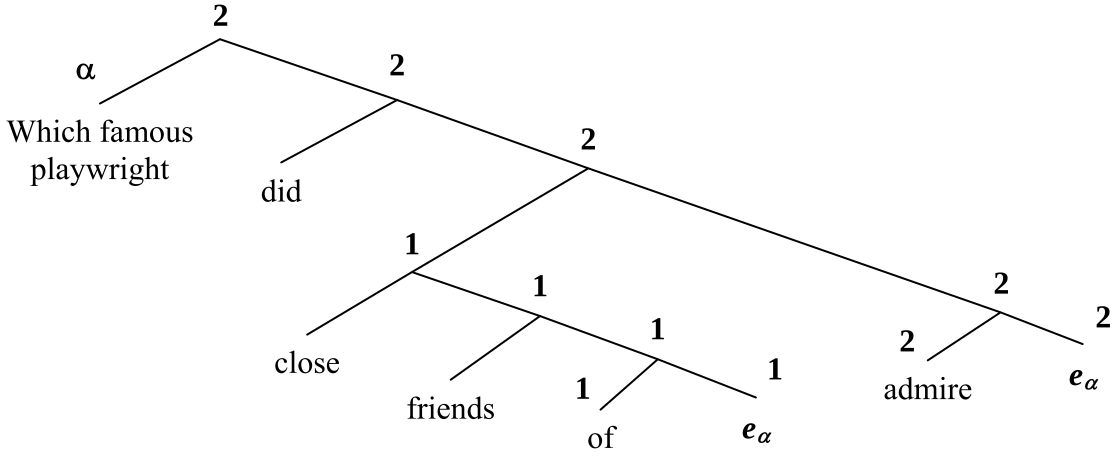
ungrammatical example in (1a): as the tree graph below makes clear, the g-projections of the gap stop at the level of the preverbal subject, which is a left branch and hence not in a canonical government configuration. Therefore, the g-projections cannot extend upward to reach the binder, and the CC is violated:[4]

(1.a) *



The rescuing effect in (1b) is due to the fact that the g-projections of the lower gap in the object position extend upward and connect to the g-projections of the illegitimate gap embedded in the subject, as shown in the tree below. As a result, the two g-projections sets form a connected subtree including the binder, and the CC is satified:

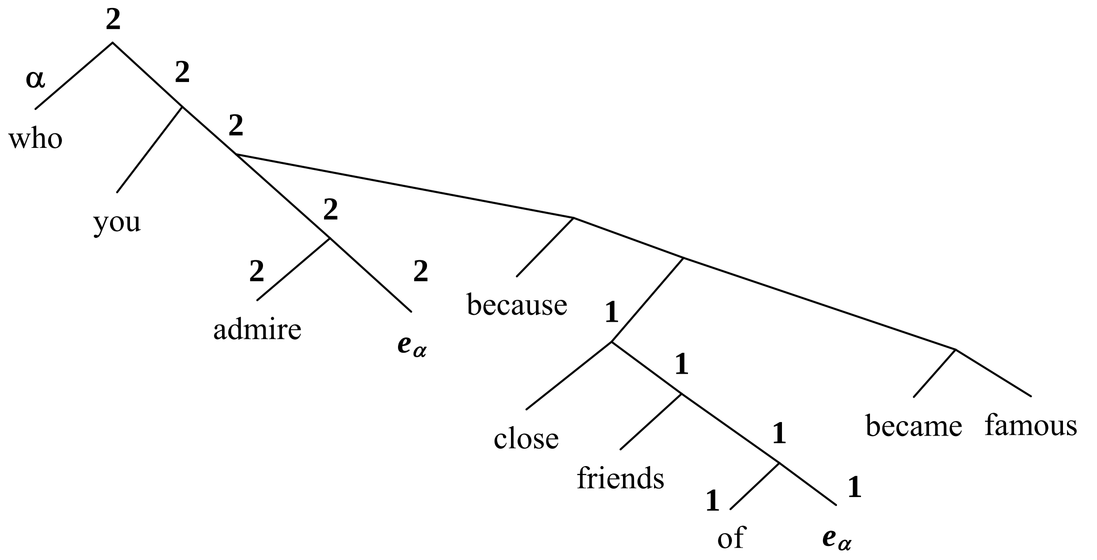
(1.b)



On the contrary, no rescuing effect arises if the legitimate gap is too high in the tree for its g-projection set to connect to that of the illegitimate gap, as in the following example:

(8) * a person who you admire e because [close friends of *e*] became famous

[4] As in Kayne (1983), the numerical indices are introduced for expository purposes to mark the g-projection paths of the empty categories, and have no theoretical significance.



The CC differs in various respects from other approaches to parasitic gaps in the GB framework. Firstly, even though there is a clear sense in which in the (b) examples of paradigms (1)-(2) the gap inside the left branch is "illegitimate" or parasitic, and the other one is legitimate, there is no other assumed difference between them, either with respect to the nature of the empty category or of its relation to the binder.

But the status of the parasitic gap and of its relation to the binder is actually debated, as can be seen in the collection of papers edited by Culicover & Postal (2001). Cinque (1990) and Postal (1994) have pointed out various types of evidence which suggest that the parasitic gap is a null resumptive pronoun rather than an ordinary extraction gap. The evidence comes from the lack of reconstruction effects in the parasitic gap position, the impossibility for parasitic gaps to occur in Postal's antipronominal contexts, and the restriction of parasitic gaps to the NP category. However, all these types of evidence have been called into question by other authors (cf. e.g. Hukari et al. 2001, Levine & Sag 2003); it seems fair to say that the issue is still open.

Secondly, note that the CC is designed to capture left branch islands only. Other strong island types, like e.g. right-hand adjuncts and relative clauses, are not subsumed under this condition. It is empirically debated whether strong islands are a uniform class falling under a single principle (as proposed for instance in Cinque 1990). As we will discuss in §4, even Longobardi's (1985) extension of the CC cannot subsume all the island effects that are usually classified as strong (unselective) islands. Despite these problems, we believe that the CC incorporates an important insight, which we will formulate as follows:

(9) Generalization on legitimate recursion and gap licensing
Legitimate gaps lie on the main recursive branch of the tree, whereas illegitimate gaps lie on “secondary” branches, which do not allow for unlimited recursion (in that such a secondary branch cannot be the lowest one in a tree).

It is this insight that we will try to capture in our approach, though in an essentially derivational perspective.

As a first pass, we will propose a derivational hypothesis that has the same empirical scope as Kayne's original CC, and only accounts for left branch islands (§3). In §4 we will come back to the problem of right-hand adjuncts.

As to the question of the (a)symmetry between legitimate and parasitic gaps, we will remain neutral. For the sake of simplicity, we will assimilate the parasitic gap-antecedent dependency to a standard antecedent-gap dependency, and treat both in terms of copy-remerging. However, we believe that the constraints on the structure of the computation that we are going to highlight are also consistent with an analysis in terms of a null resumptive pronoun.

Our proposal will be implemented in the computational model of a top-to-bottom oriented Minimalist Grammar proposed in Chesi (2004, 2007). Although limitations of space prevent us from fully justifying the proposed model, we will now give a brief sketch, which will constitute the background of our proposal.

2. The computational model

2.1. The general architecture

Chesi (2004, 2007) proposes a formalization of a minimalist grammar (adapting the formalism discussed in Stabler 1997) with two main components:

a. a lexicon consisting of feature structures composed of semantic, syntactic and phonetic features[5];
b. three structure building operations (*Merge*, *Move* and *Phase Projection*).

Chesi argues that for reasons of computational efficiency and cognitive plausibility[6], the grammar should have the property of *flexibility*: namely, it should be directly usable both in a parsing and in a generation context. The flexibility requirement leads Chesi to abandon the bottom-to-top orientation of the standard minimalist derivation, and to assume instead a top-to-bottom orientation (as in Phillips 1996).

Assume a Structural Description (SD) to be definable simply in terms of immediate relations (immediate dominance[7] and immediate precedence); assume, moreover, that any item is licensed within a SD (leading then to grammaticality) if and only if it is *selected*[8] or it is a possible *functional specification* of a lexical head. Accordingly, a lexical head is specified for two types of features: the SELECT features specify its argumental valency, and license the head's arguments (they correspond to the standard theta-grid or argument structure); the LICENSOR features instead specify the possible functional specifications that can be associated with the head: these correspond to the standard functional heads (FPs) in the lexical head's extended projection (Grimshaw 1991). Importantly, the LICENSOR features associated to a given lexical head are limited in number and are hierachically ordered, much as in the cartographic approach

[5] Nothing in this paper hinges on the feature theory we will use. For sake of simplicity, let us assume a simple (non recursive) privative system (Adger 2003, 2007). In the original version (Chesi 2004) complete attribute-value matrices where allowed, as in most unification grammars (e.g. HPSG, Pollard and Sag 1994).

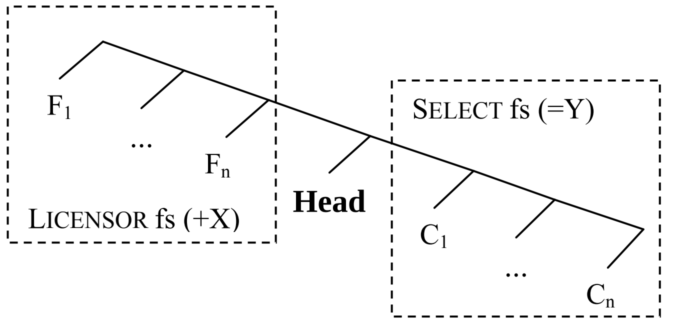
[6] It is hardly plausible that we would speak a language using a particular grammatical competence and that we would produce the very same language using a different knowledge.

[7] The statement "A *immediately dominates* B" would correspond to the result of a merge operation where A projects over B: $[_A$ A B].

[8] Here *selection* means both *C(ategorial)-selection* and *S(emantic)-selection* (Pesetsky 1982).

proposed by Cinque (1999), Rizzi (1997, 2004). The general schema is then the following:

(10)



Chesi (2004) then defines a general top-to-bottom algorithm which can be exploited both in generation and parsing; specifically, in generation, the algorithm converts a set of immediate dominance relations among semantic/formal feature structures into a set of immediate precedence relations among lexicalized phonological feature structures**;** vice versa, in parsing, it converts a set of immediate precedence relations among phonological structures into a set of immediate dominance relations among lexicalized semantic/formal structures).

From this perspective the structure building operations can be redefined as follows[9]:

(11) **Merge** is a binary function (sensitive to temporal order) which takes two feature structures and unifies them (in the sense of unification grammars, Shieber 1983)

(12) **Phase Projection** is the minimal set of dominance relations introduced in the SD based on the expectations triggered by the SELECT features of the currently processed lexical head.

(13) **Move** is a top-down oriented function which stores an unselected element in a memory buffer[10] and re-merges it at the point of the computation where the element is selected by a lexical head.

An *unselected* element is any element that is processed before the lexical head is found, and hence temporally and linearly precedes the head itself, according to the following Linearization Principle (inspired by Kayne's (1994) LCA):

(14) **Linearization Principle**
a. <A, B> if A (is a lexical head and) selects B as an argument
b. <B, A> is B is a functional specification of A.

Though limitations of space prevent us from fully describing the proposed model, we will illustrate how the structure building operations work in a simple example, where all the basic ingredients are involved:

---

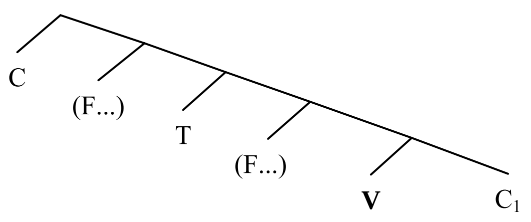
[9] See Chesi (2004), Ch. 3.3.2 for an explicit and thorough formalization.

[10] Limitations of space do not allow us to fully characterize the memory buffer (the reader is referred to Chesi 2004). Let us simply emphasize two points. First, the memory buffer must be *multidimensional*, i.e. different kinds of elements are stored in separate lists; this will account for the selectivity of intervention (Relativized Minimality) effects, cf. Rizzi (1997, 2001). Second, the minimality effect itself can be captured by assuming a Last In First Out memory, so that at a given point of the computation only the last element that was inserted in the buffer can be retrieved, and the previously inserted ones cannot.

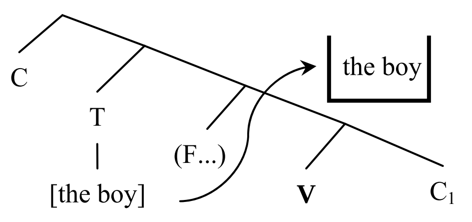
(15) The boy kissed the girl.

i. As the initial step, the system projects a top-down expectation of a verbal phase (i.e. a CP),[11] whose lexical head will have to be a verb.
ii. The constituent [*the boy*] is processed[12] and, being compatible with the functional Tense-related specification[13], it is inserted at the corresponding functional level. Since the element is not selected in this position, it is also stored in the memory buffer.

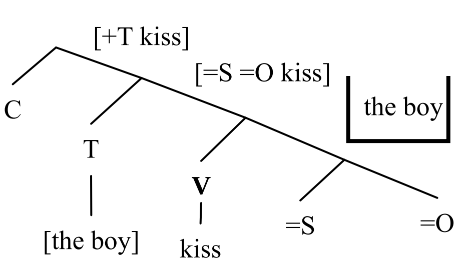
(16) i.



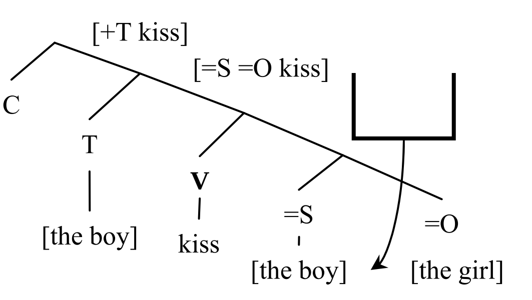
ii.



(16) iii.



iv - v.



iii. The lexical item *kissed* (analysed as *kiss* +T) is processed; this introduces in the derivation the verb's SELECT features, here abbreviated as =S (external argument) and =O (internal argument) which are projected, according to Phase Projection, starting from the most external one.[14]
iv. The constituent [*the boy*] previously stored in the memory buffer is re-merged as a sister to the verb to satisfy the verb's =S feature.
v. As a final step, the computation proceeds by processing the direct object.

We return immediately to the special status of the lowest selected complement, which follows from a novel definition of phase.

2.2. Phases

Chesi (2004) argues that in order to gain computational tractability, the derivation must be broken up into *phases*, i.e. subparts of the computational process with a fixed upper bound in complexity. The phase can be roughly defined as follows:

---

[11] This root application of Phase Projection is obviously not triggered by any SELECT feature.
[12] This actually constitutes a separate and "nested" computational phase, as will become clear in §2.2.
[13] Let us avoid complications with respect to the exact position of the subject, which is irrelevant for the present discussion.
[14] This is because we want to preserve scope relations and, as Phillip's (1996) Merge Right, from a derivational perspective, we expect these intermediate constituents to be built in the following order: [$_V$ S V] → [$_V$ S [$_V$ V O]]

(17) **Phase**

A phase is the minimal part of a top-to bottom computational process in which all the functional and selectional specifications associated to a given lexical head are satisfied.

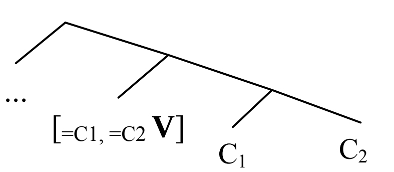
Intuitively, each phase corresponds to the computation of a "minimal chunk" of syntactic structure like (10) above. Importantly, each phase will have a fixed upper bound in depth, determined by a limited number of possible functional specifications (Cinque 1999, 2002) and of selected arguments (Pesetsky 1982). Note however that, contrary to the standard bottom-to-top derivation, here a phase does not correspond to a complete subtree. In fact, when Phase Projection is triggered by the last SELECT feature of the lexical head, the current phase gets closed, and the computation of the complement constitutes the next phase. Thus, a phase corresponds to a subtree whose lowest selected element is not yet expanded.[15] For the sake of simplicity, we assume here that only V and N can head a phase, and accordingly, phases correspond to the computation of a CP or DP chunk.[16]

Crucial to our argument is the distinction between *sequential* and *nested* phases.[17] As we have just said, when a phase reaches the lowest position selected by the lexical head, it is closed off: the expansion of the complement constitutes the next, *sequential* phase. A sequential phase thus follows the phase of the selecting head, and is separated from it.

On the other hand, any DP or CP within a phase P_n that does not occur in the lowest position selected by the lexical head of P_n constitutes a *nested* phase, which must be processed while P_n is still incomplete. Hence, all unselected DPs or CPs preceding the lexical head of P_n are necessarily nested phases: a preverbal subject, a fronted wh- or topical phrase can only be a nested phase (and additionally, when its computation is completed it is stored in the memory buffer of P_n). In (15), for instance, the subject DP [*the boy*] constitutes a nested phase within the matrix CP phase.

Consider now a situation where a lexical head selects two complements. Since both are selected, in principle it is possible to apply Phase Projection for both:

(18)



One possibility is to allow for both C_1 and C_2 to be computed as sequential phases (with C_2 sequential to C_1, which is in turn sequential to the selecting V's phase).

Alternatively, we can make the more restrictive assumption that only C_2 can be sequential to the V's phase, and C_1 constitutes instead a nested phase. These assumptions will have different consequences for the islandhood of double complement structures (see below the discussion around (22)-(24)).

[15] Alternatively, the last SELECT feature can be satisfied by discharging an already processed constituent stored in the memory buffer by a previous application of Move.

[16] From our perspective, vP is not a separate phase from CP.

[17] The distinction between sequential and nested phases is independently justified by their different effects on the computational complexity function (see Chesi 2004 for thorough discussion).

If phases are minimal chunks of the syntactic computation, it is reasonable to assume that each phase has its own local memory buffer for Move.[18] However, since long-distance movement can cross phase boundaries, it is necessary to devise a way to transmit the content of a phase's memory buffer to that of another phase. For this purpose we adopt the following Success Condition:

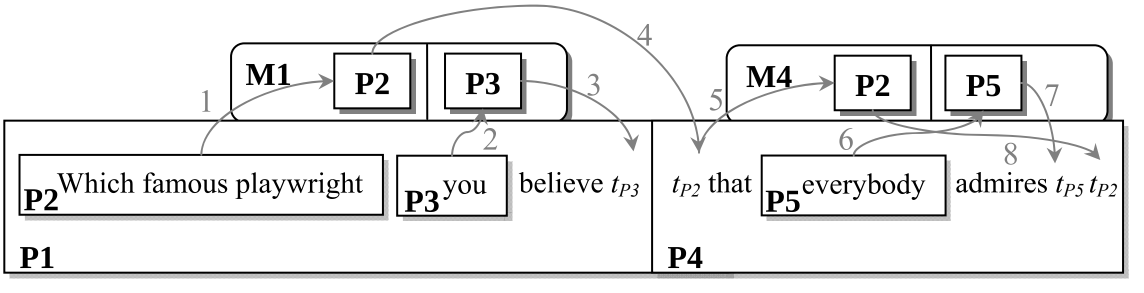
(19) **Success Condition**
At the end of each phase the local buffer is empty, or else its content is inherited by the memory buffer of the next sequential phase (if any).

Crucially, this condition only allows for communication between the memory buffer of two adjacent sequential phases. (Obviously, at the end of the last phase of all the local buffer will have to be empty). This accounts for the transparency of the lowest recursive branch of the tree.

To see this, consider for instance a computation for (20), as schematically represented in (21) (where the boxes identify phase boundaries):[19]

(20) Which famous playwright do you believe that everybody admires?
$[_{CP}$ $[_{DP}$Which famous playwright$]_i$ do you believe
$[_{CP}$ *whP_i* [that everybody admires *whP_i*]?

(21)



The algorithm initializes a CP phase 1 (P1). Then it computes the wh-phrase, which constitutes a separate nominal phase 2 (P2). Since the wh-phrase is not selected, it is stored in the local memory buffer (M1) of P1 by Move (step 1). Then, the computation of P1 proceeds, down to the complement position of the matrix verb *believe* (we disregard the computation, storage and retrieval of the subject phase P3: step 2, 3). At this point P1 is closed and the wh-phrase (P2) in its memory buffer is discharged into the complement CP phase 4 (P4), since the latter is sequential and selected. We propose that this takes place by re-merging the content of the memory buffer of P1 in the left periphery of the complement CP, P4 (step 4); since this position is unselected, the wh-phrase is re-stored in the local memory buffer of P4 (step 5). As a result, the "inheritance" mechanism leaves an intermediate copy/trace in the edge of the complement CP phase.[20] The computation proceeds down to the object position of the

[18] This is our way to reconstruct Chomsky's "Phase Impenetrabilty Condition" for movement.
[19] The graphic representation (21) and the following ones are not very perspicuous, but it is the best possible representation on a bi-dimensional sheet that we could figure out. Intuitively, the left-to-right orientation of the written line corresponds to the progress of the computation in time; the boxes represent phase boundaries; the arrows represent steps involving storage into / retrieval from a memory buffer; the "sectors" in the memory buffer represent the different cells in which different types of moved elements are stored (cf. note 8).
[20] Although this assumption is not strictly necessary for the algorithm to work, it seems fairly natural and it allows us to capture various successive cyclicity effects, like e.g. Irish complementizer

verb *admires*, where the wh-phrase P2 is discharged from the local memory buffer of P4 and re-merged (step 8): the Success Condition is thus satisfied at the end of the computation.

Going back to the problem of double complements as in (18), if both are computed as sequential phases we expect both to be transparent for successive cyclic “extraction” or – to state the same thing from our current perspective – to be able to inherit the content of the memory buffer of the matrix V's phase. In fact, in a system like Chomsky (1986) or Cinque (1990) both complements are expected to be transparent, since both are selected (theta-marked) by the V head. However, the empirical evidence is not that simple. Kuno (1973, 380 ff.) pointed out long ago that for certain speakers, the first complement in a double complement structure resists subextraction:

(22) a. John gave a picture of Mary a finishing touch.
b. ?? Who did John give [a picture of *t*] a finishing touch?

(23) a. John handed a picture of Mary to Bill.
b. ?? Who did John hand [a picture of *t*] to Bill?

(24) a. John gave moving to Florida serious consideration.
b. ?? Where did John give [moving to *t*] serious consideration? [21,22]

These data led Kuno to propose the "clause nonfinal incomplete constituent constraint" (cf. Pollard & Sag 1994:190), also subsuming subject islands. According to Kuno, this constraint only holds for some speakers. On the other hand, judging from the literature, subject islands of the type exemplified in (1)-(3) seem to be much more robust than the island effects in (22)-(24): we suspect that the two phenomena should not be collapsed under one and the same constraint. From our perspective, preverbal subjects can never be sequential phases (see §3 below for more discussion); however, the first complement in a double complement structure might be sequential or nor, depending on the choice we make for the configuration (18). It is at least conceivable that certain speakers might be more restrictive, computing only the last complement as a sequential phase, while others might be less restrictive and allow for both complements to be computed as sequential phases. In our own native language, we do perceive a contrast in (25), where the first complement is a finite clause undergoing extraction in the (b) example:

(25) a. (?) Ho annunciato [che licenzierò Maria] [a tutti i miei colleghi].
(I) have announced that (I) will-fire Mary to all my colleagues
b. ?* Chi hai annunciato [che licenzierai t] [a tutti i tuoi colleghi]?
Whom have (you) announced that (you) will-fire to all your colleagues

The data are not clear enough yet to draw a firm conclusion: we leave the issue at that. In the following discussion we will not consider double complement structure. To

alternations (McCloskey 1990) or French stylistic inversion (Kayne & Pollock 1978). We thank Luigi Rizzi for discussion of this point.

[21] According to Pollard & Sag (1994:182ff.), this configuration too gives rise to connectedness effects:
(i) Which of our relatives should we send [snapshots of *e*] [to *e*]?

[22] According to Kayne (1983), ECM subjects constitute left branch islands:
(i) * [Which book]$_i$ do you believe [the first chapter of e_i] to be full of lies?
This is exactly what the CC (and our reconstruction of it) would predict on the basis of the classical GB analysis of ECM complements. However, three informants whom we consulted found this sentence only mildly deviant. We leave the problem open for future research.

summarize, the following points of Chesi's (2004) model will be crucial for the development of our analysis:

a. Every computation is a top-down process divided into phases of fixed maximal size.
b. A phase gets closed when the lowest selected position of its head is processed; the lowest selected complement constitutes the next sequential phase.
c. All unselected constituents are instead nested phases: they are processed while the superordinate phase has not been closed yet.
d. The Move operation stores an unselected element found before (i.e. to the left of) the head in the local memory buffer of the current phase, and discharges it in a selected position if possible; if not, when the phase is closed the content of the memory buffer is inherited by the next sequential phase. The memory buffer of the last phase must be empty at the end of the computation.

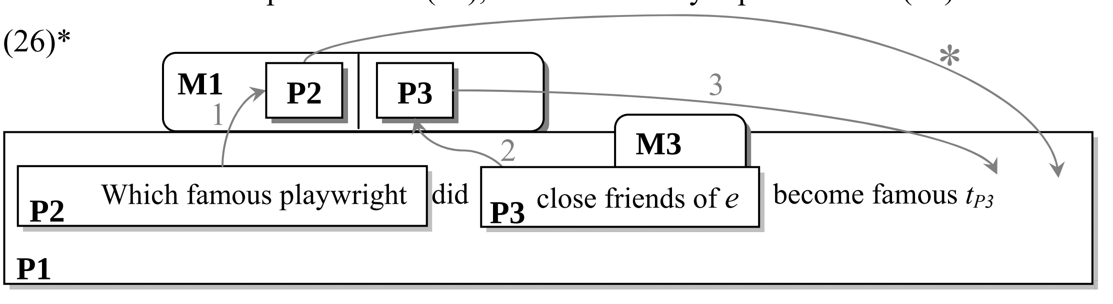
3. Left-branch islands are computationally nested phases

With this background, we can now go back to our initial problem, namely, left branch islands and the connectedness effect. The contrast in (1) is repeated here for convenience:

(1) a. *[Which famous playwright]$_i$ did [close friends of e_i] become famous?
b. [?][Which famous playwright]$_i$ did [close friends of e_i] admire e_i ?

Consider now a computation for (1b), as schematically represented in (26):

(26)*



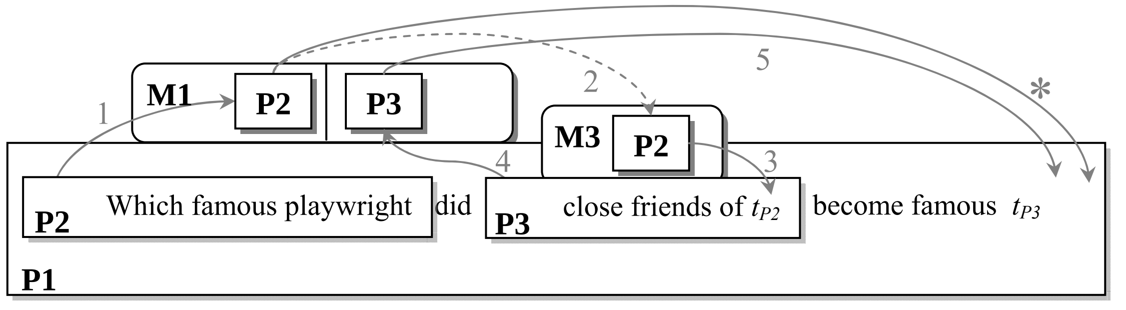
Once again, the algorithm initializes a CP phase 1; then it computes the wh-phrase in a separate nominal phase 2, and stores it in the local memory buffer of phase 1 (M1, step 1). The computation of phase 1 proceeds, inserting *did* in C. As a next step, a distinct nominal phase 3 for the subject DP must be opened, while the clausal phase 1 is still incomplete. The DP phase 3 is thus a nested phase, and its local memory buffer (M3) does not contain the wh-phrase which was stored in the memory buffer of phase 1 (M1): hence, the wh-phrase cannot be discharged in the selected gap position within the subject DP. The wh-phrase also remains undischarged at the end of the computation of phase 1, violating the Success Condition (19). This accounts for the strong island effect.

Suppose now that we optionally allow the memory buffer of the nested subject DP phase 3 to "copy" the buffer of the immediately superordinate phase 1, which contains the wh-phrase (this "parasitic copying" is represented by a dotted line in (27), step 2).[23] Then, the wh-phrase can be discharged in the gap position within the DP phase 3

[23] Optional copying would actually introduce non-determinism in the computation. In order to avoid this, we could assume either the possibility of using (a.) backtracking or (b.) parasitic usage (without discharging its content) of the first active memory buffer of a super-ordinate phase: when the computation of phase 3 reaches the position selected by the noun *friends*, since there is no more lexical

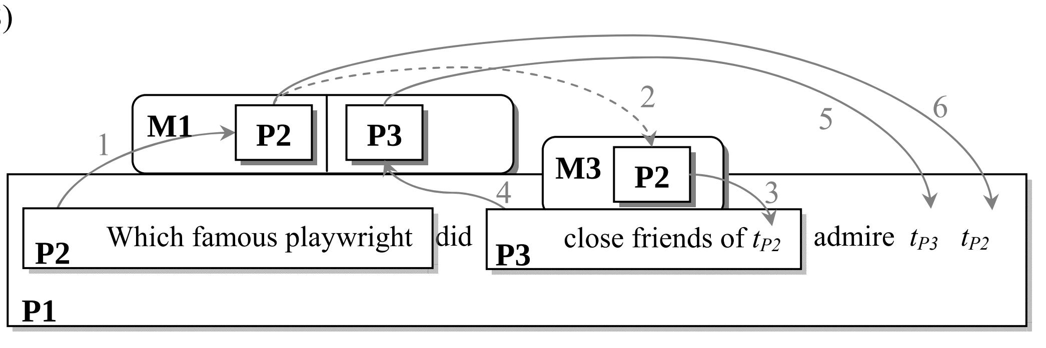
(step 3). However, this step will only empty the local memory buffer of phase 3. We crucially assume that parasitic copying into the memory buffer of the nested phase cannot discharge the memory buffer of the superordinate phase. As a result, even after the "parasitic gap" is computed, the local memory buffer M1 of the yet incomplete matrix phase P1 still contains the wh-phrase. This remains undischarged at the end of the computation, violating the Success Condition (19).

(27)*



On the other hand, the copying mechanism does lead to a successful computation in the case of (1b). As in (27), the "parasitic" copy of the wh-phrase in the memory buffer of the subject DP phase 3 is discharged in the first gap position (step 3); however, the matrix CP phase 1 contains another selected position where the wh-phrase can also be discharged from the memory buffer of phase 1 (step 6). This derivation complies with the Success Condition, as shown in (28). This accounts for the connectedness effect.

(28)



Consider also the more complex configurations in (29) and (30) (from Kayne 1983):

(29) a. ?a person who [cousins of [friends of *e*]] usually end up hating *e*
b. *a person who [friends of *e*]'s parents] usually end up hating *e*

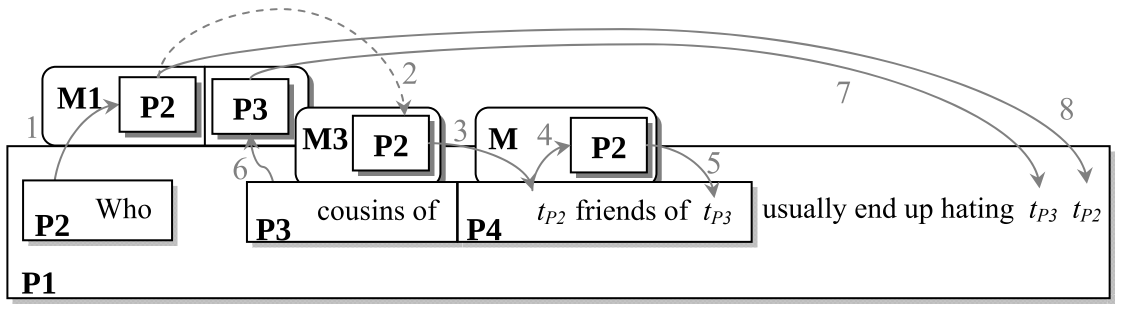
In these examples, the parasitic gap is embedded in a phase within the subject DP phase. However, in the (a) example the embedded phase is selected, hence sequential

material available in phase 3 and the local memory buffer is empty, the system (a.) could backtrack and copy in the memory buffer of phase 3 the content of the buffer of the immediately super-ordinate phase 1; alternatively, (b.) the computation could simply access the first active memory buffer of a super-ordinate phase (i.e. a memory buffer with at least one element inside) and use the first available item without copying it in its memory buffer. The crucial point is that in both solutions "parasitic copying/usage" do not discharge the local buffer of phase 1. Data in (31) suggest the second solution (b.) is not restrictive enough and is empirically inadequate.

to the subject DP phase; in the (b) example, instead, the embedded phase is nested within the subject DP phase (which is itself nested in the relative CP phase).

Consider first a schematic computation of (29a), as represented in (30):

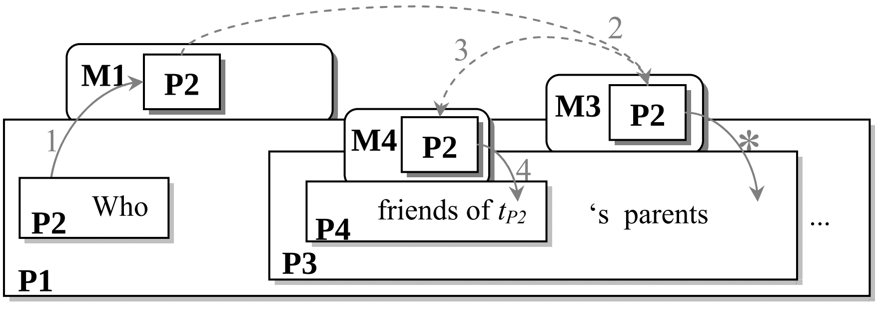
(30)



The relative phrase *who* is first stored in the memory buffer (M1) of the relative CP phase 1 (P1). Then, it is "parasitically" copied in the memory buffer (M3) of the subject DP phase 3 (step 2). When this phase gets closed, the content of M3 is inherited by the next phase sequential P4, that is, the complement of the lexical head *cousins*, via intermediate trace re-merge (step 3); M3 thus remains empty. The relative wh-phrase is then re-stored in the memory buffer of P4 (step 4) and locally discharged in the lowest position selected by the lexical head *friends* (step 5), so that M4 remains empty as well.[24] The computation goes back to the phase P1, and the wh-phrase is also discharged from M1 in the position selected by the matrix predicate (*hating*). All the memory buffers have been emptied by the end of the computation, so that the Success Condition is satisfied.

Consider now a schematic computation for (29b):

(31)



As the dotted lines make clear, here "parasitic copying" must apply twice: first, the wh-phrase is copied from the memory buffer M1 of the relative CP phase onto the memory buffer M3 of the subject DP phase P3 (step 2); second, the wh-phrase is parasitically copied from the memory buffer of P3 into the memory buffer M4 of the unselected phase P4 in pre-nominal position (step 3). The lexical head of P4 (*friends*) selects a complement position where the wh-phrase can be re-merged (step 4); this will empty the memory buffer of P4, but crucially, the buffer of P3 (M3) remains undischarged (since it has been parasitically copied, but not inherited by a phase sequential to P3). Since the head of P3 (*parents*) has no other selected position where

[24] Note that under the "backtracking" view (cf. the preceding note), the system should here backtrack from the doubly embedded phase which contains the parasitic gap to the CP phase 1.

the wh-phrase could possibly be discharged, at the end of the computation of P3 the local memory buffer remains non-empty, in violation of the Success Condition, so that the derivation fails.

To conclude this discussion of left branch islands, let us summarize the main aspects of the proposed analysis. The Connectedenss Condition has been recast in derivational terms, by assuming:

a. a top-to-bottom derivation divided in phases
b. a "storage" conception of the Move operation, which stores an unselected element in the local memory buffer of the current phase and re-merges it in a selected position;
c. a distinction between sequential phases (corresponding to the "canonically governed" branches on the recursive side of the tree) and nested phases (corresponding to the "non canonically governed" branches on the non-recursive side of the tree).

The crucial element in our account of left branch islands is the idea that the content of the memory buffer of a phase can only be inherited by the next sequential phase, and not by a nested phase; in other terms, the content of the memory buffer can be "bequeathed" only after the relevant phase has been completed. In order to account for parasitic gaps licensed under connectedness, we have allowed for the possibility of parasitically copying the content of the buffer of a matrix phase into the buffer of a nested phase; this parasitic copy, however, cannot empty the matrix memory buffer, whence the necessity of *another* (selected) gap within the matrix phase itself (or within a phase that is sequential to the matrix one).

At this point, we return to the problem of right-hand strong islands, which apparently lie on the recursive side of the tree.

4. Right-hand modifiers

As we noted in the introduction, Kayne's original CC only accounts for left branch islands; it does not account for the islandhood of right-hand modifiers, which, however, also show connectedness effects, as was pointed out by Longobardi (1985):

(32) a. ??Those boring old reports, Kim went to lunch [without reading e_i].
b. √ Those boring old reports, Kim filed e_i [without reading e_i].

In order to subsume this kind of data under the CC, Longobardi (1985) proposed to strengthen the notion of g-projection, by requiring that each maximal g-projection be properly governed. We report below the relevant definitions.

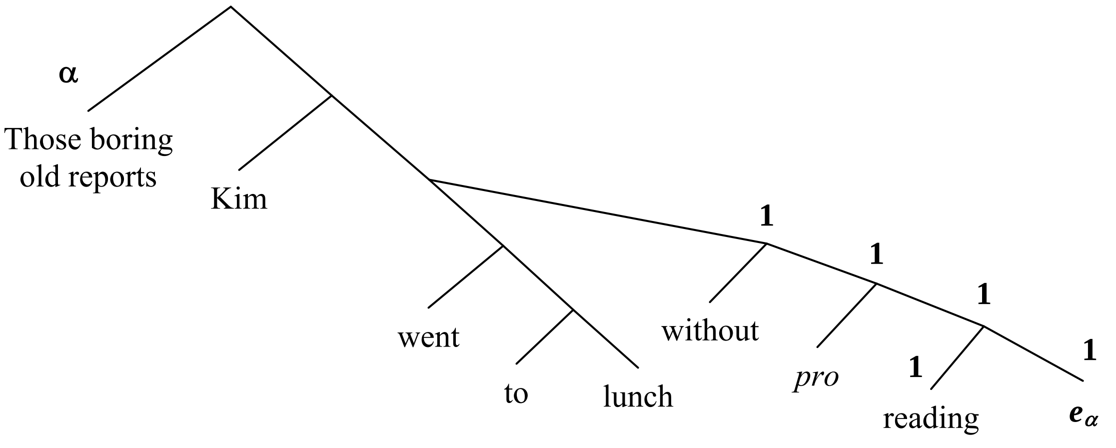
(33) α **governs** β iff γ
i. α is lexical or α and β are coindexed and
ii. β is minimally contained in the maximal projection of α, or β is in the Spec or the head of γ, γ is minimally contained in the maximal projection of α.

(34) α **properly governs** β iff
i. α governs β and
ii. β is (in) a complement (or predicate) of α.

(35) A **non properly governed** maximal projection is a boundary to the extension of g-projections.

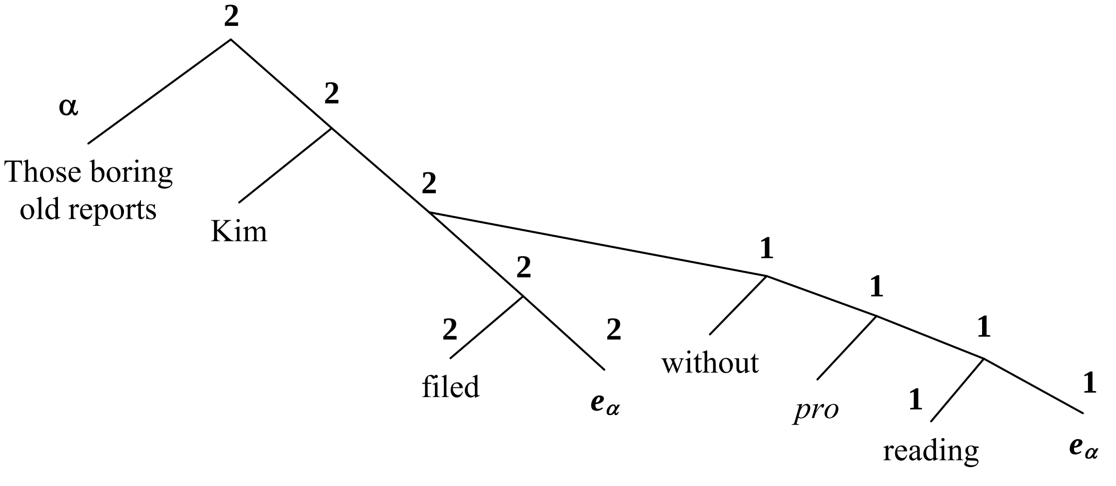
It is easy to see that under these definitions, a right-hand adjunct clause blocks the upward extension of g-projections even though it lies on the canonically governed side of the tree in a VO language like English:

(32)a.*



In (32b), on the other hand, whatever the precise level of adjunction of the adverbial clause, the g-projection path stopping at the top of this clause can connect to that of the VP-internal gap, as shown in the following tree representation:

(32)b.



The adjunct island is thus assimilated to the subject island, much as in Huang's (1982) Condition on Extraction Domains.[25]

There are some problems with this move, though. First, many authors have pointed out that not all right-hand adjuncts give rise to strong island effects, whereas subject islands are much more robust (see for instance Pollard & Sag 1994, 191; Levine & Sag 2003, §3.2; Haider 2003, among many others):

(36) a. Who did you go to Girona [in order to meet *e*]?
b. This is the blanket that Rebecca refuses to sleep [without *e*].
c. How many of the book reports did the teacher smile [after reading *e*]?
(Pollard & Sag 1994)

[25] For recent reformulations of the CED, see Saito & Fukui (1998) and Nunes & Uriagereka (2000).

(37) a. the car that he left his coat [in *e*]
b. the day that she was born [on *e*]
c. *the day that she was born in England [on *e*]

(Haider 2003, 3)

On the basis of this difference, Pollard & Sag (1994) decide not to unify adjunct islands with left branch (subject) islands. Another possibility is to divide the class of right-hand modifiers in two subclasses: true adjuncts (which give rise to strong islands) and "oblique complements", inserted in a complement position (cf. e.g. Larson 1988, 1990), which may be transparent for extraction, or at least, do not constitute real adjunct islands.[26]

Another problem with Longobardi's extension of the CC arises with respect to complex NP islands of the relative clause type.[27] Consider the following examples (from Kayne 1983):

(38) a. *A person who [people [$_{CP}$ that talk to e_i]] usually have money in mind
b. $^?$A person who [people [$_{CP}$ that talk to e_i]] usually end up fascinated with e_i

The connectedness effect in (38b) implies that the relative clause does not block the extension of g-projections up to the NP/DP node which itself constitutes a left branch.

This means that the complex NP island effect on extraction does *not* follow from the blocking of g-projections at the relative CP level, but it must be dealt with by an independent constraint. This was actually an immediate consequence of Kayne's (1983) approach, since a right-hand relative CP is in a canonical government configuration w.r.t. the modified NP. On the other hand, Longobardi (1985) assumes that the relative clause is properly governed because it is predicated of its sister node (cf. definition (34)): this accounts for the possibility of a connectedness effect in (38b), but leaves the complex NP island unaccounted for, as Longobardi himself points out.[28] Thus, even Longobardi's (1985) extension fails to cover all the island effects that have been descriptively classified as "strong islands".

In sum, the variable island effects of right-hand modifiers and the unresolved status of the relative clause complex NP island in the connectedness approach cast some doubt on the idea that right-hand modifiers should be assimilated to left branch

[26] Our approach, exactly like Kayne's (1983) original CC, cannot account for "symbiotic gaps" of the kind discussed by Levine & Sag (2003, §§ 2.3 and 3.2):
(i) What kind of books do [authors of _] argue about royalties [after writing _]
unless the right-hand adverbial clause is of the "transparent" type (cf.(36), (37)); this is actually close to the conclusion drawn by Levine & Sag (§3.2).

[27] As for complex NP islands of the N-complement type, it has been repeatedly noted in the literature that they are weaker islands than the relative clause type. From our perspective, if the CP complement is selected by the N head, then it constitutes a sequential rather than a nested phase (and the island effect does not follow from computational nesting of the CP complement). We leave aside this kind of island effect.

[28] Actually, we suspect that there might be something more going on in examples like (38b). Chomsky (1986, 48 ff.), quoting Chung & McCloskey (1983), discusses the fact that the complex NP island effect is sometimes suspended when the subject position is relativized:
(i) $^?$ This is a paper$_1$ that we need to find [someone who understands e_1]
(ii) ***** This is a paper$_1$ that we need to find [someone that we can intimidate e_2 with e_1]
Chomsky proposes the Vacuous Movement Hypothesis, whereby in (i) the relative operator does not move from the subject position to Spec,CP, and the latter is available as an escape hatch for the extracted phrase. (This is actually equivalent to the GPSG analysis proposed by Chung & McCloskey.) Notice that the connectedness effect in (38b) involves a complex NP island whose relative clause has a relativized subject, much as in (i). We suspect that this might play a role in the acceptability of (38b), but we leave the problem open for the time being.

islands, as in Longobardi's approach (cf. Pollard & Sag 1994, Levine & Sag 2003 for a similar conclusion).

From our present perspective, right-hand modifiers have a somewhat intermediate status w.r.t. our classification of nested Vs. sequential phases. On the one hand, they follow the lexical head of the phase and its complements, so that the superordinate phase is potentially complete; on the other hand, since they are not selected by the lexical head, they cannot be inserted in the computation by an application of Phase Projection (12). In the following discussion we tentatively sketch out a possible treatment of right-hand adverbials as nested phases.

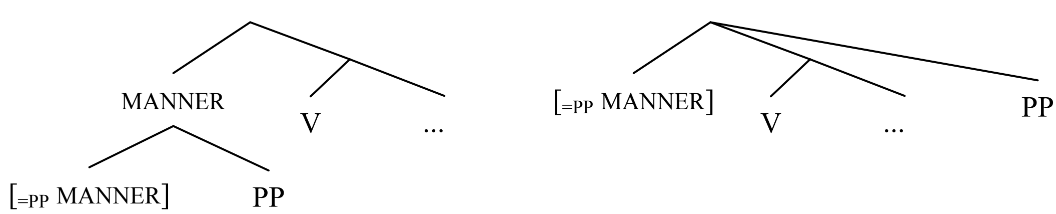
First we will assume, as in the strictly cartographic approach proposed by Cinque (1999, 2002), that each modifier corresponds to a functional specification of the lexical head: namely, in our terms, to a licensor feature. In particular, we assume that a subset of the licensor features (which we can dub Mod(ifier) features) are intrinsically relational, in that they license a relation between a subpart of the lexical head's extended projection and a modifying constituent (e.g. a prepositional phrase). For instance, a MANNER feature placed at a given point of the licensor hierarchy will establish a link between the lower structure and a manner-modifying phrase; a TEMPORAL LOCATION feature will establish a link between an event (or time) denoting portion of the verb's extended projection and a temporal adverbial.

The mediating role of the Mod features is rendered necessary by the fact that, in our system, the modifier cannot directly select the modified portion of clausal structure (as proposed by Gonzalez Escribano 2004): this would in fact interrupt the continuity of the modified phase. It is fair to say that the stipulated Mod features are equivalent to silent functional heads, which mediate the relation between the modifier and the VP. As a concrete implementation of this idea, we assume that a Mod feature, e.g. MANNER, has a selectional specification associated with its (empty) head, e.g.:

(39) $[_{=PP}\ H_{manner}]$

The Mod feature effectively acts as a head: it selects an argument, the PP, giving rise to the configuration presented in (40.a):

(40) a. b.



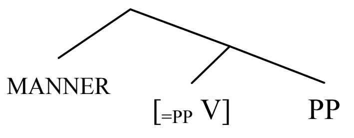
Since manner is a licensor feature of the verb, it has to be computed while this verbal phase is still open: hence, the selected PP constitutes a nested phase.

The right-hand position of the selected PP can be attributed to the special status of the Mod feature. We stipulate that the selected PP can be actually inserted/expanded (by Phase Projection) in the selected position only after the matrix phase select features have been projected (as shown in (40.b)). Note that although the right-hand modifier is selected, it is not selected by the verbal lexical head (unlike Larson 1988, 1990)[29] hence, it is not a sequential phase but a nested one.

[29] In this way the modifier is structurally superior to the VP-internal constituents; this avoids a number of problems with a generalized Larsonian "adjunct as complement" analysis (see Bianchi 1997, 2000, 2001 for detailed discussion).

Moreover, we could capture the unexpected behavior of some right-hand modifiers which seem to be transparent for extraction (cf. (36)-(37)) by assuming a minimal difference between (40.b) above, where the island PP is “selected” by a Mod feature, and (40.c), where the select feature is specified on the verbal head rather than on the Mod feature, so that the PP constitutes a sequential phase.[30]

(40) c.



This is just a sketch of a possible analysis, whose development we leave for further research.

5. Further prospects and conclusions

To summarize, in this paper we have proposed an approach to strong islands and to the connectedness effect within a top-to-bottom derivational framework (formalized in Chesi 2004, 2007).

Among the problems we cannot discuss in these few pages, a major one that immediately springs to mind is the status of phases in a strictly head-final language like Japanese. In such a language all phases, whether selected or not, seem to (linearly and) temporally precede the processing of the superordinate phase’s head; thus, at first sight, they all appear to be "nested". Nevertheless, head-final languages do not block extraction from “selected” pre-head phases; on the contrary, it has been claimed that they even allow extraction from subjects[31] (as is actually predicted by Kayne's original CC, since in these languages left branches are in a canonical government configuration):

(41) a. ?Nani-o$_i$ [John-ga [$_{NP}$ [$_{IP}$ Mary-ga t_i katta] koto]-o mondai-ni siteru] no.
what-ACC John-NOM Mary-NOM bought fact-ACC problem-into making Q
‘What$_i$, John is making an issue out of [the fact that Mary bought t_i].’

b. ?Nani-o$_i$[John-ga [$_{CP}$ [$_{NP}$ [$_{IP}$ Mary-ga t$_i$ katta] koto]-ga mondai-da to] omotteru] no.
what-ACC John-NOM Mary-NOM bought fact-NOM problem-is that think Q
‘What$_i$, John thinks that [the fact that Mary bought t$_i$] is a problem.’
(Saito & Fukui 1998)

This fact could be captured within the proposed model in three alternative ways. One possibility is to propose a parameterization of the Linearization Principle that would allow the selected phases to be linearized to the left of the head; this solution is however not fully satisfactory, since it would predict a generalized transparency of all left-hand constituents (e.g. adjuncts), which is incorrect (Saito & Fukui 1998).

[30] The idea that a selectional specification for a manner PP can be associated directly to a lexical head is made plausible by the existence of “selected adjuncts” (cf. Rizzi 1990): e.g., a verb like *behave* requires a manner specification; a verb like *weight* requires a measure specification of a certain kind; a verb like *be born* requires a locative or temporal specification, etc.

[31] We thank Shoichi Takahashi for discussion of these data.

Another possible approach to pre-head selected phases in head-final languages is to consider them a. properly selected (Chesi 2008) or b. selectors (Choi and Yoon 2006). In the first case (Chesi 2008) the phase head is actually introduced before the arguments even though the phase head is spelled out at the very end of the phase. In the second case (Choi and Yoon 2006) nominal heads select verbal head, against standard assumptions. Obviously all these solutions would deserve more thorough discussion that we cannot carry forward in these pages.

To conclude, we also wish to point out some further consequences of our general approach: first, the top-to-bottom orientation of the computation allows for a relatively straightforward solution to the problem of phase-by-phase linearization, since phases (both nested and sequential ones) are processed in a well defined order, driven by the LICENSOR and SELECT features of the relevant phase heads.[32] Second, the storage conception of Move avoids the "teleological" mechanism of raising to the edge of each phase in case of bottom to top successive-cyclic movement: such moves are teleological in that in the lower phases the final trigger of movement, i.e. the probe/EPP head, has not been inserted yet.[33]

To the extent that our proposal is tenable, it supports a general conception whereby considerations of computational efficiency and cognitive plausibility at the interface with the performance tasks directly constrain the architecture of the grammar itself.

References

Adger D. 2003. *Core Syntax*. Oxford University Press.

Adger D. 2007. *A Minimalist Theory of Feature Structure.* http://ling.auf.net/lingBuzz/000583

Bianchi, V. 1997. On the structural position of time clauses. *Quaderni del Laboratorio di Linguistica* 11, 66-90. Pisa, Scuola Normale Superiore.

Bianchi, V. 2000. On Time Adverbials. *Italian Journal of Linguistics* 12.1, 77-106 (monographic issue "On adverbs and adverbial modification", eds. D. Delfitto & N. Corver).

Bianchi, V. 2001. Antisymmetry and the Leftness Condition: Leftness as anti-c-command. *Studia Linguistica* 55, 1-38.

Chesi, C. 2004. *Phases and Cartography in Linguistic Computation: toward a Cognitively Motivated Computational Model of Linguistic Competence.* Ph.D. Thesis, University of Siena.

Chesi, C. 2007. An introduction to Phase-based Minimalist Grammars: why move is Top-Down from Left-to-Right. *STiL - Studies in Linguistics, CISCL Working Papers* Vol.1, 38-75

Chesi, C. 2008. *Rightward movement from a Top-Down perspective*. Rightward Movement in a Comparative Perspective; Workshop at the DGfS Meeting, February 27-29, 2008. Bamberg

[32] See van Riemsdijk (2005) on the linearization problems in a bottom-to-top computation.

[33] As far as we can see, the problem is not really solved in Fox & Pesetsky's (2004) approach: the wh-phrase moves to the edge of the phase (linearization domain) in order to be linearized to the left of the other internal constituents of the phase; however, the necessity to alter the linearization derives from the ultimate goal of reaching the final left-hand landing site. A common solution is to stipulate uninterpretable wh- or EPP features, which are however justified only by internal constraints of the bottom to top derivation. See McCloskey (2001) for a defense of such features and Rizzi (2002) for a critical discussion.

Choi, Y., Yoon, J. 2006. *Argument Cluster Coordination and Constituency Test (Non)-Conflicts*, NELS 37**,** University of Illinois at Urbana-Champaign.
Cinque, G. 1990. *Types of A' dependencies*. Cambridge, MA, MIT Press.
Cinque, G. 1999. *Adverbs and Functional Heads*. Oxford University Press.
Cinque, G. 2002 *Complement and adverbial PPs: implications for clause structure*. Abstract, University of Venice.
Cinque, G. 2003. Issues in adverbial syntax. *Lingua* 114, 683-710.
Culicover, P. & P. Postal (eds.). 2001. *Parasitic Gaps*. Cambridge, MA, MIT Press.
Fox, D. & David P. 2004. *Cyclic Linearization of Syntactic Structure*. Ms., MIT.
Grimshaw, J. 1991. *Extended projection*. Ms., Brandeis University.
Haider, H. 2003. Pre- and post-verbal adverbials in OV and VO. *Lingua* 114.
Kayne, R. S. 1983. Connect & M. Calcagno. 2001. Parasitic gaps in English: some overlooked cases and their theoretical implications. In *Parasitic Gaps*, ed. P. Culicover and P. Postal, 181-222. MIT Press.
Kayne, R. S. 1994. *The Antisimmetry of Syntax*. MIT Press.
Levine, R., I. A. Sag. 2003. Some empirical issues in the grammar of extraction. In S. Müller (ed.), *Proceedings of the HPSG03 Conference*, Michigan State University, East Lansing. CSLI Publications.
Longobardi, G. 1985. *The theoretical status of the adjunct condition*. Ms. Scuola Normale Superiore, Pisa.
Longobardi, G. 1985b. Connectedness, scope, and c-command. *Linguistic Inquiry* 16.
McCloskey, J. 2001. The morphosyntax of WH-extraction in Irish. J. *Linguistics* 37:67-100.
Munn, A. 2001. Explaining parasitic gap restrictions. In P. Culicover & P. Postal (eds.), 369-392.
Nunes, J. & Uriagereka, J. (2000). Cyclicity and extraction domains. *Syntax*, 3,20-43.
Pesetsky, D. 1982. *Paths and Categories*. PhD thesis, MIT.
Phillips, C. 1996. *Order and Structure*. PhD thesis, MIT.
Pollard, C. & I. Sag. 1994. *Head-driven Phrase Structure Grammar.* Stanford: CSLI.
Postal, P. M. 1994. Contrasting extraction types. *Journal of Linguistics* 30,159-186.
Rizzi, L. 1990. *Relativized Minimality*. MIT Press.
Rizzi, L. 1994. Argument/adjunct (a)symmetries. In Cinque et al. (eds.), *Paths towards Universal Grammar,* 361-376. Georgetown University Press.
Rizzi, L. 1997. The fine structure of the left-periphery. In *Elements of Grammar* ed. Haegeman, L. Dordrecht: Kluwer.
Rizzi, L. 2002. Locality and left periphery. In Belletti, A., ed. (2002*) Structures and Beyond. The Cartography of Syntactic Structures*, vol. 3, Oxford University Press.
Saito & Fukui 1998. Order in Phrase Structure and Movement. *Linguistic Inquiry* 29.3,439-474
Stabler, E. 1997. *Derivational minimalism*. in Retoré, ed. Logical Aspects of Computational Linguistics. Springer
Starke, M. 2001. *Move dissolves into Merge*. Ph.D. Thesis, University of Genève.