Chapter 1
Reference-Set Computation

As mentioned in the introduction, reference-set computation (the selection of the optimal competitor out of a relevant reference set) moved to the forefront of linguistic theory in the 1990s. A restricted version of this process was assumed at the early stages of the minimalist program, and simultaneously, it has been the central notion developed in Optimality Theory. It turned out that none of the original arguments in the early minimalist program actually justify this move, which is the major reason it was eventually rejected. The present assumption in the minimalist framework is that none of the operations of the computational system require reference-set comparisons (section 1.1). But research in this area led to the discovery that there are certain instances where interpretation-based reference-set comparisons are still needed (section 1.2). Originally, these cases were associated with the Minimal Link Condition (MLC). I argue that these cases are not related to the MLC, but restricted instances of reference-set computation are operative at the interface, in areas where the outputs of the computational system do not meet the (contextual) interface needs and adjustments are required. These, indeed, are areas where there are imperfections in the computational system. We may expect therefore that there should also be some observable processing cost associated with these imperfections (section 1.3).

In this chapter, I examine the formal properties of the reference-set type of strategy at the interface, and in the following chapters I turn to the various instances where it applies. To establish the type of computation involved, I begin with a survey of the development of the concept of reference-set economy in the minimalist program, and the reasons it was abandoned.

1.1 The Minimal Link Condition

The early stages of the minimalist program, in Chomsky 1992 and 1994, introduced the concept of economy of derivations. There are two types of economy considerations in that early framework, which are summarized in (1) and (2). (As the theory developed, some of the terminology changed. I quote here from the earliest formulation of these ideas in Chomsky 1992, with later changes noted in brackets.)

(1) "If a derivation D converges without application of some operation, then that application is disallowed" (Chomsky 1992, 47).

(2) *Minimal Link Condition (MLC)*
"Given two convergent derivations D_1 and D_2 [out of the same numeration[1]] ... D_1 blocks D_2 if its links are shorter" (Chomsky 1992, 48).

Condition (1) states that operations are only allowed if they enable a derivation to converge—that is, that derivations are driven only by the need to check features, which, if not checked, will disable convergence. Condition (2) governs the strategies that should apply if there is more than one possible way for a derivation to converge (i.e., there are two or more ways to satisfy feature checking). Chomsky argues that the strategies governed by (1) (which were, at the time, *greed* and *procrastinate*) could be viewed as reducing the computational complexity of the syntax. Given that the second strategy (2) requires comparing derivations and choosing one of them, the more permissible derivations they can select from, the bigger the computational effort is. If the syntactic operations permitted are only those that satisfy (1), the number of permissible (convergent) derivations to compare is dramatically reduced. When there is, nevertheless, more than one way a derivation can converge, (2) requires choosing the shortest one.

The MLC in (2) will be our center of attention in this chapter, because it is this condition that introduces reference-set computation into syntax. A given convergent derivation α is evaluated against a set of alternative convergent derivations: its reference set. If a derivation more economical than α is found in this set, α is blocked. Of course, the reference set should be strictly defined. (We do not want to compare derivations related by some arbitrary notion of similarity.) In a framework assuming syntactic levels, the reference set should include all and only derivations with identical input—that is, the same deep structure. In the minimalist program, syntactic levels were abolished. What guarantees that we compare only

derivations with identical input is the concept of numeration: a derivation starts with a numeration list of all the elements that it will use. Only derivations with identical numeration count as candidates for a reference set.

Let us follow the development of the MLC in (2) and the concept of a reference set for a derivation, through the history of one problem of *wh*-movement, known as *superiority*. It is revealing to examine this problem in detail, because of all the putative instances of the MLC, superiority seemed at first the clearest instance of a restriction that could not be explained locally by conditions on syntactic movement. The question then is whether handling this problem indeed requires reference-set computation, either for capturing the derivation, or for the interpretation of the relevant sentences. (The interpretation question is discussed in section 1.2.) Though the answer in both cases will turn out to be no, the exploration may facilitate understanding the formal properties of reference-set computation, and identifying other instances where it is at work.

Chomsky (1973) noted the contrasts between the (a) and (b) derivations in cases like (3)–(5).

(3) a. Who e discussed what with you?
b. */?What did who discuss e with you?

(4) a. What did Lucie discuss e with whom?
b. */?Whom did Lucie discuss what with e?

(5) a. Whom did Lucie persuade e [PRO to visit whom]?
b. *Whom did Lucie persuade whom [PRO to visit e]?

In the (a) cases, the *wh*-NP that moved originates higher in the tree than the one that stays in situ. If the lower one moves, as in the (b) cases, the derivation is worse. In cases like (3b) that involve just the subject and the object, the violation seems weak, and it has been argued not to exist in all languages. However, things deteriorate with VP-internal arguments in (4b); the movement in (5b), across a clause boundary, is even worse.

Chomsky (1973) assumed that these facts illustrate the operation of a syntactic constraint on *wh*-movement, which he labeled "superiority." The relation "superior" is the predecessor of c-command, and the superiority condition requires that given two or more *wh*-candidates for movement, the one that moves is the superior one, which in later terms means that which c-commands the others.

At the time, this constraint posed a problem, and seemed inconsistent with what was known about syntax. A striking property of the superiority restriction is that there seems to be no way to state it as an absolute constraint on syntactic movement, like the number of syntactic barriers

crossed. To see this, observe the difference between (5b), repeated here, and (6).

(5b) *Whom did Lucie persuade whom [PRO to visit e]?

(6) Whom did Lucie persuade Max [PRO to visit e]?

The distance between *whom* and its trace is precisely identical in the bad example (5b) and the good example (6). This means that the movement of *whom* in (5b) does not violate any island condition, or any absolute prohibition on movement. So well-formedness appears here to be a relative matter: for (6), there is no other candidate for movement, while for (5b) there is. There was no obvious way to state such facts in the syntax of 1973, apart from a descriptive constraint.

Along with the conceptual problem that the superiority constraint seemed to pose, there were empirical problems, which cast doubt on whether this was the correct generalization, and led to the abandonment of the idea. The problems showed up with *wh*-adjuncts, as in (7). The superiority constraint rules out (7a), where *why* presumably originates lower than *who*. But by the same reasoning (7b) should be permitted, which is not the case. In fact, (7b) was felt to have the same status as (7a) or (3b).

(7) a. */?Why did who arrive e?
b. */?Who e arrived why?

(8) a. *Who fainted when you behaved how?
b. Who fainted when you attacked whom?

The judgments are again clearer in cases like (8a). There is no superiority violation here. In terms of syntactic movement, (8a) is identical to the acceptable (8b). Still, when the *wh*–in situ is an adjunct, as in (8a), the derivation gives the same appearance of being a superiority violation.

Huang (1982) observed that the problem in (8a) resembles the problem in (9), where syntactic movement extracts an adjunct out of an embedded clause, violating a constraint known as the ECP.[2]

(9) *How did Max faint when you behaved e?

Based on such facts, Huang argued that all instances of *wh*–in situ must undergo further covert movement at LF to join with the question operator. If this is the case, then the covert movement of *how* in (8a) violates the ECP just as its overt movement in (9) does. Huang's analysis was extremely influential, and the idea that *wh*–in situ must undergo LF-movement gained popularity in the 1980s, when it was believed that such movement is also needed for interpretative reasons.

Huang's hope was that the LF-movement analysis would explain both the superiority and the adjunct effects as instances of ECP violations at LF. On this view, overt syntax movement can apply to any of the *wh*-candidates (subject to standard restrictions on syntactic movement), but at LF, all other *wh*-elements must raise. In the specific implementation of Huang, *who* of (3b), repeated below, adjoins to *what* in SpecCP, as in (10). From that position it does not c-command its trace (since the index of this Spec remains that of *what*). So the trace of *who* is not antecedent governed—violating the ECP. The same is true for (3a), repeated below, but there the trace is head governed, hence the ECP permits the derivation.

(3) a. Who e discussed what with you?
b. */?What did who discuss e with you?

(10) LF of (3b): *[who$_1$ [what$_2$]]$_2$ [e$_1$ discussed e$_2$ with you]

This account captures correctly all adjunct cases, since adjuncts always require antecedent government, and it also happens to capture superiority with subjects, as in (3b). What has gone unnoticed, though, is that it leaves the other superiority cases (4) and (5) unexplained—for example, the LF of (5b), repeated below, should be (11) in Huang's system. In this LF-derivation, the trace is appropriately head governed. Hence it is not ruled out by the ECP.

(5b) *Whom did Lucie persuade whom [PRO to visit e]?

(11) LF of (5b): [whom$_1$ [whom$_2$]] [Lucie persuaded e$_1$ [PRO to visit e$_2$]]

Though several other implementations of the LF-movement approach exist, it remained the case that this approach did not solve the full range of the superiority problem.

In the minimalist program (starting with Chomsky 1992), Chomsky returned, in a sense, to the analysis of Chomsky 1973. Regardless of whether covert LF-movement of *wh*-constituents is still independently needed, Chomsky argued that superiority is a restriction on overt movement—an instance of the economy strategy (2) of preferring shorter links. Traveling to SpecCP, the c-commanding *wh* has to cross fewer nodes that dominate it than any *wh* it c-commands. Hence the movement in the (a) cases of (3)–(5) is more economical than that in the (b) cases.

This may appear to leave us precisely where we started, with the problem of *wh*-adjuncts unsolved. However, Tsai (1994) and Reinhart ([1994] 1998) argued, on different grounds, that this problem is, indeed,

independent of the problem of superiority with *wh*-arguments. Note first that the problem in (8a), repeated below, is not a general problem with *wh*-adjuncts, as assumed by Huang, but is restricted to adverbial *wh*-phrases. Example (12), in which *how* is replaced with *what way*, is fine. Syntactically and semantically, the *wh*-phrase is an adjunct in both. Still, only the adverbial adjunct causes problems.

(8a) *Who fainted when you behaved how?

(12) Who fainted when you behaved what way?

I argued in Reinhart [1994] 1998 that the standard interpretation of instances of *wh*–in situ involves no LF-movement, and they are interpreted in situ by a mechanism of choice functions, whose details I will examine in chapter 2. But adverbial *wh*-elements cannot be interpreted this way. One thing that would be agreed on in all frameworks is that *wh*-adverbials are different from *wh*-NPs, first, because they do not have a common noun set (N-set), and second, because they denote functions ranging over higher-order entities (Szabolcsi and Zwarts 1990). This means that choice functions selecting an individual from a set cannot apply to them (since there is no set of individuals that the choice function could select from). In (12), the adjunct that stays in situ is still an NP, hence it is interpreted in situ by applying a choice function. But in (8a), the same procedure cannot apply. Adverbial *wh*-expressions, then, pose a specific problem, because they are uninterpretable in situ.

Two routes are open to proceed from this observation. One is that *wh*-adverbials in situ, and only they, must indeed undergo LF-movement in order to be interpreted. Hence, Huang's (1982) account still holds for such adverbials, and (8a) is an ECP violation. Another route is to pursue the alternative account offered for the problem in Reinhart 1981b, namely, that such adverbials are, in fact, base generated in SpecQP, hence (8a) cannot be generated. The analysis assumed two Specs, which would correspond in current syntax to CP and QP, and among the arguments for base generating adverbials in SpecQP was the fact that we never find more than one such adverbial per clause. While (13a), which could be obtained by some sort of scrambling of the adverbial to final position, is marginal, (13b) is completely out.

(13) a. ?Who spoke how?
b. *Who spoke when how?

Either way, we may conclude that the problem of *wh*-adverbials is independent of superiority, and the latter indeed reflects, a restriction on overt

syntactic movement. The road is open, then, to pursuing Chomsky's (1992) assumption that superiority is an instance of the MLC in (2)—in other words, that it requires reference-set computation.

While at the previous stage, in 1973, the superiority condition seemed arbitrary and structure-specific, in the early 1990s the MLC was believed to govern a broad spectrum of facts. It was intended to entail the relativized minimality effects of Rizzi 1990, as well as minimizing the number of chain-formation operations, in cases discussed by Epstein (1992) and Collins (1994). Let us look more closely at the intuition behind (2), and its implications for the theory of syntax.

At the transitional stage between the principles-and-parameters framework and the minimalist program, it was noted that certain, apparently distinct, constraints on syntactic movement have something in common that could be characterized as "least effort." Following Rizzi's relativized minimality, it was felt that what the bad derivations in (14) have in common is that the (italicized) moved element skips an (underlined) potential landing site, which is closer to the original position of the moved element, so, in some sense, the movement is "longer" than necessary.

(14) *Relativized minimality*
   a. Head movement (HMC): *Where *find* Max will t the book.
   b. A-movement (superraising): **Max* seems [that it is certain [t to arrive]]
   c. A′-movement (*wh*-islands): *I wonder *what* you forgot from whom you got t t.

In the superiority cases that we discussed, there is no intervening landing site. Still, the derivations seem longer than necessary, since to check the *wh*-features of C, the *wh*-element closer to it could move.

In the first implementation of the minimalist program (MP) (Chomsky 1992, 1994), movement was motivated by the need of the moved element to check its features ("greed"). Under this implementation, it was not possible to state the "shorter-link" intuition locally. For example, in (14b), once we select and merge *it* in the second cycle, there is no shorter way for *Max* to check its case or DP features. Similarly, from the perspective of the *wh* that moved in (3b) (**/?What did who discuss e with you?*), the route it took is the only (hence the shortest) way to check its own features. Capturing this intuition required, therefore, comparing a set of competing convergent derivations, which was later labeled the *reference set*. The MLC condition (2), repeated here, is based on constructing such a set.

(2) *Minimal Link Condition (MLC)*
"Given two convergent derivations D_1 and D_2 [out of the same numeration] ... D_1 blocks D_2 if its links are shorter" (Chomsky 1992, 48).

For the superraising case in (14b), the relevant reference set is the pair ⟨15a, 15b⟩, which contains two possible derivations from the same numeration (the same "deep structure," in the previous model).

(15) a. [(F) It seems that [(F) Max_i is certain [t_i to arrive]]]
b. *[(F) Max_i seems that [(F) it is certain [t_i to arrive]]]

In (15a) *Max* moves in the second cycle (of *certain*), to check the feature F, and in the higher cycle *it* is merged. Example (15b) is (14b). Since the link between *Max* and its trace is shorter in (15a) than in (15b), (15a) blocks (15b). Similarly, the reference set for the superiority violation in (5b), repeated in (16b), is the pair ⟨16a, 16b⟩. Derivation (16a), with the shorter link, blocks (16b).

(16) a. Whom did Lucie persuade e [PRO to visit whom]?
b. *Whom did Lucie persuade whom [PRO to visit e]?

We noted that a characteristic property of the superiority restriction is that it is impossible to state it as an absolute condition in terms of the distance between the original position and the target position of movement. The distance between these positions is identical in the problematic (16b) and in the innocent (6), repeated in (17).

(17) Whom did Lucie persuade Max [PRO to visit e]?

This is precisely the type of property that can be explained by assuming reference-set computation. For (17), there is no alternative derivation that will satisfy the *wh*-feature (no alternative convergent derivation), so it is the single member in its reference set and hence, the shortest possible derivation in this set.

Let us reflect now on the formal properties of the computation we have been assuming. The characteristic properties of reference-set computation are that it assumes a relative concept of well-formedness (as we saw), and, next, in the specific instances under consideration, that it requires global computation. In (15), for example, it is useless to construct a reference set locally, at the second (*certain*) cycle, since the effects of either inserting *it* or moving *Max* are only noticeable at the next cycle. So the whole derivation must be kept open and available at that top cycle. As pointed out by Collins (1997), the problem is more general. Since (4) requires compar-

ing only convergent derivations, the construction of the reference set is only possible at the very end, where nonconvergent derivations can be filtered out. My focus of attention here is on instances of reference-set computation with this second property of requiring global computation. (Other instances may involve local reference-set computation, which does not raise the questions I will be turning to.)

Optimality Theory (OT), which developed in about the same period, is based on the same notion of global reference-set economy, with these two properties, though the technical details of the implementation are different. But the OT system is much richer, assuming, first, that what needs to be checked against a reference set is not just which derivation is shorter, but a currently open list of constraints, and next, that these constraints are ranked, with possible variations of the ranking across languages.

The global nature of reference-set optimality poses a problem if we assume that the parser is essentially transparent, in the sense outlined in the introduction—that is, that it actually implements (a subset of) the computations required by the CS, with minimum parser-specific computations. If we translate the computation into actual processing terms, it requires, first, holding all nodes in the derivation accessible in working memory, until the full derivation can be completed, and at the same time constructing (or attempting to construct) alternative derivations, with which to compare the stored material. The type of load on working memory assumed here exceeds what is known to be realistic for the human parser. The assumption shared by all processing studies (since, at least, Fodor, Bever, and Garrett 1974) is that given the limitations of working memory, the human processor attempts to close constituents as soon as possible. Chunks of the derivation that are closed are assigned some abstract representation, and the nodes they dominate are no longer available for subsequent processing. Opening a closed constituent to access its subparts is possible but can be highly costly, leading to a garden-path effect. If the parser requires global reference-set computation, either nothing gets closed and eventually the overload is too great for processing (as in the case of center embedding), or constituents constantly close and reopen (garden-path effect). Neither option is consistent with the fact that in actual language use, sentences ordinarily get processed smoothly. The least we can infer is that the human parser does not operate, in processing, by computations of this kind.

An approach developed to address this problem, particularly in the OT framework (though it is also still found in the earlier parts of Chomsky

1995), is that one should not attempt to deduce the properties of the computational system (competence) from properties of the parser (performance). The actual processing of derivations need not literally compute optimality, but rather some algorithms, or heuristic strategies, are developed by speakers for a quick assessment. (For some algorithms proposed for acquisition, see Pulleyblank and Turkel 1998 as well as Tesar 1998.) This approach cannot yet be evaluated, given that the full range of algorithms guiding the parser still needs to be specified. But rather than dwelling on this point, we may note its implications for the hypothesis of optimal design outlined in the introduction, based on Chomsky 2000. Suppose we have successfully defined a computational system that is an optimal solution to the elementary interface conditions, but it still fails the other conditions—for example, it is not fully adequate for processing with limited working memory, so we have to add many parser-specific algorithms that enable it to bypass the required computation of the CS. This would mean that the optimal-design hypothesis is false and human language is not optimally designed. If reference-set computation is found only in isolated cases governed by the MLC, as is the case in the early minimalist program, this does not constitute a complete failure of optimal design, as in Optimality Theory, because the problem is confined to specific areas. Nevertheless, it is appropriate to check further whether it really needs to be assumed even in these isolated cases. The question at stake is whether syntax—the computational system—includes (even restricted) computations of this kind.

In fact, it turned out that there was no real motivation to assume the complex computation of the MLC in (2), since whatever is correct about the intuition of "least effort" or "shortest move" can also be captured by a local computation. In chapter 4 of Chomsky 1995, both the views on what triggers movement and on the MLC are revised. *Greed* is replaced with *attract*: movement is not triggered by the requirements of the moving element, but by the higher (functional) category, which needs this element in order to be interpreted or deleted. This enabled building the MLC into the definition of *attract*.

(18) *"Attract" (combines last resort and MLC)*
"K attracts F if F is the closest feature that can enter into a checking relation with a sub-label of K" (Chomsky 1995, 297).

From the perspective of the attracting target, there is nothing complex about finding its nearest candidate. Suppose we reach a stage in the derivation where a functional category (a feature) has been merged. At this

point we search in the chunk of the derivation we have just built, for the necessary element to check it, and the search stops as soon as the first such element (going from top to bottom) is found. For example, in the superiority cases of (16), repeated here, the relevant state of the derivation is (19), where the *wh*-feature has just been merged at the matrix.

(16) a. Whom did Lucie persuade e [PRO to visit whom]?
b. *Whom did Lucie persuade whom [PRO to visit e]?

(19) Q+*wh* [Lucie persuade whom [PRO to visit whom]]

This feature now attracts the nearest *wh*-element it can find, which is the complement of *persuade*. Hence (16a) is derived, and there are no further options for continuing the search that could derive (16b). In (17), repeated below, the first *wh* that can be found is the complement of *visit*, hence it is this one that is attracted. I will return shortly to the cases of relativized minimality in (14).

(17) Whom did Lucie persuade Max [PRO to visit e]?

The MLC on this view is not a relative condition, but an absolute one. The first relevant element must be selected, regardless of any other considerations that may have tempted us to do otherwise. On this formulation, no reference set is constructed at all—(16b) is not ruled out by comparison to alternative options, but it is underivable. The MLC is also local, in the sense that it applies as soon as the attracting node has merged, with no need to know about any potential future steps in the derivation.

This is the place to note that there has always been something puzzling about the view of the original MLC as a "least-effort" or economy principle. An extremely costly computation, which exceeds standard processing limitations, was needed to save the effort posed by a longer link than necessary. On the other hand, under the present formulation it is possible to observe that this absolute condition is indeed a "least-effort" condition in terms of actual processing. It minimizes the search for a checking element, thus enforcing the quickest possible conclusion of the given step in the derivation and freeing working memory for the next task.

There is still a difference between the revised MLC and the other absolute conditions on syntactic movement, which prevent movement out of an island. The latter define the limits of the search—the domain beyond which a functional category cannot attract elements to check it. For example, when the Q-feature is merged in (20a), it starts the search for a *wh* that it can attract. However, the search cannot reach into the syntactic

island, which is why (20b) cannot be derived. Hence no *wh*-feature can be attracted, and a derivation starting with this numeration has no way to converge. The same is true for the CED island in (21).

(20) a. Q+*wh* [you resign [after Max behaved (in) what way]]
b. *In what way did you resign after Max behaved t?

(21) *Which shelf did you borrow the books on t?

In terms of processing, islands correspond to units that have been closed and stored at the stage of the derivation where the attractor is introduced. Their unavailability, again, decreases the load on working memory.

The properties of the computational system that emerge out of this view of "least effort" provide no evidence for a need to impose "imperfections," such as an altogether separate parser, or processing algorithms. On the contrary, the revised MLC and the island restrictions appear to be conditions enabling the computational system to match the processing limitation of human users—that is, the limitation of working memory. The computation is local, which means that only chunks of the derivation that are actively at work need to be retained in working memory; syntactic islands define the absolute limit for search operations, and the revised MLC imposes further acknowledgment of this limitation, forcing the quickest conclusion of operations required in a given step in the derivation.

Conceptual issues aside, the reasons the global reference-set approach was discarded in the minimalist framework are also empirical. Even for the small corpus examined here, we can see that the version of the reference-set MLC, as stated in (2), yields the wrong results in the case of *wh*-islands. (This was pointed out in Reinhart [1994] 1998.)

The reference set for (14c), repeated in (22c), is ⟨22b, 22c⟩. (In terms of "deep structure," (22b, c) are both derived from (22a).) Recall how this was determined: with the numeration used in (22c), we could obtain all three derivations in (22), as well as several others. However, only the derivations in (22b, c) converge: in (22a), as well as in the other conceivable options, the *wh*-feature is not checked.

(22) a. I wonder [Q+*wh* [you forgot [Q+*wh* [you got what$_j$ from whom$_i$]]]]
b. *I wonder [from whom$_i$ [you forgot [what$_j$ [you got t$_j$ t$_i$]]]]
c. *I wonder [what$_j$ [you forgot [from whom$_i$ [you got t$_j$ t$_i$]]]]

Given the reference-set MLC as stated in (2), there are now two possible conclusions: either we decide that the two derivations have equally short

links, or one of them is shorter than the other. (Computing here is not simple, but nothing hinges on deciding this.) In the first case, both derivations should be allowed; in the second, one of them (the shorter one) should be permitted. Both these conclusions are wrong. This in itself does not prove that the idea of reference-set economy in the computational system is wrong, since one may reasonably argue that *wh*-islands are governed by an independent absolute constraint. Nevertheless, the problem illustrates the danger of using such a strategy freely.

The account suggested in chapter 4 of Chomsky 1995 for these cases rests on another option of satisfying "attract" in (22), which we have overlooked so far. Suppose *what* moved to check the *wh*-feature of its clause as in the first step of (22b). When the next Q+*wh* is merged and looks for a feature to attract, the nearest one it can find is this same *what*. Hence the "attract" version of the MLC in (18) determines that this is the only option, and *what* must move again. Thus the only derivation permitted from this numeration (from the "deep structure" in (22a)) is (23).

(23) I wonder [what$_j$ Q [you forgot [t_j Q [you got t_j from whom$_i$]]]]

The assumption is that (23) indeed converges, in the sense that all relevant features are checked, but it is semantically defective.[3] Similar reasoning applies in the case of superraising in (14b), though it entails some further complications.[4]

We should note that this specific account of *wh*-islands and superraising is a matter of implementation, which is being continually revised in the MP framework. Another possibility, suggested in Reinhart [1994] 1998, is that these are not, in fact, instances of the MLC, even in its present formulation, but they follow from other conditions.[5] An issue still open in the MP is the precise account of syntactic islands (which originally also included *wh*-islands). The decision regarding the division of labor between the MLC and other conditions must await such an account.

Either way, it is clear that none of the cases that originally motivated the introduction of reference-set computation into the computational system justify this move. If anything, they show that such a computation is not, in fact, available in this system.

1.2 Interpretation-Dependent Reference Sets

Though it was found irrelevant for syntax, the concept of reference-set computation, in the early minimalist program, inspired a line of research

on its role at the interface of the computational and the conceptual systems. Interestingly, the first formulations of reference-set strategies at the interface also evolved around the earlier version of the MLC in the area of superiority. So let us first trace this development.

There are a residue of facts noted over the years that pose problems for any analysis of superiority effects. One such problem, noted by Lasnik and Saito (1992), is given in (24). Example (24a) is a standard superiority violation (the lower rather than the higher *wh*-phrase has moved). But (24b), where precisely the same thing happens in the embedded clause, is much better.

(24) a. */?I know [what [who bought e]]?
b. Who e knows [what [who bought e]]?

(25) a. Who e knows who e bought what?
b. *Lucie* does. (= Lucie knows who bought what.)
c. *Lucie* knows who bought a *car* ...

(26) a. Who e knows what who bought e?
b. **Lucie* does. (= Lucie knows what who bought e)
c. *Lucie* knows what *Max* bought ...

(27) a. For which $\langle x, y \rangle$, x knows what y bought.
b. For which x, x knows for which $\langle z, y \rangle$, y bought z.

As Lasnik and Saito noted, this is only possible if *who* has matrix scope. In principle, sentences with this structure have two scope construals, as seen in (25), which does not involve a superiority violation. If the *wh*–in situ (*what*) takes scope in the lower clause, a possible answer would be (25b); if it takes scope in the top clause, the answer will have the form of (25c). (The italicized constituents correspond to the *wh*-constituents that are being answered. For independent reasons, a *wh*-constituent in SpecCP cannot have scope beyond that CP, so there is no additional scope construal for the question.) By the same token, (24b) should also be ambiguous regarding the scope of the embedded *who*, but it is not. As we see in (26), it cannot be answered with (26b), which is obtained by interpreting *who* with scope over the embedded clause, but only with (26c), which corresponds to the higher scope construal. In other words, of the two informal scope representations in (27), (26a) can only be construed as (27a). (I ignore here the precise details of the interpretation of questions and of *wh*–in situ, issues that are discussed in Reinhart 1992, [1994] 1998.)

Golan (1993), followed by Reinhart [1994] 1998, argued that to capture such facts, we need to assume that the MLC, which is behind the superi-

ority effects, is interpretation-dependent—that is, it determines the most economical derivation relative to interpretative goals. In the standard bad instances of superiority violations, the derivations with the long and the short movement yield precisely the same question. For example, derivation (28a), which violates superiority, results in (28b), which is precisely the same as (29b), obtained by the shorter derivation (29a). In this case the more economical derivation (shorter link) blocks the other.

(28) a. *What did who buy e?
b. For which ⟨x, y⟩ x bought y.

(29) a. Who e bought what?
b. For which ⟨x, y⟩ x bought y.

In the problem case (24b), repeated in (30a), the derivation appears to violate the MLC as well, since a shorter derivation exists, as in (25a), repeated in (31a), where the c-commanding *who* is moved.

(30) a. Who e knows what who bought e?
b. For which ⟨x, y⟩, x knows what y bought.

(31) a. Who e knows who e bought what?
b. For which ⟨x, z⟩ x knows who bought z.

But in this case the questions denoted by these two derivations are not identical. With a matrix scope of the *wh*–in situ, (30b) asks for a value for *who*, while (31b) asks for a value for *what*.[6] So, if we try to ask the question (30b), there is no other, more economical derivation that could arrive at this question. Hence, this is the most economical way to reach an interface goal.

The line of argument in Reinhart [1994] 1998 is that considerations of this type apply at the stage of translating syntactic forms into semantic representations. (It is not necessarily full semantic representations that need to be checked, but some representation in which variables are introduced and bound.) The way it was stated in Reinhart [1994] 1998, if at the stage of translating a given convergent derivation D into some semantic representation, we discover that an equivalent semantic representation could be obtained by a more economical derivation D′ (from the same numeration), D′ blocks D. (That is, D′ blocks D unless their translations are not equivalent.) I argued further that under this view, the computation found in superiority has properties similar to the strategy that I proposed in Reinhart 1983 for the coreference aspects of conditions B and C. Abstracting away from the technical details (which are worked out in Grodzinsky and Reinhart 1993), the coreference generalization is

that two expressions in a given LF, say D, cannot corefer if, at the translation to semantic representations, we discover that an alternative LF, D′, exists where one of these is a variable bound by the other, and the two LFs have equivalent interpretations. In other words, D′ blocks coreference in D, unless they are semantically distinct. (I will return to this strategy in chapter 4.)

But this formulation of the computation involved here is somewhat vague. Fox (1995) proposed a precise formal statement of this intuition. He built it into the definition of the reference set, and at that stage it was applicable only for interface strategies governed by the MLC: the set out of which the MLC selects the most economical derivation includes only derivations that end up with the same interpretation. Technically, this means that the reference set consists of pairs $\langle d, i \rangle$ of derivation and interpretation, where the interpretation *i* is identical in all pairs. A given $\langle d_i, i \rangle$ pair is blocked, if the same interface effect could be obtained more economically—that is, if the reference set contains another competitor $\langle d_j, i \rangle$, where d_j has a shorter link.

For illustration, again consider the reference set for (28a), repeated below. It consists of the pair ⟨32a, 32b⟩. Each member of this pair is itself a pair of a derivation and the interpretation assigned to it. The reason both derivations are included in the reference set is that (they start with the same numeration and) their interpretation member is identical. In this reference set, the link in the d-part of (32b) is shorter than that in (32a), hence (32a) is ruled out by (32b).

(28a) *What did who buy e?

(32) { a. ⟨What did who buy e, for which ⟨x, y⟩ x bought y.⟩
       b. ⟨Who e bought what, for which ⟨x, y⟩ x bought y.⟩ }

The acceptable superiority violation in (24b), repeated here, contains only one member in its reference set—the $\langle d, i \rangle$ pair based on (27a), repeated here. This is so because, as we saw, no other derivation out of the same numeration has the same interpretation. Hence, 24b is the most economical derivation (relative to this interpretation).

(24b) Who e knows what who bought e?

(27a) ⟨Who e knows what who bought e, for which ⟨x, y⟩, x knows what y bought.⟩

This approach, then, retains the earlier view of the MLC as a selection out of a reference set, but restricts the set further by interpretative considerations.

Fox (1995) and Reinhart [1994] 1998 argued that QR as well is sensitive to reference-set computation. In Fox's implementation, QR obeys the MLC, under this interpretation-sensitive formulation.[7] I will return to this question in greater detail in section 2.7, but let us just follow the gist of the idea here.

Following the tradition in the LF-theory of the principles-and-parameters framework, and in Heim and Kratzer 1998, Fox assumes that all non-subject-quantified NPs necessarily undergo QR at LF. Whether their final scope would correspond to their overt position depends on where they move to at LF. Thus, a sentence like (33) is ambiguous regarding whether *every patient* has narrow scope, as determined by its overt syntactic position, or it scopes over *a doctor*.

(33) A doctor will examine every patient. (Ambiguous)

(34) a. A doctor$_2$ [e_2 will [$_{VP}$ every patient$_1$ [$_{VP}$ examine e_1]]]
(There is a doctor x, such that for every patient y, x will examine y.)
b. Every patient$_1$ [a doctor$_2$ [e_2 will [$_{VP}$ examine e_1]]]
(For every patient y, there is a doctor x, such that x will examine y.)

The narrow-scope interpretation is obtained by raising the quantified object just to the VP, as in (34a). (This is the position proposed for raised VP-internal quantifiers in May 1985.) The wide scope of *every patient* is obtained by movement to the topmost IP position, as in (34b).

Assuming that a quantified VP-internal argument can adjoin either to VP or to IP to be interpreted, it appears that the MLC should determine that only the first is allowed in practice, since the link between the quantifier and its trace is shorter in (34a) than in (34b). However, if the MLC does not compare just derivations, but $\langle d, i \rangle$ pairs of a derivation and its interpretation, the movement in (34b) is licensed, since it yields a distinct interpretation from the shorter derivation in (34a). Hence, the reference set of (34b) contains only this derivation.

Fox provides impressive evidence for this view of the MLC. His point of departure is a puzzle noted by Sag (1976) and Williams (1977). Although (33), repeated here, is ambiguous, as we just saw, the ambiguity disappears in the ellipsis context of (35).

(33) A doctor will examine every patient. (Ambiguous)

(35) A doctor will examine every patient, and Lucie will [ ] too. (Only narrow scope for *every*)

When (33) occurs as the first conjunct of the ellipsis, it allows only the narrow scope for *every patient*, represented in (34a) (i.e., (35) is true only if there is one doctor that will examine all the patients).

The account Sag and Williams offered for this fact is based on their assumption that VP-ellipsis is an LF-operation: an LF-predicate is copied into the empty VP (at least in Williams's analysis). The predicate should be well formed, and, specifically, it cannot contain a variable bound outside the copied VP. Now let us look again at the two LFs generated for (33), repeated in (36a, b).

(36) a. A doctor$_2$ [e$_2$ will [$_{VP}$ every patient$_1$ [$_{VP}$ examine e$_1$]]]
b. Every patient$_1$ [a doctor$_2$ [e$_2$ will [$_{VP}$ examine e$_1$]]]
c. And Lucie will [ ] too.

The second ellipsis conjunct is generated, as in (36c), with an empty VP, into which an LF-VP should be copied from the first conjunct. If we copy the (top) VP of (36a) ([$_{VP}$ *every patient*$_1$ [$_{VP}$ *examine* e_1]]), the result is well formed. But the VP of (36b) is [$_{VP}$ *examine* e_1]. This VP contains the trace of *every patient*, which is bound outside the VP. Hence this is not an independent well-formed predicate, so it cannot be copied. It follows, then, that only the LF (36a) allows interpretation of the ellipsis, hence in (35) there is no ambiguity.

Sag and Williams viewed this as strong evidence for their LF-analysis of ellipsis. However, Fox, also citing Hirschbühler 1982, points out that this could not be the correct explanation, based on examples like (37).

(37) A doctor will examine every patient, and a nurse will too.

Unlike (35), (37) is ambiguous—that is, the ambiguity of the first conjunct is not canceled in the context of ellipsis. Example (37) differs only minimally from (35) (*a nurse*, instead of *Lucie*). So the question is why that minimal difference should matter. Though there have been many attempts at an answer since Hirschbühler pointed the problem out, it remained, essentially, a mystery.

Fox's solution rests on the alternative view of ellipsis as a PF-deletion developed in the minimalist program (see Chomsky and Lasnik 1993 and Tancredi 1992 for some of the details). The inputs of VP-ellipsis, then, are two full derivations (clauses). Then one of the VPs is "deleted"—that is, it is not spelled out phonetically. This is subject to parallelism considerations, which also may affect other PF-phenomena, like deaccenting. The least we know about what counts as parallel derivations is that all LF-operations, like QR, that apply to one of the conjuncts should apply also

to the other (though many additional considerations may play a role). Let us see, for example, how (37) is derived, under the construal of *every patient* with wide scope.

(38) a. Every patient$_1$ [a doctor$_2$ [e$_2$ will [$_{VP}$ examine e$_1$]]] and
b. Every patient$_1$ [a nurse$_2$ [e$_2$ will [$_{VP}$ examine e$_1$]]] too.

Both conjuncts are derived in full, as in (38). QR has applied, independently to both. The result, then, is that the two VPs are precisely identical, and the second one need not be realized phonetically, so the PF is the string in (37). If QR does not apply in precisely the same way to both conjuncts, no ellipsis is possible, as witnessed by the fact that (37) cannot have different scope construals in the first and second conjuncts.

The question, now, is why the same is not true also for (35). For ellipsis to be possible under the wide-scope construal of *every patient*, QR should apply in both conjuncts, as in (39). For convenience, in the following examples I ignore the LF-movement of the subject argument. If QR applies freely, as in the standard view, this should be possible, and there is, again, no explanation for why this reading is impossible for the ellipsis in (35).

(39) a. Every patient$_1$ [a doctor will [$_{VP}$ examine e$_1$]] and
b. Every patient$_1$ [Lucie will [$_{VP}$ examine e$_1$]]

(40) a. [A doctor will [$_{VP}$ every patient$_1$ [$_{VP}$ examine e$_1$]]] and
b. [Lucie will [$_{VP}$ every patient$_1$ [$_{VP}$ examine e$_1$]]]

This is where the interpretation-dependent MLC enters the picture. The intuitive idea is that the MLC determines that the longer-link QR (outside of the VP) applies only if this is required to obtain an interpretation not available otherwise. The problem in (39) lies in the second conjunct. The movement of *every patient* here is longer than necessary for interpretation. In (38b) long-distance QR results in a different interpretation than that obtained if *every patient* is assigned scope inside the VP. But in the case of (39b), the reading obtained by long-distance QR is equivalent to the reading obtained in (40b) with the shorter movement, so there is no interpretative need that could motivate the longer movement. This can be observed by examining the reference set for (39b), given in (41).

(41) $\left\{ \begin{array}{l} \text{a. } \langle \text{Every patient}_1 \text{ [Lucie will [}_{VP} \text{ examine e}_1\text{]]}, \\ \quad \text{For every patient x, Lucie will examine x} \rangle \\ \text{b. } \langle \text{[Lucie will [}_{VP} \text{ every patient}_1 \text{ [}_{VP} \text{ examine e}_1\text{]]]}, \\ \quad \text{For every patient x, Lucie will examine x} \rangle \end{array} \right\}$

Since the interpretation is identical in (41a) and (41b), the reference set includes both derivations. The shorter-link derivation (41b), then, blocks the derivation (41a). Correspondingly, the only construal permitted in the second conjunct is (41b). Returning to (40), parallelism determines, therefore, that for ellipsis to be allowed, the first conjunct should have the same LF-structure, hence, only (40) is the source of ellipsis in (35).

In the case of (38), repeated below, the long-distance movement of *every patient* in the second conjunct yields an interpretation distinct from the shorter movement (inside the VP). This is so because *every patient* moves across an existential quantifier—*a nurse*—and whether it is inside or outside the scope of this quantifier has interpretative consequences.

(38) a. Every patient$_1$ [a doctor$_2$ [e$_2$ will [$_{VP}$ examine e$_1$]]] and
b. Every patient$_1$ [a nurse$_2$ [e$_2$ will [$_{VP}$ examine e$_1$]]] too.

Hence the reference set of (38b) will contain only this one derivation, and nothing rules it out. The same is true for the first conjunct, so parallelism allows this construal of (38).

Fox shows the same pattern in several other cases, where long-distance QR cannot change the interpretation (with two universal quantifiers in the second conjunct, and with negation). In all these cases, the ambiguity of the first conjunct is lost in the ellipsis context.

As impressive as the arguments are for the interpretation-based MLC, this view also poses serious problems.

First, as observed in section 1.1, the MLC has broad coverage in the minimalist program. It is assumed, for example, to also cover all instances of relativized minimality. However, unlike superiority effects and QR, none of the other movement instances governed by the MLC show any interpretation dependence. *Wh*-islands provide a clear instance. These may be weaker than the cases of relativized minimality with A-movement, and they may vary in unacceptability. Thus, (42) is worse than (43) even in English. (In Hebrew, (42) is fine and (43) is out, for reasons discussed in Reinhart 1981b.) But whatever status they have, it is not affected by context or interpretative needs.

(42) *I wonder from whom you forgot what you got.

(43) ?I wonder what you forgot from whom you got.

It is not too difficult to imagine which questions would be denoted by each of these derivations, had they been allowed. It is also clear that in each case, the given derivation is the only way to express the relevant question, based on the given numeration. Still, this does not improve the

derivation. This means, then, that the status of (42) and (43), including the issue of why the first is worse, is determined by the computational system with no access to any interface considerations.

The question, then, is why just the two instances of the MLC we discussed should show interface sensitivity, and more generally, what determines when a syntactic condition is sensitive to interpretation. This is a serious problem, since if we cannot define precisely the set of operations subject to interface reference-set computation, we face the danger of a vacuous theory, where all movement depends on our undefined feelings about meaning. Possibly the problem of relativized minimality can be dismissed, if it turns out that relativized minimality is not an instance of the MLC, as might indeed be the case, independently of the problem under consideration. If so, then only superiority and QR are governed by the MLC, but it is still appropriate to wonder why just this condition is sensitive to interpretation.

Next, recall that the analysis rests crucially on the earlier view of the MLC as global reference-set computation. As we saw in section 1.1, the reference-set view of the MLC was rejected with good reason. The global nature of this computation poses a serious problem for optimal design, since it is inconsistent with what is known about human processing, hence it would require bypassing the computational system by heuristic algorithms. Fox (2000) offers a reanalysis of QR that can be viewed as local rather than global computation. But it remains the case (as we saw in section 1.1) that there was never any empirical reason to assume that this kind of computation is involved in the relevant problems, and it was just a mistaken formulation of the procedure of feature checking that led to the view that *wh*-movement obeys the MLC.

Regarding superiority, we should note that the argument cited above for why interpretation-based reference-set computation is needed is not as strong as it may seem; in fact, it was probably mistaken. A point I overlooked in Reinhart [1994] 1998 is that the same argument does not extend to other instances of superiority violations. We saw in section 1.1 that superiority effects are worse across a clause boundary, as in (5), repeated in (44).

(44) a. Whom did Lucie persuade e [PRO to visit whom]?
b. *Whom did Lucie persuade whom [PRO to visit e]?

(45) *Who remembers whom Lucie persuaded whom to visit e?

(24b) Who e knows [what [who bought e]]?

In cases like (44b), the violation remains the same if the derivation is embedded in a context like (45). This is precisely the same context as (24b), repeated above, which appeared to license such violations inside the clause. Example (45) has a reading that cannot be obtained with a derivation that does not violate superiority, but still, the derivation is not improved.

Furthermore, even within the same clause, the generalization illustrated in (24b) has been challenged. Chomsky (1995, 387, note 69) points out that (46) is unacceptable, and suggests that perhaps (24b) only reflects "preference for association of likes."

(46) *What determines to whom who will speak?

It appears that superiority violations inside the clause are weak to begin with, and that various factors may affect their acceptability. However, there is no systematic account in terms of truth-condition differences that can explain the full range of variations here.[8] A more promising direction in investigating the superiority phenomenon is that superiority inside the clause is affected by focus and stress considerations at the PF-interface. In any case, at least for the time being, it would not be wise to base any theory on a phenomenon so poorly understood. So there is no reason to conclude that reference-set computation is involved in such cases. For all we know, superiority remains a purely syntactic condition, and it can be captured by the revised MLC (based on "attract"), with no appeal to either reference sets or interpretation.

In the case of QR, Fox's findings are completely solid, and only strengthened by further inquiry into the facts. I will argue that these findings support the general claim that QR is subject to reference-set computation at the interface. The question is whether this computation is indeed governed by the MLC. Note that Fox's specific account of his findings rests on the prevailing assumption that QR is always obligatory for the interpretation of all quantified DPs, and the only question is how far an internal quantified argument can travel. This is the question that, according to Fox, is addressed by the MLC. But the underlying theoretical assumption has never, in fact, been motivated by empirical considerations. Rather, it is purely conceptual, or theory internal.

In the introduction, I outlined several ways that derivations (D) can be associated with interpretations (their possible uses U). The theoretical preference in linguistics has been to already code everything needed for the interpretation in the syntax. On this view, if the logical representation of VP-internal quantifiers requires some lambda abstraction, the variables needed for the λ-operator should be available in the syntactic (LF) repre-

sentation, which would be obtained by applying QR (see, e.g., Heim and Kratzer 1998). Though it is easy to see why this is convenient, it is not the only conceivable solution to this problem of association. Another possibility mentioned in the introduction is that in some instances, the set U is determined by independent properties and computations of the external systems, which apply to legible CS representations and further modify them. In this specific instance, the system of logic (inference) that accesses syntactic derivations may apply its own computations to interpret them, whether by inserting λ-predicates as in the Montague tradition, or by other means of type shifting available to logical syntax. On this view, what makes the representation legible to the inference system is the lexical semantic properties of the DPs (including their semantic definitions), but the rest of the semantic computation is carried out at that system, not at the CS.

Since the debate between these two possibilities is conceptual rather than empirical, we may as well choose the one that renders the CS itself more efficient. For instance, if the MLC is needed only for such unverified instances of covert movement, the alternative that this does not happen in the syntax should be seriously considered. The position I defend in chapter 2 is that there is no covert movement just for the interpretation of quantifiers. The only situation where further covert movement must be assumed is when the scope of a given quantified DP is not identical to its scope at the overt syntactic structure. If we adopt this view, there is no reason to assume that the MLC is involved here.

Nevertheless, scope-shifting of this type applies only when needed for interface purposes—that is, to obtain an interpretation that is otherwise unobtainable. In the literature surveyed above, these types of considerations were labeled "interface economy"—economy considerations that allow a certain operation to apply only if it is required by the interface.

An alternative way to capture interface economy was proposed by Chomsky (1995). Its essence is building the relevant interface considerations of QR into the numeration. Chomsky assumes (not just for this problem) that any item "enters the numeration only if it has an effect on output" (economy principle (76)). He appears to assume, further, that QR needs to apply only to capture scope-shift, as described above (which reflects a change in his earlier position). Suppose it is a movement of some feature like QUANT. Some functional feature must, then, be included in the numeration to host this feature, and it will eventually be merged in a topmost IP position. This functional feature will be allowed into the numeration only if it has an effect on the output—that is, if the interpretation obtained is not identical to what will be obtained without scope-shift.

Fox's insight is captured in this framework just the same: the relevant QUANT projection can be inserted into the numeration only in cases where Fox's analysis allowed long-distance QR to apply. On this view, then, interface needs determine the shape of the numeration; the underlying intuition may be that it is at the stage of choosing the building blocks for the derivation that speakers select items according to what they want to say. (Theoretically, this line of thought resembles the earlier position that all aspects of meaning are determined in deep structure.)

Under this view, it appears that no reference-set computation is involved in QR, which is an advantage in terms of optimal design.[9] This move still raises some conceptual questions, but the question I am more concerned with here is that of psychological reality. A crucial implication of this view is that once the relevant QUANT feature is selected into the numeration, the QR-operation is motivated by convergence, just like any other operation. Thus, scope-shift obtained by QR ends up indistinguishable in status from any other syntactic operation. In practice, however, it was found that scope-shift derivations are harder to process and less common in discourse than overt scope. (I elaborate on this point in chapter 2.) No such complexities are found in standard cases of syntactic movement. There would be no obvious way to explain this difference under the view that QR is indistinguishable from any other movement operation.

Furthermore, if there are, as I will argue, other instances of interface economy with the same properties, they will all have to be encoded into the computational system in the same way. Feature coding is, in fact, what guarantees that this solution is fully explicit and restrictive. There are, however, cases where syntactic encoding is more problematic than it is for QR (though of course this may always be possible, at a serious theoretical cost). So some alternative account is needed anyway.

The line I will pursue is that QR, and other instances of interface economy, indeed involve reference-set computation of the type examined above—that is, the reference set consists of pairs $\langle d, i \rangle$ of derivation and interpretation. A given $\langle d, i \rangle$ pair is blocked if the same interface effect could be obtained more economically—in other words, if there is a better $\langle d, i \rangle$ competitor in the reference set. However, this is not governed by the MLC, nor by feature encoding in the computational system. Reference-set computation, though available to the CS, is a "last-resort" procedure enforced at the interface in a restricted set of cases to be defined. It is not enforced by the needs of the syntactic derivation, but by some deficiency of the outputs of the system at the interface.

1.3 The Interface Strategy: Repair of Imperfections

I know of only four instances where there is substantial evidence for assuming that reference-set computation is at work at the interface: QR, already discussed; stress-shift for the purpose of focus construal; the co-reference strategy of Reinhart 1983a (binding conditions B, C); and the computation of scalar implicatures. I will discuss the first three instances in detail in chapters 2–4, and scalar implicatures more briefly in section 5.3. But first it may be appropriate to ask what they might have in common, or when reference-set computation must apply.

In Reinhart 1995, I suggested that reference-set computation is involved when an uneconomical procedure is needed in order to adjust a derivation for use at the interface. So this computation is triggered only by the application of such uneconomical procedures. The first question, then, is what sense of *economy* is involved here, specifically, what counts as a noneconomical way to satisfy an interface need (in other words, what is the metric, in terms of Optimality Theory).

In the case of QR, I believed at the time that an answer could be drawn from Chomsky's economy principle (1), repeated in (47), which we examined briefly in section 1.1.

(47) "If a derivation D converges without application of some operation, then that application is disallowed" (Chomsky 1992, 47).

Principle (47) poses a severe restriction on the computational system: in each given derivation, the system is allowed to apply an operation (from the available inventory of operations) only if applying this operation is needed for convergence, which was implemented as feature checking. If true, then the computational system is a most efficient or economical system, with no superfluous steps in its derivations. It is obvious that QR is not an operation needed for convergence in the strict sense of checking syntactic features. (As we saw, it is possible to create a feature for the occasion; however, this move is not motivated by syntactic needs of the derivation, but by interface needs.) Recall that we are assuming that QR applies only for scope-shift, and it is not otherwise needed for the interpretation of quantifiers.

Applying QR at a given derivation means, then, that we select a move operation from the available inventory, even though it is not needed for convergence—that is, we violate principle (47). It is at this stage of violating a basic economy principle of the computational system that a reference set should be consulted to verify that this indeed is the only way to

meet the interface needs. So it would be approved only if scope-shift has an effect on the interpretation, as Fox (1995) showed.

For this line of reasoning to also apply to the other instances of reference-set computation at the interface, it would be necessary to extend principle (47) so it covers any superfluous operation, not just syntactic movement. Thus, for a derivation to meet the PF-interface, it needs to have main stress. Assignment of this stress, then, is not superfluous. However, stress-shift is, so it has to be checked against a reference set. In the case of coreference and implicatures, what I assumed to be at stake is applying a superfluous interpretative procedure.

We should note, however, that the status of (47) is not, in fact, fully clear when overt syntactic operations are concerned. Originally, it represented a theoretical hope that there is no optional movement in the derivation of sentences, and all applications of movement can be reduced to feature checking (convergence). In practice, however, we do find across languages many instances of operations that seem to apply optionally in terms of syntactic convergence, like scrambling, topicalization, PP-preposing, and a variety of "stylistic-movement" options, which change word order. The assumption that the computational system abides by (47) has led to an industry of analyses attempting to show that each instance of such movement is either motivated by syntactic features and the corresponding functional projections to host them, or involves interface economy, namely, reference-set computation at the interface. The cost to the computational system, if all these analyses are correct, is much higher than if we assume that optional movement exists.

In some instances, work within the framework of interface economy has entailed ranking optional operations in terms of their "cost." For example, is it more costly to apply word-order shift by an optional syntactic operation, or to apply optional stress-shift? Thus, the theory of the interface is in danger of becoming an unconstrained version of Optimality Theory: if a system includes ranking of operations, which may vary across languages, its expressive power (the set of possible languages it generates) is greater than that of a system with no such ranking. Such a system is bound to allow many more options than are actually found in natural language. Even if we do not enter the realm of ranking, a point I would keep returning to is that reference-set computation is costly in terms of processing, even if it applies at the interface. If all, or many, of the instances of optional movement involve such computation, language cannot be very optimally designed. I have not yet discussed what counts as evidence that computation of this nature is indeed involved in a given

instance. Once this is defined, we will also be able to observe that there is no empirical evidence for computation like this in most instances of optional movement.

In section 2.7, I will argue that something like (47) may still be needed for covert movement (i.e., movement after the phonetic spell-out). A computational system allowing unrestricted covert movement is in danger of not being optimally usable at the interface, since each phonetic string may allow many possible interpretations, depending on which covert operations took place. Allowing this to apply just for purposes of convergence makes covert operations fully recoverable, thus restricting the set of interpretative options. But for overt movement, there is no such danger, since all operations are overtly visible. A reasonable alternative is to assume that optional overt movement simply exists—that is, the computational system allows it.

Once optional movement is available, it would make sense for speakers to use this option to improve and refine the context interface. For instance, the well-established tendency to place topic material in sentence-initial position may make use of fronting operations. If these are optional, there is no need to assume that reference-set computation is involved in the choice to apply fronting. As argued in Reinhart 1995, part IV, topic considerations indeed make no use of this kind of computation. This does not mean, though, that in practice all word-order options available are functional at that interface—there may be a certain amount of arbitrariness. But even if all the options are used at the interface, this does not guarantee that we have, as linguists, the theoretical tools to define all these uses at present. The context interface is the hardest to formulate, so we may have to live with a certain lack of clarity regarding this question for a while.

But we are still left with the question of which interface considerations do enforce reference-set computation. It cannot simply be the need to apply a superfluous operation, because as we just saw, there may be many innocent superfluous operations. The intuition that (47) enabled us to state is that applying the operations in question violates some principle that prohibits their normal application (even if (47) is not itself the relevant principle in all cases).

As I have mentioned, in the case of QR, the principle violated may still be (47), if it is formulated to apply to covert movement only. But the other instances I will examine do not involve covert movement. The question of what principle is violated may vary with the operations applied, and I will get back to this question in subsequent chapters, where

I examine specific instances of reference-set computation. For now, let us call the operations resulting in reference-set computation *illicit operations*, in the sense that their application violates some prohibiting principle. Since the reference-set type of strategy applies to rule out illicit operations when not required by interface needs, the type of operation itself is not determined by this strategy, and instances can be found in unrelated modules.

My basic assumption, then, is that the reference-set type of strategy at the interface is a kind of repair mechanism, activated when the outputs of the computational system fail to meet an interface need. In other words, it is invoked when there is an imperfection in the system. Recall from the introduction that the basic requirement of the computational system is that it should enable the interface. In an optimally designed system, the bare minimum needed for convergence should also be sufficient to satisfy the interface conditions. Instances where this fails to be the case may be viewed as imperfections in the system. (Note that I am talking here about operations needed for convergence and not about their other applications. An operation that in one context applies obligatorily for convergence may apply optionally in another, where not required for convergence.)

Let me illustrate this notion of imperfection with a preview of the focus problem, to be discussed in chapter 3. A basic requirement of the context interface is that sentences be associated with a focus (or foci). The question is how the computational system guarantees the identification and marking of the focus constituent. An independent requirement of the PF-interface is that each sentence carry some main stress, which is necessary for pronunciation. Let us now imagine a perfect computational system. In that system, the obligatory assignment of main stress to the derivation would also be sufficient for the association with focus, in that it would provide the marking of the focus constituent. In fact, such a view of the perfect focus assignment was proposed in Chomsky 1971.

Let us see how this approach works. Assuming that the main-stress rule applies independently, the simple rule in (48) selects a set of possible foci for each derivation.

(48) The focus of a given derivation is any constituent containing the main stress of IP.

(49) a. My neighbor is building a **desk**.
b. [$_{DP}$ a **desk**]
c. [$_{VP}$ building a **desk**]
d. [$_{IP}$ My neighbor is building a **desk**]

Suppose that in (49a) main stress falls on *a desk*. (Main stress is marked by means of boldface throughout.) All the constituents in (49b–d) contain this main stress. Hence, (48) determines that any of them can serve as a focus. We may refer to (49b–d) as the focus set of (49a).

At the context interface, one member of the focus set is selected as the actual focus of the sentence. Sentence (49a), repeated below, can be used as an answer in any of the contexts in (50), with the italicized F-bracketed constituent as focus.

(49) a. My neighbor is building a **desk**.

(50) a. Speaker A: What's your neighbor building?
Speaker B: My neighbor is building [$_F$ *a **desk***].
b. Speaker A: What's your neighbor doing these days?
Speaker B: My neighbor [$_F$ *is building a **desk***].
c. Speaker A: What's this noise?
Speaker B: [$_F$ *My neighbor is building a **desk***].

At this stage, it is up to the discourse conditions, rather than the computational system, to determine the relevant focus to be selected in a given context.

If the foci defined by (48) were sufficient for the use of sentence (49a) in all possible contexts, we could conclude that we have a perfect system. So far we have only applied the stress operation needed anyway for phonetic convergence, and a general interface rule (48) links all derivations to appropriate contexts.

The actual human computational system, however, is not that perfect. We can easily find contexts where we would want to use derivation (49a), but none of the foci associated with it fit the given context. For example, (49a), with the same main stress indicated with boldface, cannot be used as an answer in either of the contexts of (51). This is so, because the context requires the F-bracketed constituents in (51) to be the foci, but the focus set defined for this derivation by (48) does not include these constituents. (The # symbol indicates, throughout, inappropriateness to context.)

(51) a. Speaker A: Has your neighbor bought a desk already?
Speaker B: #No, my neighbor is [$_F$ *building*] a **desk**.
b. Speaker A: Who is building a desk?
Speaker B: #[$_F$ *My neighbor*] is building a **desk**.

This means, then, that our computational system contains an imperfection. The stress operation needed for PF-convergence is not sufficient to

meet all the needs of the context interface. Thus the question is what to do when facing an imperfection in the system.

Note that the problem of QR is essentially of the same type. In a perfect system, the overt structure associated with a derivation would be sufficient to capture all its scope construals in different contexts. In practice, this is not the case, and the context may require a construal not generated by the computational system (without an illicit covert operation). Indeed, there is also a certain resemblance in the history of how quantifier scope and focus have been conceptualized in theoretical linguistics.

As we just observed, at the earlier stages—Chomsky 1971—focus was essentially viewed as a property defined in terms of PF-structures. This approach rested on the notion of "normal" or "neutral" intonation, namely, in present terminology, the assumption that there is an independent stress operation needed for PF-convergence. For the cases of imperfections, where this stress operation is not sufficient for the interface, Chomsky (1971, 199) argued that "special ... processes of a poorly understood sort may apply in the generation of sentences, marking certain items as bearing specific expressive or contrastive features that will shift the intonation center." A distinction is implicit here between neutral stress and marked stress, obtained by applying special required operations.

In Keenan and Faltz 1978 and Reinhart 1983a, the same was assumed for the scope of quantifiers: scope is determined by the syntactic configuration of the overt structure. A rule like QR is used only when it is necessary to derive scope construal wider than the overt c-command domain, and it is viewed there as a marked, discourse-driven, operation. On this view, overt scope is always the preferred option, with one systematic exception in the case of internal NP-scope, noted in Reinhart 1976, who argued that these cases require an independent analysis. (I return to these questions in section 2.7.)

However, the concept of markedness was problematic. It appears easy to find examples of covertly determined wide scope that sound perfectly natural. (For instance, as Hirschbühler (1982) noted, in a sentence like *An American flag was hanging in front of every building*, the most natural construal is with wide scope for *every building*.) If it can at times be as easy to get the marked derivation as the unmarked one, it is not clear what empirical content the concept of markedness could have.

Similarly, the distinction between marked and neutral stress has also been challenged. As an argument against the Nuclear Stress Rule (NSR) or Chomsky's (1971) focus analysis, it was repeatedly pointed out that in

the appropriate context, main stress can fall anywhere, with effects hardly distinguishable from that of the neutral stress. (For an overview, see Selkirk 1984.) The crucial problem here as well is whether any content can be given to the concept of markedness. If there is no obvious way to distinguish neutral and marked stress, we run into the danger of vacuity—having a theory that excludes nothing regarding stress. The facts that follow from its rules are labeled "neutral," and everything else, "marked." (This type of theory is always true, regardless of what its rules are, by virtue of being unfalsifiable.)

A more realistic conclusion appeared to be that there is no sentence-level generalization governing the selection of possible foci, and any expression can be a focus, subject only to discourse appropriateness. Hence, it was concluded that main stress cannot be assigned at PF independently of the semantics of the sentence, and it must be the other way around: sentence intonation reflects its independently determined focus structure.

The prevailing solution since Chomsky 1976, where LF-movement was introduced, has been that both scope and focus are identified at the covert structure: LF. A focus constituent has been marked by a focus feature, and the marked constituent moved at LF. Thus, covert "focus movement" has been assumed to be obligatory for every derivation. QR has been assumed to be obligatory in all derivations with quantified constituents. Thus, the problem of markedness has been avoided.

But this solution is problematic as well. First, while focus movement does eliminate the problem of markedness, the relations between stress and structure become a complex issue, raising questions about the visibility of the covert structure to PF-rules (stress). More generally, this solution placed much of the burden of capturing the interface requirements on the covert structures. I have already noted a problem with this approach and will return to it later. Generally, the more information that is captured covertly, the more mysterious it is that speakers are able to understand each other.

Admitting an imperfection in the system, we may still wonder whether it must be as sweeping as entailed by this analysis—for example, that the derivation's main stress is uniformly determined at the covert structure. Furthermore, we may note that this massive imperfection still does not take us very far toward capturing the actual interface conditions. Though no satisfactory content could be given to the notion of markedness, in practice it is not the case that covert quantifier scope is always as free and easy to get as overt scope, and certainly not that the so-called marked stress is completely free. Introducing the machinery of covert movement

is thus just the first step in formulating the question of when it can actually be used. Answering this question will require introducing more conditions and rules (more imperfections). One may wonder whether it is not possible to start directly by answering the second question, skipping the massive imperfection we introduced just to formulate it.

In an influential work, Cinque (1993) offered a new perspective on the NSR and argued that the earlier view of the relations between stress and focus can be maintained. This direction is pursued here in chapter 3. We should note, however, that the analysis is based on a revival of the distinction between neutral and marked stress: when the stress assigned by the NSR is not appropriate to the context, a special stress-shift operation applies, yielding marked stress. So the question "How do we know it is marked?" is relevant again.

In Reinhart 1995, 1998, I argued that it is a mistake to search for evidence of markedness in the realm of direct intuitions. A marked derivation is a derivation that involves an illicit operation, as defined above. (Both QR and stress-shift are viewed here as illicit operations.) When this is done with no reason, the result is visibly awkward. But if using the illicit operation is unquestionably the only way to satisfy a certain interface need, the result sounds perfectly fine, and it is only indirectly that we can see that it is marked nevertheless. As we observed in the case of QR, Fox (1995, 2000) provides ellipsis evidence consistent with the claim that QR does not take place when not needed for interpretation. The evidence for the illicit status of the stress-shift operation will be discussed in chapter 3.

In more precise terms, what is claimed in the last paragraph is that computing QR and stress-shift involves constructing a reference set and checking whether it contains a better $\langle d, i \rangle$ pair—that is, a pair derived without applying the illicit operation. If it does, the derivation is blocked (in other words, if we nevertheless produce it, it is visibly marked).

Thus, to conclude the question we started with in this section, reference-set strategies are "last-resort" strategies used to repair or make up for imperfections in the computational system. They are used when the need arises to apply an illicit operation in order to adjust a derivation to the interface needs.

That illicit operations need to apply at all remains an imperfection in the core system. However, this imperfection is much less serious than we previously assumed. First, PF-procedures, like stress, operate, as they should, on the overt structure. Next, the illicit QR and stress-shift cannot apply just anywhere but are restricted by reference-set checking. On the

other hand, as we saw, global reference-set computation has a serious processing cost, which is problematic for the secondary requirement of meeting the empirical conditions of use—processing and acquisition. Here, too, the problem is far less massive than that demonstrated by the Minimal Link Condition, since reference-set computation is triggered only if an illicit operation applies. Nevertheless, in these restricted cases, we do have a deviation from optimal design.

The strongest interpretation of the concept of imperfection is that if we have to admit it into our theory, there should also be some way to observe the imperfection in the use of language itself, say in the processing of sentences. I will argue that this is indeed the case when reference-set computation needs to apply to repair an imperfection—it comes with an observable processing cost.

Reference-set computation imposes a greater load on working memory than local computation does. Adults can apparently cope with this load (with limitations on the size of the reference set, as I will argue in section 2.7), but there is reason to believe that this load is too big for children, whose working memory is not yet as developed. Grodzinsky and Reinhart (1993) argue that the (relatively rare) chance pattern found in the acquisition of coreference (Condition B, or their Rule I) indicates guess performance. The reason is that the relevant coreference strategy involves reference-set computation, and children are unable to execute the computation, which, as they know innately, is required for this task. In chapter 5, I will provide further evidence for this claim in the area of coreference, and argue that there is growing evidence that the same pattern is found in the other instances of reference-set computation.

If true, then acquisition findings also provide the most direct confirmation that reference-set computation is indeed involved in the relevant cases. This enables us to form a strong and strictly falsifiable hypothesis that if it is independently established that a certain interface problem requires global reference-set computation, we should also find out that children are unable to process and solve this problem. This puts a severe restriction on our theoretical freedom to postulate reference-set computation anywhere, as in Optimality Theory.

In conclusion, we should keep in mind that the reference-set strategy governing the application of illicit operations is just one of the interface strategies, and as I have pointed out, it is only if evidence for the computational complexity is found for some linguistic instance that we can conclude that it might fall under this type. Operations for which no such evidence can be shown must belong elsewhere. One option, noted

in section 1.2, is that they are directly encoded in the computational system, say, as optional features whose selection is governed by the interface requirement on the numeration, as suggested in Chomsky 1995. Or they may be governed by different context-adjustment strategies that apply at the interface and that do not involve reference-set computation. In Reinhart 2004, I discuss strategies of assessment and retrieval from discourse storage that govern the identification of topics and certain types of discourse presuppositions. As in the case of focus, I do not see a need to assume that the topic constituent is marked with a feature at the CS. However, the strategy governing its identification involves no comparison of derivations. Similarly, assessing the accessibility hierarchy that governs discourse anaphora resolution in Ariel's (1990) analysis involves no such comparisons.