Incremental Constraint-based Parsing: An Efficient Approach for Head-final Languages

Zelal Güngördü

PhD
University of Edinburgh
June, 1997

Declaration

I declare that this thesis has been composed by myself and that the research reported here has been conducted by myself unless otherwise indicated.

Edinburgh, 27 June 1997

Zelal Güngördü

Acknowledgements

I would like to thank my supervisors Matt Crocker and Elisabet Engdahl for their support and advice throughout my PhD years. I also greatly appreciate the constructive criticism, comments, and suggestions provided by my thesis examiners Claire Grover and John Nerbonne. Thanks also to Ewan Klein and Kemal Oflazer for very helpful comments on several chapters of this dissertation. And I thank everybody at the Centre for a very stimulating and friendly environment for interdisciplinary research.

I also thank the following people for providing me with native speaker judgements: Yasemin Aydemir, Çiğdem Balım, Cem Bozşahin, Tevfik Bultan, Aydın Durgunoğlu, Elvan Göçmen, Yılmaz Kılıçaslan, Dilsun Kırlı, Tahsin Kurç, İlker Kuruöz, Kemal Oflazer, Ülkü Şencan, Reha Tütüncü, and Dilek Yılmaz.

I am grateful to the following institutions for providing financial support during my PhD years: the Scientific and Technical Research Council of Turkey, the Committee of Vice-Chancellors and Principals of the Universities of the United Kingdom, and the Faculty of Science and Engineering of the University of Edinburgh.

Thanks also to my friends Scott McDonald, Padraic Monaghan, Merce Prat, David Sterratt, Simone Teufel, Jesse Tseng, Renata Vieira, and Gert Westermann for their love and support, especially in the last two years. I will never forget the dinner parties, brunches, ceilidhs, etc. we have had/been together.

Finally, I am grateful to my family for all the love, care, and support they have provided me with throughout my life.

Abstract

In this dissertation, I provide a left-to-right incremental parsing approach for Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag (1987, 1994)). HPSG is a lexicalized, constraint-based theory of grammar, which has also been widely exploited in computational linguistics in recent years. Head-final languages are known to pose problems for the incrementality of head-driven parsing models, proposed for parsing with constraint-based grammar formalisms, in both psycholinguistics and computational linguistics. Therefore, here I further focus my attention on processing a head-final language, specifically Turkish, to highlight any challenges that may arise in the case of such a language. The dissertation makes two principal contributions, the first part mainly providing the theoretical treatment required for the computational approach presented in the second part.

The first part of the dissertation is concerned with the analysis of certain phenomena in Turkish grammar within the framework of HPSG. The phenomena explored in this part include word order variation and relativization in Turkish. Turkish is a head-final language that exhibits a considerable degree of word order freedom, with both local and long-distance scrambling. I focus on the syntactic aspects of this freedom in simple and complex Turkish sentences, detailing the assumptions I make both to deal with the variation in the word order, and also to capture certain restrictions on that variation, within the HPSG framework. The second phenomenon, relativization in Turkish, has drawn considerable attention in the literature, all accounts so far being within the tradition of transformational grammar. Here I propose a purely lexical account of the phenomenon within the framework of HPSG, which I claim is empirically more adequate than previous accounts, as well as being computationally more attractive.

The motivation behind the work presented in the second part of the dissertation mainly stems from psycholinguistic considerations. Experimental evidence (e.g. Marslen-Wilson (1973)) has shown that human language processing is highly incremental, meaning that humans construct a word-by-word partial representation of an utterance as they hear each word. Here I explore the computational effectiveness of an incremental processing mechanism for HPSG grammars. I argue that any such processing mechanism has to employ some sort of nonmonotonicity in order to guarantee both completeness and termination, and propose a way of doing that without violating the soundness of the overall approach. I present a parsing approach for HPSG grammars that parses a string of words from left to right, attaching every word of the input to a global structure as soon as it is encountered, thereby dynamically changing the structure as the parse progresses.

I further focus on certain issues that arise in incremental processing of a "free" word order, head-final language like Turkish. First, I investigate how the parser can benefit from the case values in Turkish in foreseeing the existence of an embedded phrase/clause before encountering its head, thereby improving the incrementality of structuring. Second, I propose a strategy for the incremental recovery of filler-gap relations in certain kinds of unbounded dependency constructions in Turkish, which further enables one to capture a number of (strong) preferences that humans exhibit in processing certain examples with potentially ambiguous long-distance dependency relations.

Contents

Chapter 1

Introduction

1.1 Research Goals

In this dissertation, I provide a left-to-right incremental parsing approach for Head-driven Phrase Structure Grammar (HPSG; Pollard and Sag (1987, 1994)). It has been shown by psycholinguistic evidence (e.g. Marslen-Wilson (1973)) that human language processing is highly incremental, suggesting that humans construct a word-by-word partial representation of an utterance as they hear each word. Therefore, in order to make any claims concerning the psycholinguistic reality of any existing linguistic theory, one has to prove that grammars embodied within that theory are ammenable to efficient incremental parsing.

While certain head-driven parsing models proposed in psycholinguistics (e.g. Abney (1987), Abney (1989), and Pritchett (1992)), and in computational linguistics (e.g. Kay (1989)) may appear incremental in the case of head-initial languages, head-final languages are known to pose problems for the incrementality of such models of parsing (Crocker (1996a)). Those models all assume that the structure of a phrase can only be built after its head has been processed, which may well mean a delay until the very last word of the phrase in the case of a head-final construction. Thus, here I further focus my attention on parsing Turkish, a head-final language, in order to highlight any challenges that may arise in the case of such a language, and improve the generality of the result.

Chapters 3 and 4 of the dissertation are concerned with the HPSG analyses of several distinguishing phenomena in Turkish grammar, namely, word order variation and relativization, respectively. In Chapter 3, I examine the syntactic aspects of "free" word order in Turkish, detailing the assumptions I make both to deal with the variation in the word order, and also to capture certain restrictions on that variation, within the HPSG framework. In Chapter 4, I propose an HPSG analysis of Turkish relative clauses, which is based upon an assumption that those clauses have lexically specified MOD values (encoded in the lexical entry of the verbal head of the clause).

I argue in this dissertation that any incremental parsing approach for HPSG grammars has to employ some form of nonmonotonicity in order to guarantee both completeness and termination, due to the possibility of left recursive structures in a language (such as English N′s with post-modifier PPs). The incremental parsing approach I propose here exploits underspecification in structure and nonmonotonicity in processing, to overcome the processing problems due to left recursion. In this approach, parsing an input string always starts with an underspecified global structure, and proceeds by attaching every word in the input string to that structure, thereby constraining the structure further and further with the processing of each word. Despite a high degree of nondeterminism in processing in the *worst* case, I propose certain strategies that would enable a serial parser for a head-final language to provide the correct analysis/analyses with the least possible number of processing steps in most cases (and with minimal reanalysis in the remaining cases).

I further focus on certain issues that arise in incremental processing of a "free" word order, head-final language like Turkish. First, I investigate how the parser can foresee the existence of an embedded phrase/clause before encountering its head, simply by exploiting certain restrictions on the co-occurrence of phrases with particular CASE values as sister constituents of the same clause in Turkish. Second, I propose a strategy for the incremental recovery of filler-gap relations in certain kinds of unbounded dependency constructions in Turkish, which further enables one to capture a number of (strong) preferences that humans exhibit in processing certain examples with potentially ambiguous long-distance dependency relations.

All theoretical proposals made in Chapters 3 and 4 (unless otherwise stated) have been fully implemented in a parsing system for Turkish based on the incremental parsing approach proposed in this dissertation. The implementation has been crucial to testing and further improving both the theoretical analyses proposed in Chapters 3 and 4, and also the parsing algorithm introduced in Chapter 6.

I further claim that as long as the underlying constraint satisfaction operation is unification, the only way of employing nonmonotonicity in a parsing approach for HPSG

– without violating the soundness of the overall approach – is to exploit underspecification in structure and a certain degree of non-determinism in processing, neither of which is easy to justify on psycholinguistic grounds.

1.2 Organization of the Thesis

In the rest of this chapter, I first present an overview of certain aspects of Turkish morphology in Section 1.3.1, and then briefly discuss definiteness and specificity of Turkish NPs in Section 1.3.2. Although the discussion in those two sections may help the reader (not familiar with Turkish) follow the rest of the dissertation, it is actually not essential to the discussion in the chapters to come.

In Chapter 2, I give an overview of HPSG, summarizing only some basic components of the theory to provide the reader with enough background for the rest of the dissertation. In particular, in Section 2.3 of that chapter, I present a view of several concepts in the theory, based on Sag (to appear), which are assumed in the chapters to follow.

In Chapters 3 and 4, I present theoretical accounts of several phenomena in Turkish grammar within the framework of HPSG. Chapter 3 is mainly concerned with the issue of "free" word order in Turkish. In Section 3.4 of that chapter, I provide a detailed characterization of the word order variation in simple and complex Turkish sentences, presenting the assumptions I make both to deal with the variation in the word order, and also to capture certain restrictions on that variation, within the HPSG framework. In Chapter 4, I present an HPSG account of relativization in Turkish.

Chapters 5 and 6 present the incremental parsing algorithm proposed in this dissertation for HPSG grammars. Chapter 5 provides an informal introduction to the approach, pointing out certain problems and outlining the solutions adopted. I also motivate in that chapter the use of case information in Turkish for improving the incrementality of structuring (Section 5.6), and the incremental recovery of long-distance dependencies in certain kinds of unbounded dependency constructions in Turkish (Section 5.7). Then in Chapter 6, I present the details of the parsing algorithm, providing answers for certain questions raised during the informal presentation in Chapter 5.

In Chapter 7, I discuss a number of further issues concerning the present parsing approach. First, in Section 7.1 I examine the degree of non-determinism embodied in the approach, and also make a number of suggestions to improve the efficiency of a parser implemented using this approach. Then in Section 7.2, I discuss the question of psychological plausibility of the approach. And in Section 7.3, I discuss certain implementational aspects of a parsing system implemented for Turkish in the LIFE programming language, which is based on the parsing algorithm provided in Chapter 6, and also incorporates the theoretical proposals made in Chapters 3 and 4.

Finally, Chapter 8 summarizes the results of the dissertation.

1.3 Background

1.3.1 Turkish Morphology

Turkish is an agglutinative language where word structures are formed by productive affixations of derivational and inflectional suffixes to root words (Oflazer (1994)). In this section, I present an overview of certain aspects of Turkish morphology (and certain consequences for the syntax) to help the reader follow the rest of the dissertation.

In Turkish, noun stems can be marked for plurality, possessiveness, case, etc. The plural suffix is *'-lEr'*.[1] Turkish has the following seven morphological cases: nominative (morphologically unmarked), genitive (marked with the suffix *'-(n)In'*), accusative (marked with *'-(y)I'*), dative (marked with *'-(y)E'*), ablative (marked with *'-(n)dEn'*), locative (marked with *'-(n)dE'*), and comitative/instrumental (marked with *'-(y)lE'*).

The possessor-possessed noun agreement is marked by a possessive suffix on the possessed noun which agrees with the genitive case-marked possessor in number and person (cf. (1.1)). Table 1.1 presents the respective possessive suffix for each person-number pair.

(1.1) a. ben-im kitab-ım 'my book'
I-GEN book-1sPoss

b. çocuğ-un kitab-ı 'the child's book'
child-GEN book-3sPoss

In addition, certain postpositions in Turkish form PPs that are structurally similar to possessive NPs. The object of the postposition is either genitive marked or unmarked

[1] Note that Turkish exhibits vowel harmony. Any suffix with an *'E'* or an *'I'* mentioned here in fact stands for all the allomorphs (vowel-harmony variants) of the suffix in question. As an example, *'-lEr'* can be realized as either *'-ler'* or *'-lar'* in a particular word.

Per \ Num	Sing	Plur
1st	*-(I)m*	*-(I)mIz*
2nd	*-(I)n*	*-(I)nIz*
3rd	*-(s)I*	*-lErI*

Table 1.1: The Possessive Suffixes

(i.e. nominative),[2] and the postposition *contains* a possessive suffix that agrees (in person and number) with its object and a case suffix (that is fixed for each postposition), as seen in (1.2).

(1.2) a. Biz [$_{PP}$ Berfu hakk-ı-nda] konuş-tu-k.
we Berfu about(-3sPoss-LOC) talk-PAST-1PL
'We have talked about Berfu.'

b. Şiir [$_{PP}$ ben-im taraf-ım-dan] yaz-ıl-dı.
poem I-GEN by(-1sPoss-ABL) write-PASS-PAST
'The poem was written by me.'

Verbs in Turkish can be morphologically marked for voice, mood, tense, number, person, question, etc. The subject-verb agreement is marked by an agreement suffix on the verb which agrees with the subject in number and person, see, for example, (1.3a). The third person singular agreement is morphologically unmarked, and the third person plural agreement can also be optionally unmarked (cf. (1.3b) and (1.3c), respectively). The subject agreement suffix on the verb differs with the mood and tense. Table 1.2, for example, presents the respective agreement suffix for each person-number pair in the progressive tense.

(1.3) a. Ben uyu-yor-um. 'I am sleeping.'
I sleep-PROG-1SG

b. Berfu uyu-yor. 'Berfu is sleeping.'
Berfu sleep-PROG

c. Çocuk-lar uyu-yor(-lar). 'The children are sleeping.'
child-PLU sleep-PROG(-3PL)

Per \ Num	Sing	Plur
1st	*-um*	*-uz*
2nd	*-sun*	*-sunuz*
3rd	∅	(*-lar*)

Table 1.2: The Subject-verb Agreement Suffixes in the Progressive Tense

In addition, non-finite verbs in Turkish such as nominalizations, infinitives, participles, and gerunds are also morphologically marked with their respective suffixes.

Nominalizations are classified into two types according to the suffixes they are marked with: i) 'act' type nominalizations are marked with *'-mE'* (cf. (1.4a)); and ii) 'fact' type nominalizations are marked with either *'-EcEk'* (future) or *'-dIk'* (non-future) (cf. (1.4b)). Nominalizations take genitive case-marked subjects, and can themselves be case-marked (just like nouns). In addition, the subject-verb agreement is marked by a suffix on the nominalization (preceding any case suffix on the same verb) which agrees with the subject in person and number, following the same morphological pattern as the possessor-possessed noun agreement in Table 1.1.

(1.4) a. Ben [$_{S_{[nomin]}}$ Berfu'-nun ev-e git-me-si-ni] isti-yor-um.
I Berfu-GEN home-DAT go-ACT-3sPoss-ACC want-PROG-1SG
'I want Berfu to go home.'

b. Ben [$_{S_{[nomin]}}$ Berfu'-nun ev-e git-tiğ-i-ni] san-dı-m.
I Berfu-GEN home-DAT go-FACT-3sPoss-ACC think-PAST-1SG
'I thought that Berfu had gone home.'

Lees (1965) observes that certain verbs in Turkish select 'act' type complement clauses (e.g. *iste- 'want'*, *bekle- 'expect'* and *çalış- 'try'*), while certain other verbs select 'fact' type complement clauses (e.g. *bil- 'know'*, *san- 'think'* and *um- 'hope'*). Compare, for example, (1.4a,b) with (1.5a,b), respectively.

[2]Although genitive case marking is in general associated with 'specificity' in Turkish (cf. Nilsson (1985)), the existence of case marking in this case doesn't seem to be related to the specificity of the NP. For example, personal pronouns are preferably marked genitive, while some other undoubtedly specific NPs such as proper nouns are usually left unmarked.

(1.5) a. * Ben [Berfu-nun ev-e git-tiğ-i-ni] iste-di-m.
I Berfu-GEN home-DAT go-FACT-3sPoss-ACC want-PAST-1SG

b. * Ben [Berfu-nun ev-e git-me-si-ni] san-dı-m.
I Berfu-GEN home-DAT go-ACT-3sPoss-ACC think-PAST-1SG

Moreover, Sezer (1991) notes that in cases where both types of clauses can occur as complements of the same predicate, sentences with different complement types have different interpretations:

(1.6) a. Ben [Berfu-nun yürü-düğ-ü-nü] bil-iyor-um.
I Berfu-GEN walk-FACT-3sPoss-ACC know-PROG-1SG
'I know that Berfu walks.'

b. Ben [Berfu-nun yürü-me-si-ni] bil-iyor-um.
I Berfu-GEN walk-ACT-3sPoss-ACC know-PROG-1SG
'I know how Berfu walks.

Infinitives in Turkish are marked with the suffix *'-mEk'*. Infinitive phrases may act as subjects, as in (1.7a), or as complements of 'subject equi' verbs such as *iste- 'want'*, as seen in (1.7b). (The unexpressed subject of the VP[*inf*] complement in (1.7b) is assumed to be controlled by the matrix subject *Berfu*.)

(1.7) a. [$_{VP[inf]}$ Ev-e git-mek] Berfu-yu mutlu et-ti.
home-DAT go-INF Berfu-ACC happy make-PAST
'To go home has made Berfu happy.'

b. Berfu [$_{VP[inf]}$ ev-e git-mek] isti-yor.
Berfu home-DAT go-INF want-PROG
'Berfu wants to go home.'

Participles, which are morphologically marked with one of the suffixes *'-(y)En'*, *'-dIk'* and *'-(y)EcEk'*, act as verbal heads of relative clauses in Turkish. (1.8a), for example, contains a participle clause where the subject is relativized and occurs as the head noun of the relative clause, whereas (1.8b) contains one where the object is relativized. Notice that in the latter case, the subject of the clause is genitive marked and the participle takes a possessive suffix which agrees with the subject, again according to the morphological pattern in Table 1.1.

(1.8) a. [$_{S[part]}$ $-_i$ kitab-ı oku-yan] çocuk$_i$
book-ACC read-PART child
'the child who is reading the book'

b. [$_{S[part]}$ çocuğ-un $-_i$ oku-duğ-u] kitap$_i$
child-GEN read-PART-3sPoss book
'the book that the child is reading'

Note that in the case of a 'headless relative clause', as in (1.9), the plural, case, etc. suffixes that would normally be affixed to the head noun of the relative clause are instead affixed to the verbal head of the S[*part*].

(1.9) [$_{S[part]}$ $-$ kitab-ı oku-yan-lar-a]
book-ACC read-PART-PLU-DAT
'to those who are reading the book'

Finally, gerunds in Turkish may be morphologically marked with one of a number of suffixes (e.g. *'-dIkcE'*, *'-(y)IncE'*, *'-madan'*, etc.), and head clauses that act as sentential modifiers:

(1.10) a. Berfu [$_{S[ger]}$ Mehmet uyu-yunca] ev-e git-ti.
Berfu Mehmet sleep-GER home-DAT go-PAST
'Berfu went home when (after) Mehmet fell asleep.'

b. Berfu [$_{S[ger]}$ Mehmet uyu-madan] ev-e git-ti.
Berfu Mehmet sleep-GER home-DAT go-PAST
'Berfu went home before Mehmet fell asleep.'

Turkish is a pro-drop language, meaning that subjects (and possessors) can be readily dropped (particularly in the case of first and second persons) mainly due to the morphological realization of the subject-verb (possessor-possessed noun) agreement on the verb (possessed noun). In addition, other complements can also be dropped so long as they are recoverable in a given discourse. Consider, for example, the following from Hoffman (1995):

(1.11) Fatma $\emptyset$ kitab-ı-nı ara-dı. $\emptyset$ $\emptyset$ Bul-ama-dı.
Fatma book-3sPoss-ACC search-PAST find-NEG-PAST
'Fatma searched for (her) book. (She) could not find (it).'

Kornfilt (1984) notes that the postpositions that bear agreement morphology with their objects (see page 5) also let their objects be dropped (just like subjects and possessors), as seen in (1.12a), unlike certain other postpositions, such as *için 'for'* and *gibi 'like'*, without any agreement morphology, cf. (1.12b).

(1.12) a. Şiir [$_{PP}$ ben-im/∅ taraf-ım-dan] yaz-ıl-dı.
poem I-GEN by(-1sPoss-ABL) write-PASS-PAST
'The poem was written by me.'

b. Ben şiir-i [$_{PP}$ sen-in/*∅ için] yaz-dı-m.
I poem-ACC you-GEN for write-PAST-1SG
'I have written the poem for you.

1.3.2 Definiteness and Specificity in Turkish

In Turkish, definiteness, specificity and referentiality of subject and object NPs are determined via factors such as word order, stress, the use of an indefinite determiner and/or case-marking, tense and modality (Dede (1986)). Turkish doesn't have a definite article.[3] Bare subject nouns can be either definite or nonreferential depending on the word order and stress (Hoffman (1995)). Sentence-initial subjects are usually interpreted as definite, referring to specific discourse entities (cf. (1.13a)). In the (non-sentence-initial) immediately preverbal position, however, subjects are preferably interpreted as nonspecific or nonreferential (cf. (1.13b)).

(1.13) a. Köpek dışarda havlı-yor. 'The dog is barking outside.'
dog outside bark-PROG

b. Dışarda köpek havlı-yor. 'Some dog/dogs is/are barking outside.'
outside dog bark-PROG

On the other hand, in cases where the sentence-initial position actually coincides with the immediately preverbal position, as in (1.14), the referential status of the subject depends mainly on stress (Dede (1986)). The subject in (1.14a), which bears the neutral stress, is ambiguous between definite and nonreferential readings. However, the shift of the stress to the verb in (1.14b) gives the subject a definite reading.

[3]It does however have the 'deictic' determiners *bu 'this'*, *şu 'that'*, and *o 'that'*, which undoubtedly render the NPs they occur in, definite.

(1.14) a. *Köpek* havlı-yor. 'The dog is barking.'
dog bark-PROG 'Some dog/dogs is/are barking.'

b. Köpek *havlı-yor.* 'The dog is barking.'
dog bark-PROG

In the case of direct objects, it is the existence of an indefinite determiner and/or accusative case-marking that determines the definiteness, specificity and referentiality of the NP. Accusative marking on direct objects is optional in Turkish. A bare object noun can be interpreted as definite or nonreferential, depending on whether it is accusative marked or unmarked (i.e. nominative) (cf. (1.15a,b), respectively). On the other hand, an object NP with either of the indefinite determiners *bir 'a'* and *birkaç 'some/a few'* can be either specific or nonspecific, again depending on the existence of accusative marking (cf. (1.15c,d), respectively) (Dede (1986), and Enç (1991)).

(1.15) a. Güneş kitab-ı oku-yor. 'Güneş is reading the book.'
Güneş book-ACC read-PROG

b. Güneş kitap oku-yor. 'Güneş is reading a book/some books.'
Güneş book read-PROG

c. Güneş bir kitab-ı oku-yor. 'Güneş is reading a (specific) book.'
Güneş a book-ACC read-PROG

d. Güneş bir kitap oku-yor. 'Güneş is reading a book.'
Güneş a book read-PROG

Dede (1986) notes that an additional factor that affects the status of NPs in Turkish is tense and modality. Consider, for example, the following (from Dede):

(1.16) a. Çocuk-lar çikolata-yı sev-er.
child-PLU chocolate-ACC love-AOR
'Children love chocolate.'

b. Çocuk-lar çikolata-yı sev-di.
child-PLU chocolate-ACC love-PAST
'The children loved the chocolate.'

In (1.16a), both the sentence-initial subject *'çocuklar'* and the accusative marked object *'çikolatayı'* have a generic reading, due to the generic context given to the sentence by the aorist tense verb *'sever'*.[4] In (1.16b), on the other hand, both the subject and the object are interpreted as definite.

[4]The aorist tense in Turkish is typically used in 'habitual' and 'generic' contexts.

Chapter 2

An Overview of HPSG

This chapter aims to give an overview of Head-driven Phrase Structure Grammar (HPSG), the linguistic theory that underlies both parts of this dissertation, summarizing only some basic components of the theory to provide the reader with enough background for the rest of the dissertation. One should refer to Pollard and Sag (1987, 1994) for a thorough discussion of the linguistic motivations behind the theory and detailed accounts of various linguistic phenomena.

HPSG is within a family of grammar formalisms known as 'unification-based' (see Shieber (1986)), members of which come from theoretical linguistic theories (e.g. Lexical Functional Grammar (Kaplan and Bresnan (1982)), and certain variants of Categorial Grammar proposed by, e.g., Uszkoreit (1986), and Zeevat *et al.* (1987)), as well as from computational linguistic formalisms (e.g. Functional Unification Grammar (Kay (1985)), and PATR-II (Shieber (1984); and Shieber *et al.* (1983)) (cf. Pollard and Sag (1987)[Chapter 1]). These theories and formalisms all use 'feature structures' to represent linguistic objects, which are essentially sets of constraints that characterize (partial) phonological, syntactic, semantic and contextual information concerning a given linguistic object. The fundamental operation on feature structures is the 'unification' operation, which operates in a monotonic manner, combining the information from a set of *compatible* feature structures in a feature structure that contains all and only the information present in that set of feature structures.

Unification-based theories of grammar are strictly 'declarative', meaning that they only embody grammars whose constructs only specify what constraints are brought to bear during language processing, but no particular order in which those constraints are to be satisfied. This is in contrast with the 'derivational' theories of grammar, such as

Government and Binding Theory (Chomsky (1981)), in which totally specified linguistic structures are successively *transformed* into different (again totally specified) linguistic structures according to a fixed ordering (Pollard and Sag (1987)[Chapter 1]).

The chapter starts below with a description of the HPSG formalism introduced by Pollard and Sag (1994), and then Section 2.2 outlines some of the basic concepts of the theory. Readers familiar with HPSG should skip to Section 2.3, which presents an alternative view of the concepts mentioned in Section 2.2, assumed in the chapters to follow. Finally, Section 2.4 presents a brief discussion of the principle of constituent ordering in HPSG, as is presented in Pollard and Sag (1987)[Chapter 7].

2.1 The Formalism

In HPSG every linguistic entity is modelled as a 'typed feature structure', where the kind of object being described by a particular feature structure is indicated by its type, and each feature in the structure, which is said to be 'appropriate' for the type in question, is constrained to have as its value an object of a particular type, which *may* further have certain appropriate features of its own. Furthermore, types are organized in an inheritance hierarchy based on a 'subsumption' relation, where more specific types inherit the properties (e.g. appropriate features) of their 'supertypes' (i.e. ancestors in the hierarchy).

The most prominent type in the hierarchy is the type *sign*, in that every linguistic description – be it a word or a phrase – is modelled as an object of this type. Figure 2.1 shows the feature structure for an object of type *sign* (in the familiar attribute-value matrix (AVM) notation) with appropriate features and the types of the values those features may take, some of which are themselves complex feature structures. Following the standard notation, type names are written in italics and feature names in small capitals.

As shown in Figure 2.1, every object of type *sign* has the two appropriate features PHON and SYNSEM, which take values of type *list(phonstring)* and *synsem*, respectively. The PHON value of a sign encodes the phonological information about it, and the SYNSEM value represents its syntactic and semantic properties.

SYNSEM is a particularly important feature of all kinds of signs, in that it brings together all and only those features of a sign that can be selected for by other signs via certain means in the grammar, such as argument selection by heads and head selection by adjuncts. The type *synsem* has the two appropriate features LOCAL and NONLOCAL with

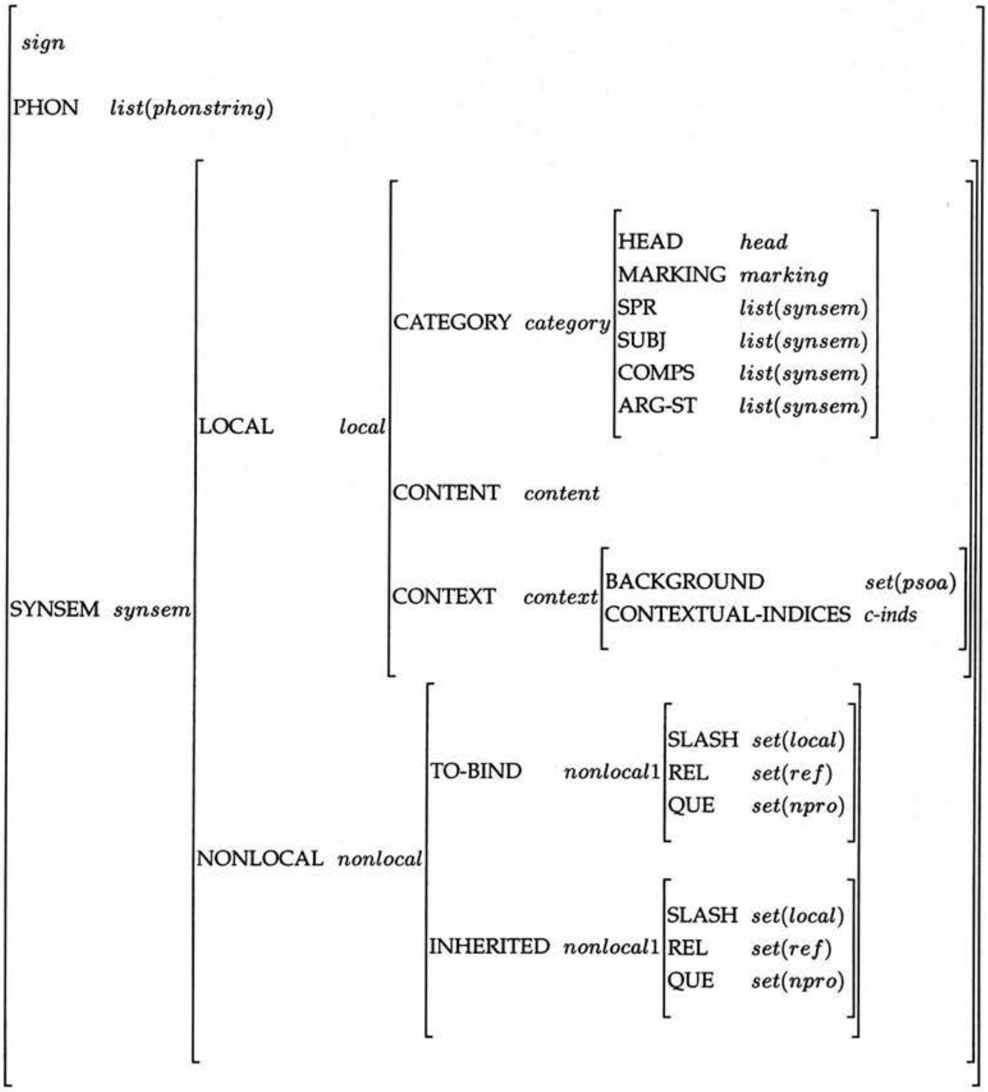


Figure 2.1: The feature structure for an object of type *sign*.

the value types *local* and *nonlocal*, respectively. The LOCAL value has three features of its own, namely CATEGORY, CONTENT and CONTEXT.

The CATEGORY value of a sign encodes mainly its syntactic features, such as the syntactic category (via the HEAD value) and subcategorization requirements (via the valence features SPR (specifier), SUBJ (subject) and COMPS (complements)[1]). The HEAD value is an object of type *head*, with the two subtypes *substantive (subst)* and *functional (func)*. The former has the four subtypes *noun, verb, adjective* and *preposition* (or *postposition)*, and the latter has the two subtypes *determiner* and *marker*. Some of these subtypes have appropriate features of their own. For example, *noun* has the feature CASE, and *verb* has the feature VFORM. In addition, the type *subst* has the feature MOD (which enables adjuncts to select the heads they modify, see below). Similarly, the type *func* has the feature SPEC (which enables specifiers and markers to select their heads).

All three valence features mentioned above (i.e. SPR, SUBJ and COMPS) take values of type *list(synsem)*, which essentially encode the kinds of *synsem* objects the respective arguments may have as their SYNSEM value. The main function of the ARG-ST feature is related to the principles of HPSG Binding Theory (see Pollard and Sag (1994)[Chapter 6]). Basic lexemes in the lexicon are assumed to obey a canonical constraint which requires the values of the three valence features to add up to the ARG-ST value. Outputs of certain lexical rules, however, violate this constraint, in that their valence lists together form a proper sublist of the ARG-ST value (Pollard and Sag (1994)[page 379]; see also the next section, page 22).

The CONTENT value constitutes the context-independent semantic content of the sign and the CONTEXT value constitutes certain context-dependent linguistic information. The CONTENT value is an object of type *content*, with the subtypes *nominal-object (nom-obj)* (e.g. the content of nominals), *parametrized-state-of-affairs (psoa)* (e.g. the content of verbs), and *quantifier (quant)* (e.g. the content of determiners).

The function of the NONLOCAL feature reveals itself in the analysis of unbounded dependency constructions (UDCs). The NONLOCAL value has the two appropriate features INHERITED and TO-BIND. The INHERITED value is used to pass up the necessary information for a nonlocal dependency from where it is first introduced to the point

[1]Pollard and Sag (1994)[Chapters 1–8] assume a single *list*-valued SUBCAT feature, whose value indicates all three kinds of valence requirements of a lexical head. However, later in Chapter 9, Pollard and Sag introduce the three valence features mentioned here (see that chapter for the relevant motivations), while still retaining the SUBCAT feature, which is later renamed ARG-S by Sag and Fodor (1994), and ARG-ST by Sag (to appear).

where it is bound off, by passing it successively from daughter to mother via a universal principle (the Nonlocal Feature Principle, see page 22). The TO-BIND value, on the other hand, guarantees that those nonlocal dependencies that become bound off are subtracted from the set of respective nonlocal feature values that are passed up to the mother, via the same principle.

The type *sign* has the two subtypes *word* and *phrase*. Apart from the features mentioned above that are appropriate for any kind of sign object, phrasal signs also have a DTRS feature, which corresponds to the immediate constituent structure of the sign in question. Accordingly, the DTRS value is an object of type *constituent-structure (cons-struc)*, which has the two subtypes *headed-structure (head-struc)* (with the appropriate feature HEAD-DTR), and *coordinate-structure (coord-struc)*. The type *head-struc* further has the six subtypes *head-subj-struc* (with the appropriate feature SUBJ-DTR), *head-spr-struc* (with the appropriate feature SPR-DTR), *head-comp-struc* (with the appropriate feature COMP-DTRS), *head-adjunct-struc* (with the appropriate feature ADJUNCT-DTR), *head-marker-struc* (with the appropriate feature MARKER-DTR), and *head-filler-struc* (with the appropriate feature FILLER-DTR).

One important mechanism in HPSG is the notion of *structure-sharing*, which is represented by identically numbered tags in feature structures. It essentially implies that two different features in a feature structure share exactly the same linguistic object as their value.

2.2 Principles and Schemata

HPSG is traditionally viewed – for expository purposes as Sag (to appear) puts it – in terms of a small set of highly generalized immediate dominance (ID) schemata that licence phrases with certain types of DTRS values by combining certain other signs (words or phrases).

Phrases are subject to a set of principles, some of which essentially let the information encoded in lexical entries be projected up to the phrase level in the grammar. The most prominent of those principles are the Head Feature Principle and the Valence Principle that are stated below in (2.1) and (2.2), respectively.

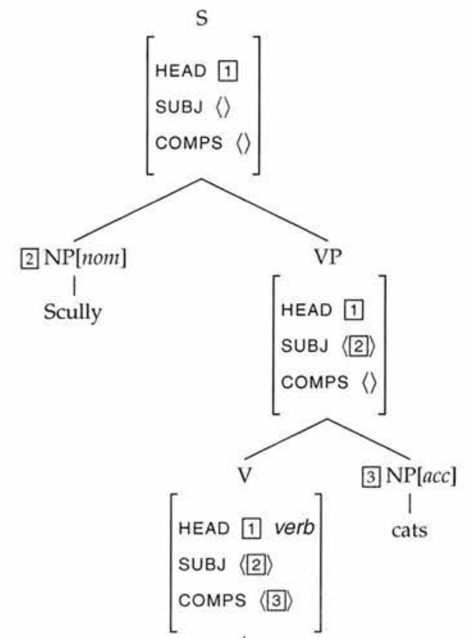
(2.1) **The Head Feature Principle (HFP)**
The HEAD value of a headed phrase is structure-shared with that of its head daughter.

(2.2) **The Valence Principle (VALP)**
For each valence feature F, the F value of a headed phrase is the head daughter's F value minus the realized non-head daughters (e.g. Subj-Dtr, Complement-Dtrs, Spr-Dtr).

Roughly speaking, the HFP guarantees that the HEAD value of any sign is always structure-shared with that of its phrasal projections, and the VALP ensures that any unsaturated subcategorization requirements of a sign are passed up to its mother. Consider, for example, the (rather simplified) structure for the sentence in (2.3), illustrated in (2.4). Ignore, for the time being, how these *sign* objects are combined, which relates to two of the ID schemata presented below.[2]

(2.3) Scully loves cats.

(2.4)



Note that on the VP node in (2.4), the HEAD value is structure-shared with that of *'loves'* (tag [1]), via the HFP. In addition, the VALP ensures that the NP[*nom*] subject required by *'loves'* is passed up to the mother VP node (tag [2]), whereas the NP[*acc*] complement

[2]In this structure – and the ones to come – only the feature values that are essential to the ongoing discussion are explicitly shown, and full path names are systematically replaced by single feature names, for purposes of both increasing the readibility and also keeping the structures at a reasonable size.

is not, since the corresponding complement daughter is realized by the NP *'cats'*. On the S node, the HEAD value is structure-shared with that of the VP, again via the HFP. Furthermore, the SUBJ list is now empty, since the corresponding subject daughter is realized by *'Scully'*.

The six schemata of HPSG, which specify partial information about universally available types of phrases, are summarized in (2.5) in the familiar phrase structure notation. The left hand side of each schema in this notation corresponds to the feature structure of the mother and the right hand side to those of its daughters. It is important to note that the order of the entries on the right hand side of a schema in this notation is not specified in any way. Any constraint imposed by a schema on any of its daughters is annotated just below that daughter's entry. Unless otherwise stated all daughter values are phrasal, except for the non-head daughter values in the first three schemata (2.5a-c), which are rather of type *list(phrase)*. Although not explicitly shown below, the DTRS value of the mother in each case is restricted to the respective subtype of the type *head-struc* mentioned in Section 2.1 (i.e. *head-spr-struc, head-subj-struc*, etc.)

(2.5) a. **Head-Specifier Schema:**

X → Head-Dtr , Spr-Dtr
[COMPS <>]

b. **Head-Subj Schema:**

X → Head-Dtr , Subj-Dtr
[COMPS/SPR <>]

c. **Head-Complement Schema:**

X → Lexical Head-Dtr , Comp-Dtrs

d. **Head-Marker Schema:**

X → Head-Dtr , Lexical Marker-Dtr
[HEAD *marker*]

e. **Head-Adjunct Schema:**

X → Head-Dtr , Adjunct-Dtr
[SYNSEM $\boxed{1}$] [MOD $\boxed{1}$]

f. **Head-Filler Schema:**

$$
X \rightarrow \underset{\left[\begin{array}{ll}\text{HEAD} & verb[\text{VFORM}\,fin] \\ \text{SPR} & <> \\ \text{SUBJ} & <> \\ \text{COMPS} & <> \\ \text{INHER}|\text{SLASH} & \{\ldots,\boxed{1},\ldots\} \\ \text{TO-BIND}|\text{SLASH} & \{\boxed{1}\}\end{array}\right]}{\text{Head-Dtr}}, \quad \underset{[\text{LOCAL}\ \boxed{1}]}{\text{Filler-Dtr}}
$$

The head-specifier and head-subject schemata (2.5a,b) both require the head daughter to have an empty COMPS value, meaning that the head daughter in these schemata must have already 'consumed' all its complements. In addition, the head-subject schema (2.5b) also constrains the head daughter to have an empty SPR value.

Note that the VP node in (2.4) above is licensed by the head-complement schema (2.5c), the verb *'loves'* being the lexical head daughter and the NP *'cats'* being the only complement daughter. And the S node is licensed by the head-subject schema (2.5b), with the subject daughter being realized by the NP *'Scully'* and the head daughter by the VP *'loves cats'*.

The head-adjunct schema (2.5e) requires the head daughter to structure-share its SYNSEM value with the MOD value of the adjunct daughter.[3] This constraint is simply a formalization of the assumption that adjuncts select the heads they modify, via the MOD feature. This assumption is motivated mainly by semantic considerations. Adjunct daughters are considered to be the semantic heads of the phrases they occur in, and hence they contribute directly to the content of their mothers. To that end, selection of heads by adjuncts in the lexicon makes it possible to incorporate the content of a head selected by an adjunct into the content of that adjunct (via structure-sharing). For example, (2.6) shows the CATEGORY|HEAD and the CONTENT values of the lexical entry for the adjective *'small'*. (By convention, a feature structure following a colon refers to the CONTENT value of the structure before the colon.) Note that the CONTENT value of the adjective is of type *nominal-object* whose INDEX value is structure-shared (tag $\boxed{1}$) with that of the head noun selected via the MOD feature, and whose restriction set includes

[3]Note that structure-sharing only applies within one single feature structure. Although the feature structures of the daughters in this representation of the schemata seem to be totally independent from each other, they are in fact nothing but sub-structures within one big feature structure, i.e. that of the mother.

the restrictions imposed by the head noun (tag $\boxed{2}$), as well as the one imposed by the adjective itself.

$$(2.6)\quad \begin{bmatrix} \text{CATEGORY|HEAD} & adj \begin{bmatrix} \text{MOD } N' : \begin{bmatrix} \text{INDEX} & \boxed{1} \\ \text{RESTR} & \boxed{2} \end{bmatrix} \end{bmatrix} \\ \text{CONTENT} & nom\text{-}obj \begin{bmatrix} \text{INDEX} & \boxed{1} \\ \text{RESTR} & \left\{ \begin{bmatrix} small & \\ \text{INSTANCE} & \boxed{1} \end{bmatrix} \right\} \cup \boxed{2} \end{bmatrix} \end{bmatrix}$$

The adjunct then structure shares its CONTENT value with that of the mother in the head-adjunct structure it occurs in, via the Semantics Principle in (2.7).[4]

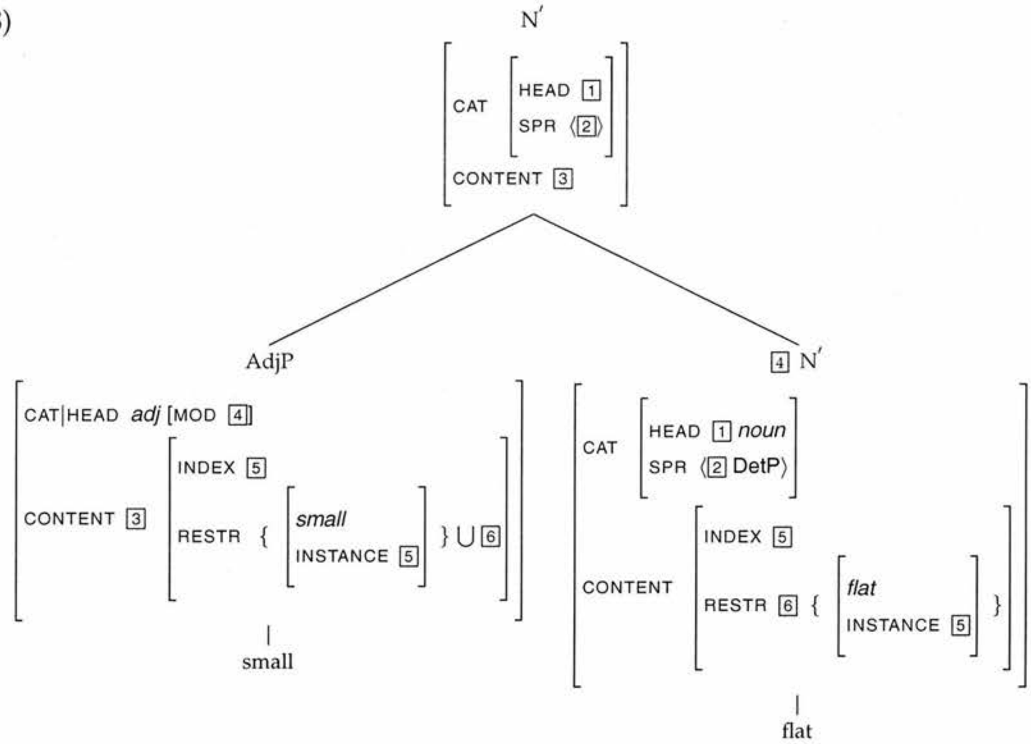
(2.7) **The Semantics Principle**

In a headed phrase, the CONTENT value is structure-shared with that of the adjunct daughter if any, and with that of the head daughter otherwise.

Consider, for example, the structure of the N$'$ *'small flat'*, given in (2.8), licensed by the head-adjunct schema (2.5e). Notice the combined impact of the structure-sharings imposed by the schema (2.5e), the lexical entry for *'small'* in (2.6) and the Semantics Principle in (2.7), on this structure (i.e. tags $\boxed{3} - \boxed{6}$). In addition, the HFP, (2.1), constrains the HEAD value of the mother N$'$ node to be structure-shared with that of its head daughter N$'$ (tag $\boxed{1}$), and the VALP, (2.2), ensures that the unsaturated SPR specification of that head daughter is passed up to its mother (tag $\boxed{2}$).

[4]This version of the Semantics Principle is in fact later revised in Pollard and Sag (1994) to provide an account of quantification, making use of a version of the quantifier storage technique of Cooper (1975) and Cooper (1983). And that version is further revised by Pollard and Yoo (to appear) to solve certain scope problems it faces in examples with raising verbs and propositional attitude verbs. I do not deal with the issue of quantification here, and hence adopt the version given in (2.7).

(2.8)



The CONTENT value of the mother N′ in (2.8) (which is structure-shared with that of its adjunct daughter *'small'*) is explicitly given in (2.9), the second psoa in the RESTR set coming from the head daughter *'flat'*.

(2.9) $$\begin{bmatrix} \textit{nom-obj} & \\ \text{INDEX} & \boxed{5} \\ \text{RESTR} & \left\{ \begin{bmatrix} \textit{small} & \\ \text{INSTANCE} & \boxed{5} \end{bmatrix}, \begin{bmatrix} \textit{flat} & \\ \text{INSTANCE} & \boxed{5} \end{bmatrix} \right\} \end{bmatrix}$$

Similarly, specifiers and markers are assumed to select their heads via the SPEC feature. However, that constraint is imposed by a separate principle called the SPEC Principle, given in (2.10), and hence not specified explicitly above in the relevant schemata (i.e. (2.5a,d)).

(2.10) **The SPEC Principle**

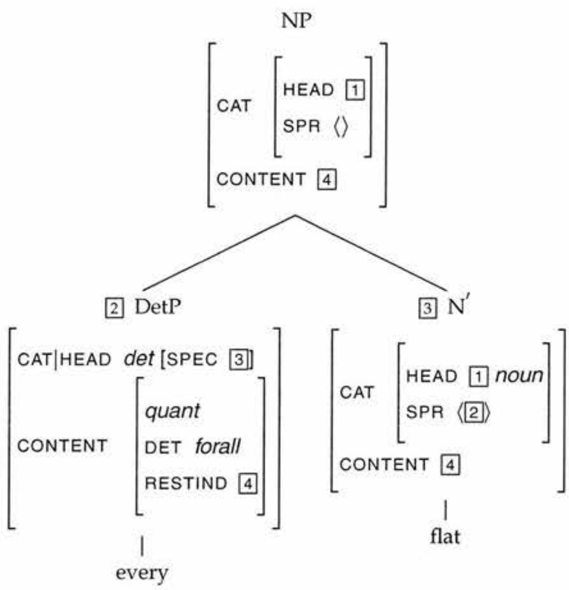
If a non-head daughter in a headed phrase bears a SPEC value, it is structure-shared with the SYNSEM value of the head daughter.

Consider now (2.11), which shows the CATEGORY|HEAD and the CONTENT values of the lexical entry for the determiner *'every'*. Note that the CONTENT value of the determiner is a *quantifier* object whose RESTRICTED-INDEX (RESTIND) value is structure-shared (tag $\boxed{1}$) with the whole CONTENT value of the selected N′ (which includes both the index of that N′ and a non-empty restriction on that index).

(2.11) $$\begin{bmatrix} \text{CATEGORY|HEAD} & det \left[\text{SPEC } N' : \boxed{1}\right] \\ \text{CONTENT} & quant \begin{bmatrix} \text{DET} & forall \\ \text{RESTIND} & \boxed{1} \end{bmatrix} \end{bmatrix}$$

(2.12) shows the structure for the NP *'every flat'*, licensed by the head-specifier schema (2.5a). Notice that the specifier daughter *'every'* and the head daughter *'flat'* reciprocally select for each other via the features SPEC and SPR. The structure-sharing between the SPEC value of the specifier daughter and the SYNSEM value of the head daughter (tag $\boxed{3}$) is imposed by the SPEC Principle (2.10). On the other hand, the structure-sharing between the element in the SPR list of the head daughter and the SYNSEM value of the specifier daughter (tag $\boxed{2}$) is imposed by the VALP, which further guarantees that that element is not passed up to the mother NP node, leaving the SPR list of that node empty. Furthermore, the HFP requires the mother NP to structure-share its HEAD value with that of its head daughter (tag $\boxed{1}$). And finally, the Semantics Principle in (2.7) constrains the CONTENT value of the mother NP to be structure-shared with that of its head daughter (tag $\boxed{4}$).[5]

(2.12)



The head-marker schema (2.5d) restricts the marker daughter to have a HEAD value of type *marker*. In addition, the Marking Principle in (2.13) guarantees that markers structure-share their MARKING value with that of their mother.

[5]Recall that I do not concern myself with the issue of quantification here, and hence totally omit the way the semantic content of the determiner contributes to that of the mother in (2.12).

(2.13) **The Marking Principle**
In a headed phrase, the MARKING value is structure-shared with that of the marker daughter if any, and with that of the head daughter otherwise.

Finally, the head-filler schema (2.5f) restricts the head daughter to be a *finite* S (via the first four constraints). It also requires the LOCAL value of the filler daughter to be structure-shared with one of the elements in the INHER|SLASH set of the head daughter, which is passed up from one of its daughters via the Nonlocal Feature Principle (NFP) given in (2.14). Furthermore, the TO-BIND|SLASH set is restricted to contain the very same *local* object so that the relevant nonlocal dependency gets bound off on the mother again via the NFP.

(2.14) **The Nonlocal Feature Principle (NFP)**
In a headed phrase, for each nonlocal feature F, the value of SYNSEM|NONLOCAL|INHERITED|F is the set difference of the union of the values on all daughters and the value of SYNSEM|NONLOCAL|TO-BIND|F on the head daughter.

Before giving an example structure to illustrate the use of the head-filler schema and the NFP, I should first note that in this dissertation I assume the traceless account of UDCs proposed in Chapter 9 of Pollard and Sag (1994). This analysis relies on the assumption that the information about a nonlocal dependency is first introduced in the structure by the lexical head that licenses the missing element, rather than an empty category in the grammar. To that end, Pollard and Sag introduce a number of extraction lexical rules to deal with the extraction of complements, subjects and adjuncts. Here, I only provide a rather simplified version of their complement extraction lexical rule, (2.15), to explain the basic idea behind the traceless account. (See Pollard and Sag (1994)[pages 376-388] for the precise formulation of this rule, and the lexical rules for subject and adjunct extraction.)

(2.15) $$\begin{bmatrix} \text{COMPS} & \langle \ldots, [\text{LOC}\ \boxed{1}], \ldots \rangle \\ \text{INHER|SLASH} & \boxed{2} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} & \langle \ldots \rangle \\ \text{INHER|SLASH} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

(2.15) basically takes as input a lexical entry with a non-empty COMPS value and outputs a lexical entry that is exactly the same except that one of the elements in the COMPS list has been removed and its LOCAL value has been placed within the INHER|SLASH set (in addition to any other *local* objects that are already in the INHER|SLASH set of the input).[6] Thus, for example, the extraction of the only complement of a transitive verb such as *'loves'* would instantiate this rule as in (2.16).

(2.16)
$$\begin{bmatrix} \text{COMPS} & \langle \text{NP}\,[\text{LOC}\ \boxed{1}[\text{CASE}\ \textit{acc}]] \rangle \\ \text{INHER|SLASH} & \{\} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} & \langle\rangle \\ \text{INHER|SLASH} & \{\boxed{1}\} \end{bmatrix}$$

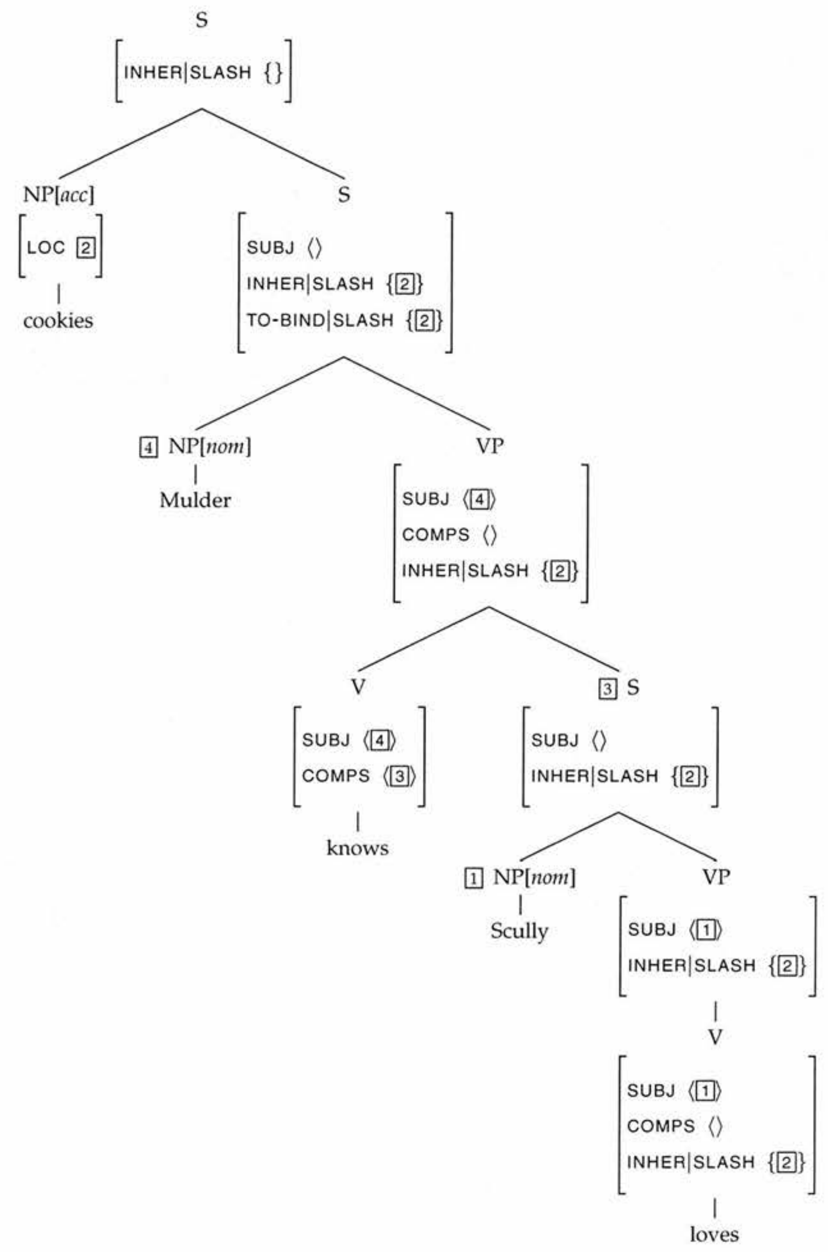
The output lexical entry in (2.16) would then license, for example, the structure of the topicalization example in (2.17), given in (2.18).

(2.17) Cookies$_i$, Mulder knows Scully loves $\__i$.

All phrasal nodes in (2.18) are licensed by either the head-subject or the head-complement schema ((2.5b,c), respectively) except for the top S node, licensed by the head-filler schema (2.5f). Notice that the non-empty INHER|SLASH value introduced by the lexical head *'loves'* is passed up by the NFP all the way up to the top S node, where it is bound off by the filler daughter *'cookies'*. The non-empty TO-BIND|SLASH value on the head daughter of that node (imposed by the head-filler schema) guarantees that the INHER|SLASH element corresponding to the nonlocal dependency that is just being discharged does not pass up any further (cf. the NFP), leaving the INHER|SLASH value on the mother S node empty.

[6]Note that the extracted complement is still present in the ARG-ST list of the output, hence being subject to the principles of the HPSG Binding Theory (Pollard and Sag (1994)[Chapter 7]). Outputs of (2.15) therefore violate the canonical constraint on words in the lexicon which requires the values of their valence features to add up to the ARG-ST value; see page 14.

(2.18)



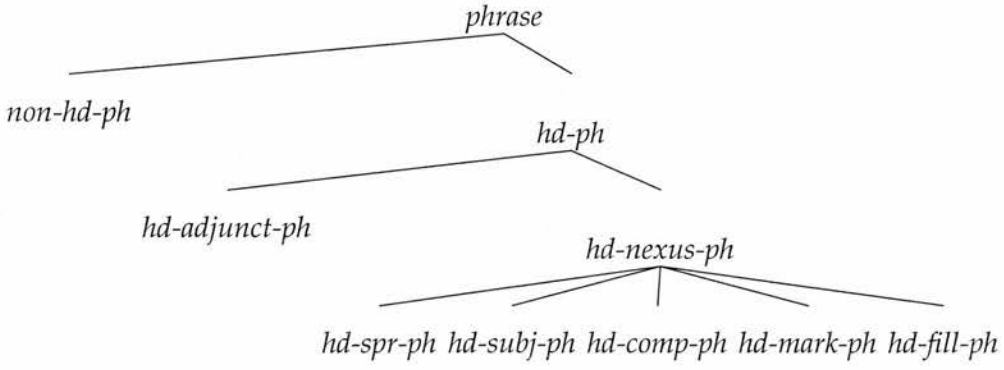
2.3 An Alternative View

Constraint-based theories in general are highly lexicalized, meaning that they aim to represent as much of the linguistic information as possible in the lexicon. The presentation of HPSG above achieves this goal to a certain extent by the use of the type hierarchy and highly articulated lexical entries, but also leaves a good deal of information outside the lexicon, in the form of principles and phrase structure schemata. Note that in HPSG words and phrases are represented in essentially the same way, that is in terms of typed feature structures. As Sag (to appear) points out, schemata then can simply be viewed as descriptions of feature structures of type *phrase*, just as lexical entries are descriptions of feature structures of type *word*. In addition, principles can further be formulated as additional constraints on phrases of certain types.

> "The grammar of a language then is just the specification of its types and the constraints that govern those types, including the inventory of words that belong to the various lexical types." (Sag (to appear)[page 7])

The description of schemata and principles in this way, however, requires use of a hierarchy of phrases that is parallel to the hierarchical classification of DTRS values (i.e. hierarchy of the type *constituent-structure*) mentioned in Section 2.1 (cf. page 15). Instead of introducing such a hierarchy, which would obviously be redundant, Sag eliminates the hierarchy of constituent structures and the DTRS feature from the formalism, and devises a feature geometry of phrasal signs that directly reflects the constituent structures of phrases. Here I adopt a simpler version of the grammar proposed by Sag that is nevertheless sufficient for the purposes of this dissertation. I assume the hierarchy of phrasal signs given in (2.19).

(2.19)



According to (2.19) phrases are classified as either *headed-phrase (hd-ph)* or *non-headed-phrase (non-hd-ph)*. The type *hd-ph* has the two subtypes *head-adjunct-phrase (hd-adjunct-ph)* and *head-nexus-phrase (hd-nexus-ph)*, and the latter has further the five subtypes *head-specifier-phrase (hd-spr-ph)*, *head-subject-phrase (hd-subj-ph)*, *head-complement-phrase (hd-comp-ph)*, *head-marker-phrase (hd-mark-ph)*, and *head-filler-phrase (hd-fill-ph)*. The type *hd-ph* has the two appropriate features HEAD-DTR (HD-DTR) and NON-HEAD-DTRS (NON-HD-DTRS) with the value types *sign* and *list(phrase)*, respectively.

The Head Feature Principle, given in (2.1) above, can then be formulated as a constraint on phrases of type *hd-ph* as in (2.20) (following the formalization in Sag (to appear)).

(2.20) Head Feature Principle (HFP):

$$hd\text{-}ph \Rightarrow \begin{bmatrix} \text{HEAD} & \boxed{1} \\ \text{HD-DTR} & [\text{HEAD}\ \boxed{1}] \end{bmatrix}$$

The formulation of the Valence Principle as a constraint on headed phrases is again borrowed from Sag (to appear). This constraint, given in (2.21), makes use of default values indicated by '/'.[7]

(2.21) Valence Principle (VALP):

$$hd\text{-}ph \Rightarrow \begin{bmatrix} \text{SUBJ} & /\ \boxed{1} \\ \text{SPR} & /\ \boxed{2} \\ \text{COMPS} & /\ \boxed{3} \\ \text{HD-DTR} & \begin{bmatrix} \text{SUBJ} & /\ \boxed{1} \\ \text{SPR} & /\ \boxed{2} \\ \text{COMPS} & /\ \boxed{3} \end{bmatrix} \end{bmatrix}$$

(2.21) imposes the structure-sharing between each of the valence features of a headed phrase and that of its head daughter, unless it is an instance of some more specific subtype of *hd-ph* that says otherwise.

Another default constraint on headed phrases proposed by Sag (and also assumed here) is the Empty COMPS Constraint (ECC), stated in (2.22), which guarantees that the head daughter of a headed phrase always has an empty COMPS value, unless it is an instance of some more specific subtype of *hd-ph* that says otherwise.

(2.22) Empty COMPS Constraint (ECC):

$$hd\text{-}ph \Rightarrow [\text{HD-DTR}\ [\text{COMPS}\ /\ \langle\rangle]]$$

I further assume one more default constraint on headed phrases, namely the Marking Principle in (2.23), which ensures the structure-sharing between the MARKING value of a headed phrase and that of its head daughter, unless it is an instance of some more specific subtype of *hd-ph* that says otherwise (cf. the descriptive Marking Principle (2.13) in Section 2.2).

(2.23) Marking Principle:

$$hd\text{-}ph \Rightarrow \begin{bmatrix} \text{MARKING} & /\ \boxed{1} \\ \text{HD-DTR} & [\text{MARKING}\ /\ \boxed{1}] \end{bmatrix}$$

[7]Sag (to appear) states that use of default values in this way follows the framework for default unification outlined in Lascarides *et al.* (1996).

In addition, I assume the following formulation of the Nonlocal Feature Principle (cf. (2.14) in Section 2.2) as a constraint on headed phrases:

(2.24) Nonlocal Feature Principle (NFP):
hd-ph ⇒ For each feature F appropriate for the type *nonlocal1*

$$\begin{bmatrix} \text{INHERITED}|\text{F} & (\text{INH-F}(\boxed{1}) \bigcup \boxed{3}) - \boxed{2} \\ \text{HD-DTR} & \begin{bmatrix} \text{INHERITED}|\text{F} & \boxed{3} \\ \text{TO-BIND}|\text{F} & \boxed{2} \end{bmatrix} \\ \text{NON-HD-DTRS} & \boxed{1} \end{bmatrix}$$

where INH-F relates a list of signs to the set union of their INHERITED|F values in the following way:

INH-F(*elist*) = *eset*;

$\text{INH-F}(\langle[\text{INHERITED}|\text{F}\ \boxed{1}] \mid \boxed{2}\rangle) = \boxed{1} \bigcup \text{INH-F}(\boxed{2})$

and '−' is the 'set difference operator'

For each NONLOCAL feature F, (2.24) constrains the INHERITED|F value of a headed phrase as the set difference of the union of the INHERITED|F values of all daughters, and the TO-BIND|F value of the head daughter.

Finally, I formulate the SPEC Principle (cf. (2.10) in Section 2.2) as a constraint on head-specifier and head-marker phrases as shown below in (2.25), which simply ensures that in every head-specifier and head-marker phrase the SPEC value of the non-head daughter is structure-shared with the SYNSEM value of the head daughter.

(2.25) SPEC Principle:

$$\textit{hd-spr-ph}/\textit{hd-mark-ph} \Rightarrow \begin{bmatrix} \text{NON-HD-DTRS} & \langle[\text{HEAD}\ [\text{SPEC}\ \boxed{1}]]\rangle \\ \text{HD-DTR} & [\text{SYNSEM}\ \boxed{1}] \end{bmatrix}$$

Table 2.1 summarizes the constraints assumed on each headed phrase type in the hierarchy given above in (2.19). The first column in Table 2.1 refers to the individual phrase types mentioned in (2.19), and the second column specifies the constraints assumed on each of those types. The third column ('ISA') refers to the immediate supertype of the type in the first column in each case, according to the hierarchy in (2.19).

Each phrase type in the first column of Table 2.1 is assumed to inherit the constraints on all its supertypes (according to the hierarchy in (2.19)), except that in the case of a default value in conflict with a non-default specification, it is the non-default one that is

TYPE	CONSTRAINTS	ISA
phrase		*sign*
hd-ph	HFP, VALP, ECC, NFP, Marking Principle	*phrase*
hd-adj-ph	[CONTENT $\boxed{1}$, HD-DTR [*phrase*, SYNSEM $\boxed{2}$], NON-HD-DTRS $\langle$[HEAD|MOD $\boxed{2}$, CONTENT $\boxed{1}$]$\rangle$]	*hd-ph*
hd-nexus-ph	[CONTENT $\boxed{1}$, HD-DTR [CONTENT $\boxed{1}$]]	*hd-ph*
hd-spr-ph	SPEC Principle, [SPR <>, HD-DTR [*phrase*, SPR $\langle\boxed{1}\rangle$], NON-HD-DTRS $\langle$[SYNSEM $\boxed{1}$]$\rangle$]	*hd-nexus-ph*
hd-subj-ph	[SUBJ <>, HD-DTR [*phrase*, SUBJ $\langle\boxed{1}\rangle$, SPR <>], NON-HD-DTRS $\langle$[SYNSEM $\boxed{1}$]$\rangle$]	*hd-nexus-ph*
hd-comp-ph	[COMPS <>, HD-DTR [*word*, COMPS $\langle\boxed{1}, \ldots, \boxed{n}\rangle$], NON-HD-DTRS $\langle$[SYNSEM $\boxed{1}$], ..., [SYNSEM $\boxed{n}$]$\rangle$]	*hd-nexus-ph*
hd-mark-ph	SPEC Principle, [MARKING $\boxed{1}$, HD-DTR *phrase*, NON-HD-DTRS $\langle$[*word*, MARKING $\boxed{1}$]$\rangle$]	*hd-nexus-ph*
hd-fill-ph	[HD-DTR [*phrase*, HEAD $verb[fin]$, SPR <>, SUBJ <>, INHER|SLASH $\{\ldots, \boxed{1}, \ldots\}$, TO-BIND|SLASH $\{\boxed{1}\}$], NON-HD-DTRS $\langle$[LOCAL $\boxed{1}$]$\rangle$]	*hd-nexus-ph*

Table 2.1: A summary of the constraints on particular types in the hierarchy of phrases.

assumed to hold. Thus, the HFP, (2.20), and the NFP, (2.24), are inherited by all subtypes of *hd-ph*.

The ECC (cf. (2.22)), on the other hand, is inherited by all headed-phrases, except for head-complement phrases, whose head daughter *may* have a non-empty COMPS list,[8] with its elements structure-shared with the SYNSEM values of the corresponding non-head daughters of the head-complement phrase.

Similarly, the Marking Principle, (2.23), is inherited by all headed phrases, except for head-marker phrases, whose MARKING value is structure-shared with that of the non-head daughter, rather than the head daughter.

Likewise, the three default specifications imposed by VALP, (2.21), apply to all subtypes of *hd-ph*, except that the specifications related to the SPR, SUBJ and COMPS features are overriden by the corresponding non-default specifications imposed on types *hd-spr-ph*, *hd-subj-ph* and *hd-comp-ph*, respectively.

And finally, the CONTENT value of a head-adjunct phrase is structure-shared with that of its non-head daughter, while the CONTENT value of any instance of *hd-nexus-ph* (and hence any instance of any of its subtypes) is structure-shared with that of the head daughter by the relevant constraints on the types *hd-adjunct-ph* and *hd-nexus-ph* (in line with the Semantics Principle in (2.7) on page 19).

In view of the grammar in Table 2.1, the structure of (2.3) above is repeated below in (2.26) (this time in the feature structure representation rather than the phrase structure tree representation as before). For the sake of simplicity, I assume that the PHON value of each phrasal sign in this structure is constrained to be the concatenation of the PHON values of its daughters (in a way that is consistent with the word order properties of English). The HFP guarantees that the HEAD value of every headed phrase in (2.26) is structure-shared with that of its head daughter (cf. for example, tag [4]). Also, the VALP ensures that any unsaturated subcategorization requirements of a sign is passed up to its mother, hence the NP[*nom*] subject requirement of the finite verb *'loves'* passed up to its mother head-complement phrase (tag [6]). In addition, the relevant constraints on the type *hd-subj-ph* impose the structure-sharing between the SYNSEM value of the non-head daughter of the head-subject phrase and the element in the SUBJ list of the head daughter of that phrase (tag [6] again), and further restrict the SUBJ list of the head-subject

[8] I assume that there are head-complement phrases with only a lexical (i.e. of type *word*) head daughter (with an empty COMPS list) and an empty list of non-head daughters, to make a distinction between, say, the word *'Edinburgh'* and the NP *'Edinburgh'* with only the word *'Edinburgh'* as its head daughter.

phrase to be empty. Similarly, the head daughter of that head-subject phrase satisfies the corresponding constraints on the type *hd-comp-ph*. Finally, notice that the constraint that restricts the CONTENT value of a head-nexus phrase to be structure-shared with that of its head daughter is satisfied by the head-subject and head-complement phrases in the structure (tag $\boxed{5}$). (By convention, any constraint written as a subscript right after an NP is assumed to refer to the INDEX value of that NP.)

(2.26)

$$
\left[\begin{array}{ll}
\textit{hd-subj-ph} & \\
\text{PHON} & \langle \boxed{1}, \boxed{2}, \boxed{3} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle\rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\
\text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\ \textit{Scully} \rangle \\ \text{SYNSEM} & \boxed{6}\ \text{NP}_{\boxed{8}}[\textit{nom}] \end{array}\right] \right\rangle \\
\text{HD-DTR} & \left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{PHON} & \langle \boxed{2}, \boxed{3} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle\boxed{6}\rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\
\text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{3}\ \textit{cats} \rangle \\ \text{SYNSEM} & \boxed{7}\ \text{NP}_{\boxed{9}}[\textit{acc}] \end{array}\right] \right\rangle \\
\text{HD-DTR} & \left[\begin{array}{ll}
\textit{word} & \\
\text{PHON} & \langle \boxed{2}\ \textit{loves} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4}\ \textit{verb [fin]} \\ \text{SUBJ} & \langle\boxed{6}\rangle \\ \text{COMPS} & \langle\boxed{7}\rangle \\ \text{CONTENT} & \boxed{5} \left[\begin{array}{ll} \textit{love} & \\ \text{LOVER} & \boxed{8} \\ \text{LOVED} & \boxed{9} \end{array}\right] \end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]
$$

2.4 Constituent Ordering

Schemata in HPSG (that is, the most specific subtypes of *phrase* such as *hd-comp-ph*, *hd-subj-ph*, etc., in terms of the terminology of Section 2.3) only specify the immediate constituent structures of phrases in a language. They do not constrain the temporal order in which the phonological realizations of constituents occur in utterances (Pollard and Sag (1987)[Chapter 7]).

It is assumed in HPSG that there is a language specific Constituent Ordering Principle (COP) for each language, which can be expressed as a constraint on phrasal signs in the following way:[9]

(2.27) Constituent Ordering Principle:

$$phrase \Rightarrow \begin{bmatrix} \text{PHON} & \text{order-constituents}(\boxed{1}, \boxed{2}) \\ \text{NON-HD-DTRS} & \boxed{1} \\ \text{HD-DTR} & \boxed{2} \end{bmatrix}$$

'Order-constituents' is a function (whose precise formulation varies from language to language) which, in general, is assumed to return the disjunction of all permutations of the phonology values of all the daughters in a phrase. At least one of those permutations is required to be consistent with all the restrictions on word order for the language in question, expressed by a number of 'linear precedence' (LP) constraints. An LP constraint for a certain language is a statement of the form 'X < Y', which simply means that for any phrase in that particular language, the phonological realization of any daughter with property X is constrained to temporally precede the phonological realization of any of its sisters with property Y.[10] English, for instance, is assumed to employ the following LP constraint, which states that in any phrasal sign of English, a daughter which is a lexical head is constrained to precede any of its sisters.[11]

(2.28) HEAD[*word*] < [ ]

In addition, the relative ordering of complements in a phrasal sign of English is assumed to be constrained by the LP constraint in (2.29).

(2.29) COMPLEMENT << COMPLEMENT

[9]In fact, the COP is formulated in Pollard and Sag (1987)[page 169] in the following way:

(i) $${}_{phrasal\text{-}sign}[\,] \Rightarrow \begin{bmatrix} \text{PHON} & \text{order-constituents}(\boxed{1}) \\ \text{DTRS} & \boxed{1} \end{bmatrix}$$

I slightly modify this formulation, as shown in (2.27), to bring it into line with the present formalism.

[10]It should be noted that the immediate dominance/linear precedence (ID/LP) distinction in the grammar was first introduced by Gazdar and Pullum (1981) in the framework of Generalized Phrase Structure Grammar (GPSG), in order to capture the generalizations over the linear ordering of constituents in different phrase structure rules.

[11]Again, the precise formulation of this LP constraint in Pollard and Sag (1987)[page 172] is in fact as follows:

(i) HEAD[LEX +] < [ ]

The feature LEX is assumed to take boolean values, namely '+' for lexical signs, and '-' for phrasal signs, in Pollard and Sag (1987).

'<<' is a special kind of restricted linear precedence constraint which only applies when the left-hand element is less oblique than the right-hand element (for instance, subjects are assumed to be less oblique than direct objects, and direct objects less oblique than indirect objects, etc.). Thus, (2.29) states that in any phrasal sign of English, any complement is constrained to precede any of its sisters which is a more oblique complement. One must note that the arguments of a lexical head in HPSG are (for independent reasons) ordered in the ARG-ST list of the head in an increasing order of obliqueness. (2.29) therefore makes essential use of the ARG-ST list of the head of the phrasal sign to which it applies.

These two LP constraints account, for example, for the particular order of the constituents in the following VP:

(2.30) $[_{VP}$ gave $[_{NP}$ Mary$]$ $[_{NP}$ a book$]]$

In (2.30), the lexical head *'gave'* precedes both its sisters, hence satisfying the LP constraint in (2.28). Also, the complement *'Mary'*, which is less oblique than its (complement) sister *'a book'*, precedes that sister, in line with the LP constraint in (2.29).

Note that we have made above the simplifying assumption that the COP is defined in terms of a function which simply concatenates the phonological realizations of constituents. It is also possible to define the COP in terms of a function which 'interleaves' (rather than orders) the phonological realizations of constituents to analyze, for example, the word order variation in free/semi-free word order languages. (See, for instance, the domain union analysis of German word order by Reape (1994).)

2.5 Summary

In this chapter, I have presented an overview of HPSG to provide the reader with sufficient background to follow the rest of the dissertation. In Section 2.1, I outlined the typed feature formalism that the theory relies on, and then, in Section 2.2, I summarized some of the basic concepts of the theory, namely the principles and schemata, in the way they are introduced in Pollard and Sag (1994). In Section 2.3, I presented an alternative view of the concepts in Section 2.2, based on Sag (to appear), which relies heavily on the type hierarchy, supporting a totally "lexicalized" characterization of HPSG grammars. Finally, in Section 2.4, I briefly discussed the principle of constituent ordering in HPSG, in the way it is presented in Pollard and Sag (1987)[Chapter 7].

Chapter 3

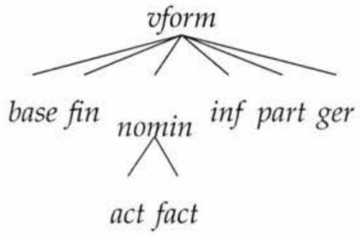
Turkish Grammar in HPSG

In this chapter, I examine certain phenomena in Turkish grammar within the framework of HPSG. I start out, in Section 3.1, by outlining the assumptions I make in relation to certain aspects of the grammar which I summarized in Chapter 1 (Section 1.3.1). Then, Section 3.2 investigates the apparent contrast between the obligatoriness of case-marking on a class of NPs, which I refer to as 'inherently specific', and other kinds of NPs, when they occur as direct objects. I propose a way of capturing this contrast within syntax, using a *syntactic* SPECIFICITY feature appropriate for nouns and determiners. In Section 3.3, I present an HPSG analysis of possessive NPs in Turkish, treating possessors as subjects. The main concern of this chapter is the issue of word order variation in Turkish, discussed in detail in Section 3.4. I focus on the syntactic aspects of this variation in simple and complex Turkish sentences, detailing the assumptions I make both to deal with the variation in the word order, and also to capture certain restrictions on that variation, within the HPSG framework.

3.1 Type Hierarchy

In Chapter 1 (Section 1.3.1), we saw that Turkish has a highly inflectional agglutinative morphology. Recall that Turkish has the following seven morphological cases, which I assume are subtypes of the type *case* in the hierarchy: *nominative (nom), genitive (gen), accusative (acc), dative (dat), ablative (abl), locative (loc),* and *instrumental (inst).* In addition, following the discussion on verbal morphology, I assume the following hierarchy for the type *vform* in Turkish grammar:

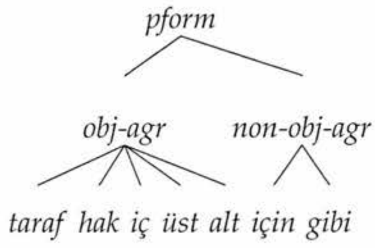
(3.1)



In addition to these main hierarchies for the types *case* and *vform*, in the discussion below, I will introduce a number of additional supertypes for certain subtypes of these two types, relying on the notion of 'multiple inheritance'. Also, recall that as well as nouns, nominalizations and participles can also be case-marked in Turkish. CASE is therefore defined as an appropriate feature for the types *noun*, *nomin*, and *part*, in the hierarchy.

We also saw in Chapter 1 (again Section 1.3.1) that certain postpositions in Turkish bear agreement morphology with their objects, and that those postpositions let their objects be dropped, while objects of postpositions without any agreement morphology are not allowed to be dropped (Kornfilt (1984)). It is interesting to note that exactly the same kind of contrast further arises in the case of scrambling and relativization of postpositional objects in Turkish (see Section 3.4.3 and Chapter 4 – Section 4.2.4 – respectively). Thus, I assume that the type *pform* in Turkish grammar has the two subtypes *obj-agr* and *non-obj-agr*, each of which further has as its subtypes the kinds of postpositions that do bear agreement morphology and those that don't, respectively, as seen in (3.2).

(3.2)



Turning now to the issue of pro-drop, I do not provide an analysis of that phenomenon in Turkish, that is, I do not account for the interpretation of a dropped complement in such cases. (See Turan (1995) for an analysis of null vs. overt subjects in Turkish discourse using Centering Theory, Grosz *et al.* (1995).) Dini (to appear) presents an HPSG account of null complements in Italian, which relies on an implementation of an event-based theory of 'unselective binding' (Lewis (1975), Kamp (1981), Heim (1982)), in HPSG. His analysis makes use of a 'complement omission lexical rule', which reduces by one the COMPS list of the input verb. The omitted complement is still present in the

ARG-ST list of the output,[1] and is constrained to have a pronominal or arbitrary index. A similar lexical rule in the case of Turkish would presumably apply to nouns, *obj-agr* postpositions, and *non-base* verbs (where *non-base* has the subtypes *fin, nomin, inf, part,* and *ger*).

3.2 Accusative Marked vs Unmarked Objects

We have seen in Chapter 1 (Section 1.3.2) that the interpretation of object NPs in Turkish depends on factors such as the existence of an indefinite determiner and/or accusative case-marking. Explicit case-marking in general correlates with a specific reading of the object (as in (3.3a,b)), whereas the absence of marking may result in either a nonspecific or a nonreferential reading of the object in question (cf. (3.3c,d), respectively), depending on whether or not it occurs with one of the indefinite articles *bir 'a'* and *birkaç 'a few/some'* (Dede (1986), and Enç (1991)).

(3.3) a. Güneş kitab-ı oku-yor. 'Güneş is reading the book.'
Güneş book-ACC read-PROG

b. Güneş bir kitab-ı oku-yor. 'Güneş is reading a (specific) book.'
Güneş a book-ACC read-PROG

c. Güneş bir kitap oku-yor. 'Güneş is reading a book.'
Güneş a book read-PROG

d. Güneş kitap oku-yor. 'Güneş is reading some book/books.'
Güneş book read-PROG

Thus, from a syntactic point of view, accusative case-marking on direct objects seems to be optional in Turkish, meaning that those objects can also occur without any morphological case-marking, i.e. nominative case in Turkish. There are nevertheless certain NPs that are obligatorily case-marked when they appear as direct objects (Nilsson (1985)). Examples are pronouns, proper nouns, possessive NPs, and NPs with any kind of determiner other than the indefinite determiners *bir 'a'* and *birkaç 'a few/some'* . The following sentences exemplify this restriction for possessive NPs (cf. (3.4a,b)), and NPs with the determiners *bu 'this'* and *her 'every'* (cf. (3.4c,d)).

[1]Thus, outputs of that lexical rule, just like outputs of the complement extraction lexical rule (see page 22), violate the canonical constraint on words in the lexicon which requires the values of their valence features to add up to the ARG-ST value; see page 14.

(3.4) a. Güneş çocuğ-un kitab-ı-nı oku-yor.
Güneş child-GEN book-3sPoss-ACC read-PROG
'Güneş is reading the child's book.'

b. * Güneş çocuğ-un kitab-ı oku-yor.
Güneş child-GEN book-3sPoss read-PROG

c. Güneş bu/her kitab-ı oku-yor.
Güneş this/every book-ACC read-PROG
'Güneş is reading this/every book.'

d. * Güneş bu/her kitap oku-yor.
Güneş this/every book read-PROG

To capture this restriction within syntax, I introduce a new HEAD feature called SPECIFICITY appropriate for the types *noun* and *det* only. SPECIFICITY takes values of type *specificity*, which further has the two subtypes *specific* and *nonspecific*.

It should be noted that the only motivation behind introducing the feature SPECIFICITY is to ensure that the 'implicitly specific' NPs mentioned above are always case-marked when they occur as direct objects. In other words, constraining an NP to have a *specific* SPECIFICITY value is meant to prevent the NP from occurring as an unmarked object. On the other hand, assigning an NP a *nonspecific* SPECIFICITY value only means that it *may* occur as an unmarked object, rather than actually imposing any semantic constraints on the 'specificity' of the NP in question.

Common nouns and the indefinite determiners *'bir'* and *'birkaç'* are assumed to be *nonspecific* in the lexicon, whereas all other nouns (e.g. pronouns, proper nouns) and determiners (i.e. universal and definite determiners) are constrained to be *specific*. In addition, possessive suffix affixation to nouns is dealt with by a lexical rule, whose details are given in Section 3.3; one effect of the rule is to change the SPECIFICITY value of the input noun to *specific*.

Note that since SPECIFICITY is a HEAD feature, the HFP will always constrain a head-specifier phrase to inherit the SPECIFICITY value of its head daughter (as part of the whole HEAD value structure-shared with that of the head daughter). However, empirical facts suggest that in certain cases one must let a head-specifier phrase inherit the SPECIFICITY value of its non-head daughter, rather than that of the head daughter.[2] For example, in (3.5a) the phrase indeed inherits the SPECIFICITY value of the head daughter, whereas in (3.5b) it should rather inherit that of the specifier daughter, hence being

[2]That is in fact why SPECIFICITY is defined as an appropriate feature for determiners in the first place.

specific in both cases, since both NPs have to be case-marked as direct objects. In (3.5c), on the other hand, both daughters are *nonspecific*, and hence either could contribute to the SPECIFICITY value of the mother. The NP then would be *nonspecific* in either case, meaning that it could be either case-marked or unmarked as a direct object, in line with the data in (3.3b,c).

(3.5)	a.	bir a [*nonspecific*]	kitab-ım book-1sPoss [*specific*]	'a book of mine'
	b.	bu this [*specific*]	kitap book [*nonspecific*]	'this book'
	c.	bir a [*nonspecific*]	kitap book [*nonspecific*]	'a book'

In other words, in a head-specifier phrase, it is the *specific* daughter, if any, that should contribute to the SPECIFICITY value of the mother, and either daughter could do the job otherwise. There are two ways to get around this apparent problem:

i) One could assume that the HFP, in Turkish grammar, is a default constraint on the type *hd-ph*, which could be overriden on certain more specific subtypes of *hd-ph*, in the present case on *hd-spr-ph*.

ii) One could leave the HFP as a 'hard' constraint as before, but rather make sure that in cases where a head-specifier phrase is to inherit the SPECIFICITY value of its non-head daughter (according to the discussion above, see (3.5b)), that value is actually structure-shared with the SPECIFICITY of the head daughter, and hence gets inherited by the mother as part of the whole HEAD value of the head daughter via the HFP, in the usual way.

Cooper (1986) notes a similar case from Swedish, namely definiteness agreement in Swedish possessive NPs. Consider, for example, the following NPs, which are based on some of the examples provided by Cooper:

(3.6)	a.	den hästen the horse[+def]	'the horse'
	b.	en häst a horse[-def]	'a horse'
	c.	min häst my horse[-def]	'my horse'

Although in (3.6a,b), the definiteness of the NP and the noun *'horse'* are the same, there is a clear mismatch in (3.6c), since the NP is definite, yet the noun has an indefinite form. On the basis of this case (and some other classes of Swedish data, which I won't consider here), Cooper argues that the Head Feature Convention (HFC) of the Generalized Phrase Structure Grammar (GPSG) – the formulation in Gazdar and Pullum (1982) – which is defined as an absolute condition should in fact be considered as a markedness convention that could be overriden by certain rules.[3]

Another related work from the literature is the GPSG analysis of English nominal gerund phrases (e.g. *'(your) having broken the record'*) by Pullum (1991). The analysis exploits the default nature of the HFC in Gazdar *et al.* (1985), and treats such phrases as noun phrases with verb phrase heads by suggesting a disagreement between the head and the mother in the major category features N and V, while they are assumed to agree in respect to all other head features. (Pullum explicitly points out that such an analysis couldn't be reconstructed in HPSG – Pollard and Sag (1987) – because of the absolute nature of the HFP in the theory.)

Finally, one must note that it has previously been argued by Borsley (1993) that the HFP should be defined as a default constraint. However, that argument is based on a somewhat different sort of motivation, namely attributing the sharing of any of the valence requirements of a mother and its head daughter (in cases where they are identical) to the HFP, rather than a separate valence principle.[4]

From the foregoing discussion, one might opt for the first alternative mentioned above, i.e. adopting a default version of the HFP in Turkish grammar. However, I see one rather undesirable consequence of such a move: since we are only interested in overriding the

[3]The definition of HFC in Gazdar and Pullum (1982) is in a way similar to the HFP, in that it requires the head feature specifications of a head to be identical to those of its mother. Note, however, that the HFC was later redefined as a default principle in the formulation of GPSG in Gazdar *et al.* (1985), allowing for head features on a head daughter to differ from those on the mother under certain circumstances. To give the reader the right chronological perspective, one must note that Cooper (1986) was first presented in June 1982 (at the Workshop on Scandinavian Syntax and Theory of Grammar, held in Trondheim, Norway), and hence refers to the formulation of GPSG (and HFC) in Gazdar and Pullum (1982), rather than the one in Gazdar *et al.* (1985).

[4]In fact, Borsley (1993) states that his argument is essentially that the (absolute) HFP is too weak, since it cannot account for the above-mentioned sharing of the valency properties. In addition, he suggests that the HFP may at the same time be regarded as being too strong, since it rules out any mismatch of any of the HEAD values of a mother and its head daughter, citing the GPSG analysis of English nominal gerund phrases by Pullum (1991) (see above). The Turkish data presented here clearly fall within the latter category.

HFP in the case of 'NP' head-specifier phrases, we would need to further introduce two subtypes of *hd-spr-ph*, say *np-hd-spr-ph* (with the HEAD value constrained as *noun*) and *non-np-hd-spr-ph* (with the HEAD value constrained as *non-noun*), and override the HFP only on the former type. I am rather reluctant to introduce those two otherwise unmotivated subtypes of *hd-spr-ph* in Turkish grammar, and hence take here a more conventional line, adopting a solution along the lines of the second alternative mentioned above.

I must first note that it is assumed that the SPR values of all the nouns in the original lexicon are constrained to be empty, since they all may occur as 'bare' NPs, and that a lexical rule is responsible for the inclusion of a DetP within the SPR lists of the nouns. One can then guarantee the proper instantiation of the SPECIFICITY value of a head-specifier phrase in the way discussed above, by constraining that lexical rule in the following way:

(3.7)
$$\begin{bmatrix} \text{HEAD } \textit{noun} \begin{bmatrix} \text{SPECIFICITY } \boxed{1} \end{bmatrix} \\ \text{SPR} \quad \langle\rangle \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{HEAD} \begin{bmatrix} \text{SPECIFICITY } \text{GET-SPECIFIC}(\boxed{1},\boxed{2}) \end{bmatrix} \\ \text{SPR} \quad \left\langle \text{DetP} \begin{bmatrix} \text{SPECIFICITY } \boxed{2} \end{bmatrix} \right\rangle \end{bmatrix}$$

where the function GET-SPECIFIC is defined as:

GET-SPECIFIC($\boxed{1}$, *specificity*) = $\boxed{1}$ if $\boxed{1}$ = *specific*;
GET-SPECIFIC(*specificity*, $\boxed{1}$) = $\boxed{1}$ otherwise

(3.7) basically constrains the SPECIFICITY value of the output noun to be structure-shared with that of the input noun if the input is *specific*, and with that of the newly included *synsem* object in the SPR list otherwise.[5] Thus, for example, the inclusion of a DetP within the SPR list of a noun with a possessive suffix, such as *'kitabım'*, which is constrained as *specific*, would instantiate the lexical rule in (3.7) as in (3.8). Note that the SPECIFICITY value of the output in (3.8) is structure-shared with that of the input, since that value is *specific*.

[5]One may also propose a similar analysis for English nominal gerund phrases, using a lexical rule that applies to *base* verbs affixing an *'-ing'* suffix to the PHON value of the input, and either i) changing the HEAD value to an object of type *noun*, and constraining the subject to have a *genitive* CASE value; or ii) leaving the HEAD value as *verb* (but changing the VFORM value to *gerund*), and constraining the subject to be *nominative*.

(3.8)
$$\begin{bmatrix} \text{HEAD } noun \begin{bmatrix}\text{SPECIFICITY } \boxed{1}\ specific\end{bmatrix} \\ \text{SPR } \langle\rangle \end{bmatrix} \Longrightarrow \begin{bmatrix}\text{HEAD} \begin{bmatrix}\text{SPECIFICITY } \boxed{1}\end{bmatrix}\\ \text{SPR } \langle \text{DetP}\rangle\end{bmatrix}$$

On the other hand, the inclusion of a DetP within the SPR list of an ordinary common noun such as *'kitap'*, which is constrained as *nonspecific*, would instantiate the same lexical rule as in (3.9). Notice that the SPECIFICITY value of the output in this case is essentially left underspecified by being structure-shared with the SPECIFICITY value of the newly included DetP in the SPR list (whatever that value may turn out to be in an actual head-specifier phrase including that output entry).

(3.9)
$$\begin{bmatrix} \text{HEAD } noun \begin{bmatrix}\text{SPECIFICITY } nonspecific\end{bmatrix} \\ \text{SPR } \langle\rangle \end{bmatrix} \Longrightarrow \begin{bmatrix}\text{HEAD} \begin{bmatrix}\text{SPECIFICITY } \boxed{1}\end{bmatrix}\\ \text{SPR } \left\langle \text{DetP} \begin{bmatrix}\text{SPECIFICITY } \boxed{1}\end{bmatrix}\right\rangle\end{bmatrix}$$

These output entries, together with the HFP, make sure that the SPECIFICITY value of the NPs in (3.5), for example, are constrained in the way discussed above. The structures of (3.5a-c) are given below in (3.10a-c), respectively. Notice that in all three cases the HFP constrains the HEAD value of the head-specifier phrase to be structure-shared with that of its head daughter (tag $\boxed{3}$), within which the SPECIFICITY value is constrained as *specific* in (3.10a,b) and as *nonspecific* in (3.10c).

(3.10) a.
$$\begin{bmatrix} \textit{hd-spr-ph} & \\ \text{PHON} & \langle \boxed{1}, \boxed{2} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\ \textit{bir} \rangle \\ \text{SYNSEM} & \boxed{4}\ \text{DetP} \end{bmatrix} \right\rangle \\ \text{HEAD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{2}\ \textit{kitabını} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3}\ noun\ [\text{SPECIFICITY } \textit{specific}] \\ \text{SPR} & \langle \boxed{4} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

b.
$$\begin{bmatrix} \textit{hd-spr-ph} & \\ \text{PHON} & \langle \boxed{1}, \boxed{2} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\ \textit{bu} \rangle \\ \text{SYNSEM} & \boxed{4}\ \text{DetP}\ [\text{SPECIFICITY } \boxed{5}\ \textit{specific}] \end{bmatrix} \right\rangle \\ \text{HEAD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{2}\ \textit{kitap} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3}\ noun\ [\text{SPECIFICITY } \boxed{5}] \\ \text{SPR} & \langle \boxed{4} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

c. $$\begin{bmatrix} \textit{hd-spr-ph} & \\ \text{PHON} & \langle \boxed{1}, \boxed{2} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\, \textit{bir} \rangle \\ \text{SYNSEM} & \boxed{4}\ \text{DetP}\ [\text{SPECIFICITY}\ \boxed{5}\ \textit{nonspecific}] \end{bmatrix} \right\rangle \\ \text{HEAD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{2}\, \textit{kitap} \rangle \\ \text{SYNSEM|CAT} & \begin{bmatrix} \text{HEAD} & \boxed{3}\ \textit{noun}\ [\text{SPECIFICITY}\ \boxed{5}] \\ \text{SPR} & \langle \boxed{4} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Returning to the issue of optionality of accusative case-marking on direct objects, one can then view this optionality as a lexical redundancy, proposing the use of the following lexical rule to deal with it.

(3.11) $$\begin{bmatrix} \text{HEAD} & \textit{verb}\begin{bmatrix}\text{VFORM}\ \textit{bse}\end{bmatrix} \\ \text{COMPS} & \left\langle ..., \text{NP}_{\boxed{1}}[\textit{acc}], ... \right\rangle \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} \left\langle ..., \text{NP}_{\boxed{1}}[\textit{nom}, \textit{nonspecific}], ... \right\rangle \end{bmatrix}$$

(3.11) simply replaces the *accusative* NP complement of a *base* verb by a *nominative* NP, whose SPECIFICITY value is constrained as *nonspecific* and whose INDEX is structure-shared with that of the original *accusative* complement, to ensure that the *nominative* NP gets assigned the appropriate semantic role within the verb's CONTENT value.

3.3 Possessive NPs

Possessors in Turkish are genitive marked and agree with the possessive suffix on the possessed noun in person and number, as shown in (3.12).

(3.12)	a.	ben-im kitab-ım I-GEN book-1sPoss	'my book'
	b.	ben-im her kitab-ım I-GEN every book-1sPoss	'every book of mine'
	c.	çocuğ-un kitab-ı child-GEN book-3sPoss	'the child's book'
	d.	çocuğ-un bir kitab-ı child-GEN a book-3sPoss	'a book of the child's'

Pollard and Sag (1994) claim that in languages like English and German, possessors are more likely to be treated as specifiers whereas in some other languages such as Welsh and Hungarian, it may well be the case that they are subjects (pp. 374-75). I argue that there are good reasons to analyze possessors as subjects in Turkish as well. First, the possessor-possessed noun agreement in Turkish follows exactly the same morphological pattern observed in subject-verb agreement in non-finite Turkish sentences with genitive marked subjects. Second, and perhaps more importantly, just like those genitive subjects, and unlike specifiers, possessors can both be relativized and extracted out of the possessive phrases they occur in.[6] (3.13b), for instance, exemplifies the relativization of the genitive possessor *'çocuğun'* out of the possessive NP in (3.13a), and (3.13c) exemplifies the extraction of that possessor, which then occurs in the postverbal position of the main clause.

(3.13) a. [$_{\text{NP}}$ Çocuğ-un kitab-ı] raf-tan düş-tü.
child-GEN book-3sPoss shelf-ABL fall down-PAST
'The child's book fell down from the shelf.'

b. [$_{\text{S}[part]}$ [$_{\text{NP}}$ $-_i$ kitab-ı] raf-tan düş-en] çocuk$_i$
book-3sPoss shelf-ABL fall down-PART child
'the child whose book fell down from the shelf'

c. [$_{\text{S}[fin]}$ [$_{\text{NP}}$ $-_i$ kitab-ı] raf-tan düş-tü] çocuğ-un$_i$.
book-3sPoss shelf-ABL fall down-PAST child-GEN
'His, the child's, book fell down from the shelf.'

I therefore propose a subject analysis of possessors in Turkish that makes use of the lexical rule in (3.14), which basically deals with possessive suffix affixation to nouns.

[6] Note also that in the case of NPs with a possessor and the reflexive pronoun (specifier) *kendi 'own'*, as in (i), the reflexive pronoun is always coindexed with the possessor, in line with the Principle A of the Binding Theory (which requires a locally o-commanded anaphor to be locally o-bound; cf. Pollard and Sag (1994)[Chapter 6]):

(i) Mehmet$_i$ [$_{\text{NP}}$ Berfu'nun$_j$ kendi$_{j/*i}$ resm-i-ni] gör-dü.
Mehmet Berfu-GEN own picture-3sPoss-ACC see-PAST
'Mehmet saw Berfu's own picture.'

(3.14)
$$\left[\begin{array}{ll} \text{PHON} & \boxed{1} \\ \text{HEAD} & noun\left[\begin{array}{l}\text{CASE } nom\end{array}\right] \\ \text{SPR} & \langle\rangle \\ \text{SUBJ} & \langle\rangle \\ \text{CONTENT|RESTR} & \boxed{3}\end{array}\right] \Longrightarrow \left[\begin{array}{ll} \text{PHON} & \text{POSSESSED-FORM}(\boxed{1},\boxed{4}) \\ \text{HEAD} & \left[\begin{array}{l}\text{SPECIFICITY } specific\end{array}\right] \\ \text{SUBJ} & \left\langle \text{NP}_{\boxed{4}}[gen]\right\rangle \\ \text{CONTENT} & \left[\begin{array}{ll}\text{INDEX} & \boxed{2} \\ \text{RESTR} & \left\{\left[\begin{array}{l} possess \\ \text{POSSESSOR } \boxed{4} \\ \text{POSSESSED } \boxed{2}\end{array}\right]\right\} \cup \boxed{3}\end{array}\right]\end{array}\right]$$

(3.14) includes a *genitive* NP argument (i.e. the possessor) in the SUBJ list of the output, as well as constraining the SPECIFICITY value of the output as *specific* (following the discussion in Section 3.2). The function POSSESSED-FORM in (3.14) is assumed to return the PHON value of the output noun after the possessive marker affixation, depending on the PHON value of the input noun (tag $\boxed{1}$) and the agreement of the possessor (tag $\boxed{4}$). Since possessive markers precede case markers in Turkish, the input of (3.14) is restricted to be unmarked with respect to case (i.e. have *nominative* case). And finally, a new psoa *possess* is included within the CONTENT|RESTR set of the output (in addition to any restrictions already imposed on the input entry, tag $\boxed{3}$), with the POSSESSOR value structure-shared with the INDEX of the possessor (tag $\boxed{4}$) and the POSSESSED value with that of the noun itself (tag $\boxed{2}$).

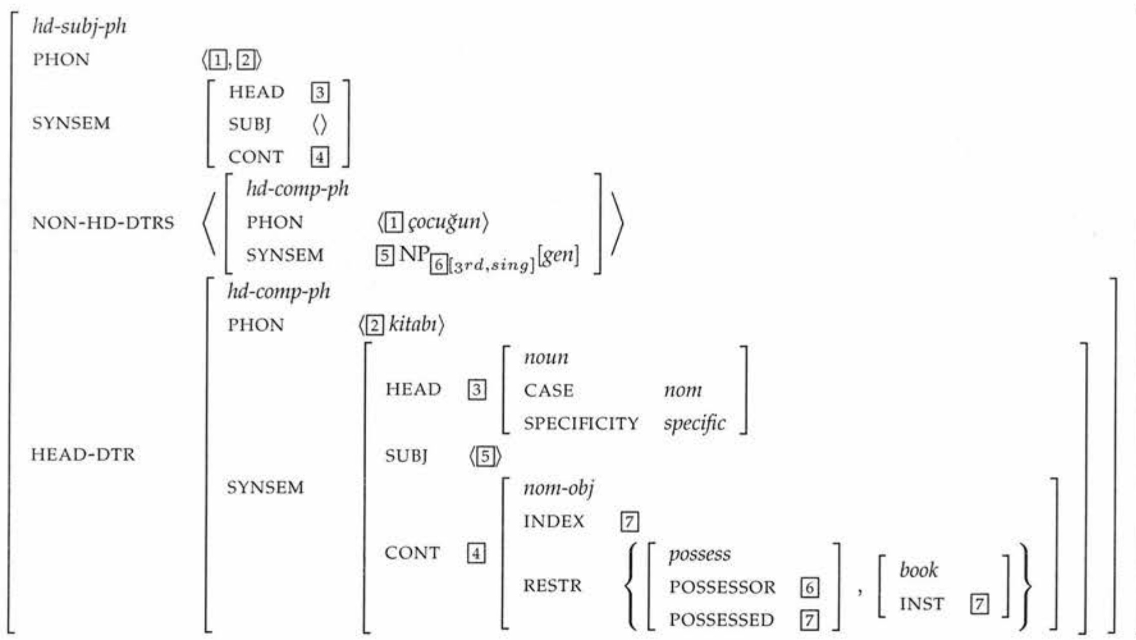
Example (3.15) illustrates the instantiation of the above lexical rule for the third person singular possessive suffix affixation to the noun *'kitap'*, resulting in the surface form *'kitabı'*.

(3.15)

$$\left[\begin{array}{ll} \text{PHON} & \langle kitap\rangle \\ \text{SUBJ} & \langle\rangle \\ \text{CONTENT} & \left[\begin{array}{ll}\text{INDEX} & \boxed{1} \\ \text{RESTR} & \left\{\left[\begin{array}{l} book \\ \text{INSTANCE } \boxed{1}\end{array}\right]\right\}\end{array}\right]\end{array}\right] \Longrightarrow \left[\begin{array}{ll} \text{PHON} & \langle kitabı\rangle \\ \text{HEAD|SPECIFICITY} & specific \\ \text{SUBJ} & \left\langle \text{NP}_{\boxed{2}[3rd,sing]}[gen]\right\rangle \\ \text{CONTENT} & \left[\begin{array}{ll}\text{INDEX} & \boxed{1} \\ \text{RESTR} & \left\{\left[\begin{array}{l} possess \\ \text{POSSESSOR } \boxed{2} \\ \text{POSSESSED } \boxed{1}\end{array}\right], \left[\begin{array}{l} book \\ \text{INSTANCE } \boxed{1}\end{array}\right]\right\}\end{array}\right]\end{array}\right]$$

The output entry in (3.15) would then license the structure of a possessive NP, such as (3.12c) above, in the following way:

(3.16)



Note that the structure-sharing of the HEAD value of the head-subject phrase in (3.16) and that of its head daughter (tag 3) is imposed by the HFP, and the structure-sharing of their CONTENT values (tag 4) is due to the relevant constraint on the type *hd-nexus-ph* (inherited by all its subtypes, including *hd-subj-ph*). Finally, the relevant constraints on the type *hd-subj-ph* itself require the non-head daughter of the head-subject phrase to structure-share its SYNSEM value with the *synsem* object in the SUBJ list of the head daughter of the phrase (tag 5), and further make sure that the SUBJ value of the mother is constrained as empty.

3.4 Word Order

The typical order of constituents in Turkish sentences is subject-object-verb (SOV). Yet this order can, in general, change rather freely, in some cases all possible permutations of constituents in a sentence being grammatical. This is due to the fact that the highly inflectional morphology of Turkish enables morphological markings on the constituents to signal their grammatical roles without relying on the word order. For example, (3.17a) is a simple transitive sentence in the typical SOV order, whose constituents can in fact appear in any possible order, as seen in (3.17b-f).

(3.17) a. Berfu kitab-ı oku-yor. 'Berfu is reading the book.'
Berfu book-ACC read-PROG

b. Kitabı Berfu okuyor.

c. Berfu okuyor kitabı.

d. Kitabı okuyor Berfu.

e. Okuyor kitabı Berfu.

f. Okuyor Berfu kitabı.

It is important, however, to note that the different orders in (3.17a-f) are in general not interchangeable in a given discourse situation. For instance, (3.17b) would be a felicitous answer for a question like (3.18a), but would sound rather awkward as an answer for (3.18b).

(3.18) a. Kitab-ı kim oku-yor? 'Who is reading the book?'
book-ACC who read-PROG

b. Berfu ne oku-yor? 'What is Berfu reading?'
Berfu what read-PROG

Erguvanlı (1984) argues that three different syntactic positions in Turkish sentences are associated with three different pragmatic functions in Turkish, in turn triggering different word orders. The sentence-initial position is associated with the 'topic', meaning that Turkish speakers place the information which links the sentence to the previous context in that position. The immediately preverbal position is associated with the 'focus', hence attracting the most 'information bearing' element (Vallduví (1992)) in a particular context. And finally, the post-predicate position is associated with the 'background', that is, the information that is in general discourse-predictable or recoverable. This pragmatics-oriented aspect of the meaning of a sentence constitutes its 'information structure'. Note that Hoffman (1995) defines a somewhat different characterization of the information structure of Turkish sentences, which she then uses in developing an integrated Turkish grammar for syntactic and pragmatic competence within the framework of Multiset-CCG, an extension of Combinatory Categorial Grammars (Ades and Steedman (1982)), developed by Hoffman. In her account, each Turkish sentence is divided into a 'topic' component ("the main element that the sentence is about"), and a 'comment' component ("the main information the speaker wants to convey about the topic"). The latter is further divided into the 'focus' ("the most information bearing constituent in the sentence") and 'ground' (the information that helps one to "ground the sentence in the current context"). In any case, there is a clear consensus that in Turkish the sentence-initial position is associated with the topic, and the immediately preverbal position with the focus of the sentence.

One can then explain, for example, the felicity/infelicity of (3.17b) as an answer for (3.18a,b), respectively, by the fact that the constituent that occurs in the immediately preverbal (focus) position in (3.17b), *'Berfu'*, is indeed the most information bearing consituent in the context of (3.18a), that is, the answer sought for by that wh-question, but not the one in the context of (3.18b) (which is in fact the sentence-initial *'kitabı'* in (3.17b)).

In this dissertation, I only concern myself with syntactic aspects of word order variation in Turkish. In other words, I do not provide an account of the differences in the contextual constraints conveyed by different word orders, hence providing an analysis of the information structure of Turkish sentences. In fact, in addition to word order variation, prosody too significantly interacts with the information structure in Turkish (cf. Erguvanlı (1984)[Chapter 4], and Vallduví and Engdahl (1996)). Thus, a complete analysis of the information structure of Turkish sentences should account for the interaction between the prosodic, syntactic, semantic and contextual constraints in the grammar. This suggests that HPSG is a framework well-suited for the integration of information structure and grammar, since its highly integrated architecture enables one to express such integration in an optimal way (Vallduví and Engdahl (1995)). Vallduví and Engdahl (1995) propose a representation of information structure in the grammar in the framework of HPSG, and use that representation to account for the focus-ground articulation in two langauges, namely English and Catalan, which employ rather different means for this kind of articulation, i.e. prosody and word order variation, respectively. More recent work by Kuhn (1996) proposes an underspecified representation for information structure to deal with 'focus ambiguity' (on the basis of prosodic marking of a sentence) together with a resolution routine based on centextual information, and provides a set of HPSG principles that build up this representation for German.

In the rest of this section, I focus on the range of word order variation in simple and complex Turkish sentences, and discuss a number of syntactic constraints that restrict this variation to a certain extent. I also detail the assumptions I make in this dissertation to deal with "free" word order in Turkish and to capture the restrictions on it, within the HPSG framework.

3.4.1 Word Order in Simple Sentences

As seen in (3.17) above, subjects and objects in Turkish sentences can scramble rather freely, as far as the syntax is concerned. In order to deal with this "free" nature of word order in Turkish, I assume a flat structure for Turkish sentences, in which all complements (including subjects) are treated in the same way. To that end, I make use of a lexical rule, given in (3.19), that applies to *non-base-inf* verbs (with subtypes *fin*, *nomin*, *part* and *ger*). (• stands for list concatenation.)

$$(3.19)\quad \begin{bmatrix} \text{HEAD} & \textit{verb}\begin{bmatrix}\text{VFORM } \textit{non-base-inf}\end{bmatrix} \\ \text{SUBJ} & \langle \boxed{1} \rangle \\ \text{COMPS} & \boxed{2} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{SUBJ} & \langle\rangle \\ \text{COMPS} & \langle \boxed{1} \rangle \bullet \boxed{2} \end{bmatrix}$$

(3.19) removes the only element in the SUBJ list of the input entry, placing it within the COMPS list of the output verb, thereby allowing that verb to select its subject via the COMPS feature, rather than SUBJ.[7] Consequently, Turkish sentences are considered instances of *hd-comp-ph*. Note, however, that the SUBJ-COMPS distinction in the grammar is still essential for the analysis of relative clauses presented in Chapter 4.

In addition, I make a distinction between pre-predicate and post-predicate scrambling, treating sentences with pre-predicate scrambling as instances of *hd-comp-ph*, while considering the ones with post-predicate scrambling as instances of *hd-fill-ph*. The motivation for this assumption comes from the fact that, as far as the other phrase kinds such as NPs and PPs are concerned, Turkish is mainly a head-final language.[8] Thus, for example, it employs postpositions, rather than prepositions, and adjuncts (including relative clauses) and specifiers always precede their nominal heads. Moreover, certain kinds of *strictly* head-final verbal phrases, namely sentences headed by participles (i.e. relative clauses) and infinitive phrases, provide further support for this assumption, by letting their constituents freely scramble in the preverbal position.

The treatment of post-predicate scrambling as resulting in head-filler phrases resembles, to a certain extent, the Lexical Functional Grammar (LFG, Kaplan and Bresnan (1982)) analysis proposed by Mohanan (1982) for Malayalam, which exhibits properties similar

[7] A similar lexical rule has been proposed by Sag (to appear) that applies to finite auxiliary verbs, to licence English inverted clauses.

[8] An additional *computational* motivation for this assumption is discussed in Chapter 7 (Section 7.1), which relates to a considerable improvement in parsing efficiency in terms of incremental parsing, due to the elimination of spurious partial solutions that would otherwise arise.

to Turkish in terms of word order. He assumes that Malayalam is a verb-final language, providing a phrase structure rule for only verb-final sentence structures, and deals with non-verb-final sentences using a stylistic scrambling rule which operates on that phrase structure rule.

Sentential adjuncts in Turkish can scramble in the same way (and for the same pragmatic reasons) as arguments. For example, the locative adjunct NP *evde 'at home'* (or the temporal adjunct *şimdi 'now'*) in (3.20) can occur in different positions as in (3.20a-d).[9] (Small capitals are henceforth used in English translations to indicate focal stress.)

(3.20) a. Berfu ev-de/şimdi kitab-ı oku-yor.
Berfu home-LOC/now book-ACC read-PROG
'Berfu is reading the book at home/now.'

b. Evde/şimdi Berfu kitabı okuyor.
'At home/now, Berfu is reading the book.'

c. Berfu kitabı evde/şimdi okuyor.
'Berfu is reading the book at HOME/NOW.'

d. Berfu kitabı okuyor evde/şimdi.
'Berfu is reading the book, at home/now.'

Note that scrambling of sentential adjuncts in this way constitutes a major problem for the main tradition HPSG analysis of sentence structure, since lexical heads are expected to combine with all their (non-subject) complements at once in a head-complement phrase.

Scrambling of adjuncts also occurs in certain other languages, among them German, for which Kasper (1994) outlines three different alternatives to capture the phenomenon within HPSG: i) allowing head-complement phrases where adjunct daughters can be sisters of complement daughters, as suggested by Pollard and Sag (1987), ii) assuming binary-branching structures with partial verb phrases, where a head combines with its complements one at a time (cf. the GPSG account of partial VP fronting by Nerbonne (1983)), or a variation of this approach where a head may combine with any number of its complements at a time (Pollard (1990)); and iii) treating such sentences as discontinuous realizations of a head-complement phrase embedded within a head-adjunct phrase, following a theory of semi-free word order by Reape (1990). Kasper discusses the strong and weak points of all three alternatives, and then proposes an analysis in

[9]In fact, all 24 permutations of the four constituents in this case are grammatical.

terms of flat phrase structures with adjuncts and complements as sisters (as in (i) above), which relies on a number of technical mechanisms that guarantee the appropriate semantic composition from phrase structure configurations. The work presented here neither attempts to explore the applicability of that approach (or any of the other alternatives) in the case of Turkish, nor does it attempt to propose an alternative analysis for adverbial modification in Turkish, leaving the issue for further consideration.

Syntactic Constraints

Having presented the above examples that suggest a "free" constituent order nature for simple Turkish sentences, it is now time to point out a number of syntactic constraints that restrict this free nature to a certain degree, and happen to be useful in eliminating potential ambiguities in certain cases.

One such constraint is related to the existence of morphological case-marking on direct objects. Recall from Section 3.2 that accusative case-marking on direct objects is optional in Turkish, and that explicit case-marking, in general, correlates with a specific reading of the object (cf. Dede (1986), Enç (1991)). The constraint is that unmarked (nominative) direct objects can only appear in the immediately preverbal position in a sentence, which, for instance, determines that *'mutluluk'* is the subject and *'huzur'* is the object in (3.21), and rules out the other interpretation. (This example is taken from Erguvanlı (1984).)

(3.21) Mutluluk huzur getir-ir.
happiness peace of mind bring-PRES
'Happiness brings peace of mind.'
*'Peace of mind brings happiness.'

Hoffman (1995) notes the following example, which shows that unmarked *indefinite* NP objects in 'contrastive gapping constructions' are allowed to scramble, as seen in (3.22a), unlike *bare* unmarked objects in similar contexts, as seen in (3.22b). (Again, (3.22a) is originally taken from Erguvanlı (1984).)

(3.22) a. Bir gömlek san-a, bir gömlek de kardeş-in-e al-dı-m.
a shirt you-DAT a shirt too sibling-2sPoss-DAT buy-PAST-1SG
'(I) bought a shirt for you and a/another shirt for your sibling.'

b. * Gömlek san-a, gömlek de kardeş-in-e al-dı-m.
shirt you-DAT shirt too sibling-2sPoss-DAT buy-PAST-1SG
'(I) bought some shirt(s) for you and some shirt(s) for your sibling.'

However, in most contexts unmarked object NPs are not allowed to scramble (Erguvanlı (1984) and Hoffman (1995)).

Another constraint is that 'non-derived' manner adverbs[10] always immediately precede the verb or, if it exists, the nominative object (Erguvanlı (1984)[pages 192–196]). Thus, for example, *'iyi'* in (3.23a) can only be interpreted as an adjective that modifies the accusative object *'yemeği'*, and in (3.23b) it is an adverb modifying the verb *'pişirdin'*. In (3.23c), on the other hand, both interpretations are possible.

(3.23) a. Berfu iyi yemeğ-i pişir-di.
Berfu good meal-ACC cook-PAST
'Berfu cooked the good meal.'
*'Berfu cooked the meal well.'

b. Berfu yemeğ-i iyi pişir-di.
Berfu meal-ACC well cook-PAST
'Berfu cooked the meal well.'

c. Berfu iyi yemek pişir-di.
Berfu good/well meal cook-PAST
'Berfu cooked some good meal.'
'Berfu cooked well.'

Indefinite NPs

It has also been suggested that Turkish exhibits additional constraints on word order that render sentences that begin with indefinite NPs ungrammatical (Erguvanlı (1984), Dede (1986), Tura (1986)). Erguvanlı (1987) attempts to provide a characterization of the cases that allow/disallow indefinite NPs to occur sentence-initially, depending on factors such as the animacy of the NP in question, and the definiteness and animacy of the other NPs in the sentence, if any. To that end, she claims that indefinite subjects of intransitive verbs can occur in the sentence-initial position when, and only when, they are animate, relying on examples such as (3.24) and (3.25). In (3.24), although the unmarked position of the animate, indefinite subject NP *'bir adam'* is the immediately preverbal position, as in (3.24a), that NP can still occur in the sentence-initial position, as seen in (3.24b); yet this is not possible for the inanimate, indefinite subject *'bir paket'* in (3.25).

[10]The term 'non-derived' in this context refers to the fact that these adverbs have not undergone any of the adverb derivation processes in Turkish, such as re-duplication, suffixation (e.g. of the suffixes '-cE', '-lE', '-leyin', etc.), or a combination of these two processes (Erguvanlı (1984)[pages 183–186]). They are in fact qualitative adjectives, but can also be used as adverbs. Examples are *iyi 'good/well'*, *hızlı 'fast'*, *güzel 'beautiful/beautifully'*.

(3.24) a. Kapı-nın ön-ü-nde *bir adam* dur-uyor.
door-GEN front-3SP-LOC a man stand-PROG

b. *Bir adam* kapının önünde duruyor.
'A man is/has been standing in front of the door.'

(3.25) a. Masa-nın üst-ü-nde *bir paket* dur-uyor.
table-GEN top-3SP-LOC a packet lie-PROG
'There is a packet on the table.' (Lit. 'A packet is lying on the table.')

b. **Bir paket* masanın üstünde duruyor.

However, Hoffman (1995) points out that inanimate, indefinite NPs *can* occur sentence-initially if they refer to specific discourse entities, as exemplified in (3.26), where the inanimate, indefinite subject NP *'bir kitap'* has been made more specific by the modifier *'mavi kaplı'*, and hence can occur sentence-initially.

(3.26) *Mavi kap-lı bir kitap* masa-nın üst-ü-nde dur-uyor.
blue cover-with a book table-GEN top-3sPoss-LOC lie-PROG
'A blue covered book is lying on the table.'

Moreover, consider, for example, (3.27b), where the indefinite subject, although animate, sounds quite odd in the sentence-initial position.

(3.27) a. Masa-nın üst-ü-nde *bir sinek* dur-uyor.
table-GEN top-3SP-LOC a fly lie-PROG
'There is a fly on the table.'

b. **Bir sinek* masanın üstünde duruyor.

Erguvanlı (1987) also claims that indefinite and inanimate subjects of transitive verbs are restricted to the immediately preverbal position when the direct object is definite and animate (cf. the indefinite subject *'bir paket'* in (3.28), but free in their order otherwise (cf. the indefinite subject *'bir araba'* in (3.29)).

(3.28) a. Sen-i ev-de *bir paket* bekli-yor.
you-ACC home-LOC a packet wait for-PROG
'A packet is waiting for you at home.'

b. **Bir paket* seni evde bekliyor.

(3.29) a. Yol-u *bir araba* tıka-mış.
road-ACC a car block-NARR

b. *Bir araba* yolu tıkamış.
'A car has blocked the road.'

However, this statement incorrectly rules out (3.30b), with the non-preverbal indefinite, inanimate subject NP *'bir araba'/'bir yıldız'*.

(3.30) a. Biz-i *bir araba / yıldız* takip ed-iyor-du.
we-ACC a car / star follow-PROG-PAST

b. *Bir araba/yıldız* bizi takip ediyordu.
'A car / star was following us.'

To sum up, I agree with Hoffman (1995) that the tendency of the indefinite NPs in Turkish to occur in the immediately preverbal position (rather than the other sentence positions) is not a syntactic restriction on the word order, but is rather a consequence of the information structure in Turkish, which places the topical information in the sentence-initial position and the focal information in the immediately preverbal position. I therefore suggest that this tendency should be captured by contextual constraints on the information structure of sentences which determine the topicability of certain discourse entities, rather than being captured within syntax by simple features such as animacy. One could then account for the contrasts in the ordering of the indefinite NPs in the sentences we have seen above, by using pragmatic notions (in addition to syntactic ones) in the linear precedence (LP) constraints for Turkish. (At the end of this chapter, Section 3.4.6, I propose a set of LP constraints for Turkish along these lines, which is further independently motivated by certain other word order facts in Turkish, as I mention in the next section.)

Word Order Constraints in HPSG

In the light of the discussion so far, one can assume the following LP constraints for Turkish.

Head-final Constraint

Turkish is a head-final language, employing the following LP constraint.

(3.31) [ ] < HEAD

(3.31) simply states that in any phrasal sign of Turkish, the head follows any of its sisters. (Note that head-filler phrases that are instances of backgrounding is an exception for this generalization. See Section 3.4.6 for a discussion of this exception, which suggests the use of pragmatic notions in the LP constraints for Turkish, as also motivated above by the discussion on indefinite NPs.)

Nominative Objects

Nominative objects can only occur in the immediately preverbal position, hence should be preceded by any other complement. Thus, I assume the following LP constraint in Turkish, which, together with (3.31), guarantees that nominative objects always *immediately* precede the head.

(3.32) COMPLEMENT < COMPLEMENT[*nom*, NON-SUBJECT]

(3.32) states that in any phrasal sign of Turkish, the nominative, non-subject complement, if any, is constrained to be preceded by any other sister complements. Note that a nominative complement is a nominative object (that is, has the property of being NON-SUBJECT) if and only if its SYNSEM value is structure-shared with the second element in the ARG-ST list of the head. (3.32) therefore makes essential use of the ARG-ST list of the head daughter of the phrasal sign to which it applies (just like the LP constraint (2.29) for English in Chapter 2, which constrains the relative linear order of any two complement sisters in a phrasal sign of English, according to their relative degree of obliqueness[11]).

3.4.2 Word Order in Complex Sentences

Complex sentences in Turkish can have embedded clauses of kind non-finite (i.e. headed by a verb that has a VFORM value of type *nominalization (act* or *fact), infinitive, participle* or *gerund*) or *finite*, the latter being further broken down into *marked* (by either of the markers *ki 'that'* and *çünkü 'because'*) and *unmarked*.

(3.33) provides examples for complex sentences containing non-finite clauses with all the constituents (in both embedded and matrix clauses) in their unmarked positions.

[11] This analogy was first pointed out to me by Ewan Klein.

(3.33a) is a complex sentence with an S[*fact*] complement. (3.33b) is one with an unsaturated VP[*inf*] complement.[12] (3.33d) contains an accusative NP complement modified by an S[*part*] (i.e. a relative clause). And (3.33d) is an example with an S[*ger*] sentential modifier.

(3.33) a. Berfu ban-a [$_{S[fact]}$ Mehmet'-in Ankara'-dan dön-düğ-ü-nü]
Berfu I-DAT Mehmet-GEN Ankara-ABL return-FACT-3sPoss-ACC
söyle-di.
tell-PAST
'Berfu told me that Mehmet has returned from Ankara.'

b. Berfu [$_{VP[inf]}$ Mehmet'-i Ankara'-ya götür-mek] iste-di.
Berfu Mehmet-ACC Ankara-DAT take-INF want-PAST
'Berfu wanted to take Mehmet to Ankara.'

c. Berfu [$_{S[part]}$ Mehmet'-in $\_{i}$ oku-duğ-u] kitab-ı$_i$ bil-iyor.
Berfu Mehmet-GEN read-PART-3sPoss book-ACC know-PROG
'Berfu knows the book that Mehmet is reading.'

d. Berfu [$_{S[ger]}$ Mehmet Ankara'-ya gid-ince] ben-i ara-dı.
Berfu Mehmet Ankara-DAT go-GER I-ACC phone-PAST
'Berfu phoned me when Mehmet went to Ankara.'

Marked finite clauses (idiosyncratically) follow the main verb in the typical order, as seen in (3.34). This behaviour, which is at odds with the typical head-final word order in Turkish, has been explained by the Persian origin of the markers *ki* and *çünkü* (Erguvanlı (1984)).

(3.34) Berfu ban-a de-di [$_{S[ki,fin]}$ ki [$_{S[unmarked,fin]}$ Mehmet Ankara'-dan
Berfu I-DAT tell-PAST that Mehmet Ankara-ABL
dön-dü.]]
return-PAST
'Berfu told me that Mehmet has returned from Ankara.'

[12]The unexpressed subject of the VP[*inf*] complement is constrained to be coindexed (i.e. structure-share its INDEX value) with the subject of the matrix clause via a constraint imposed by the lexical entry of the matrix verb *iste 'want'*. See Pollard and Sag (1994)[Chapters 4,7] for details of the HPSG analysis of unsaturated complements and complement control.

Finally, unmarked finite clauses occur as complements of only a small number of verbs, namely *san- 'assume/think'*, *zannet- 'assume/think'* and *tahmin et- 'guess'*, as seen in (3.35). (Note that the verbal complement in (3.35b) is an unsaturated one.[13])

(3.35) a. Ben [$_{S[unmarked,fin]}$ Mehmet Ankara'-dan dön-dü] san-dı-m/zannet-ti-m.
I Mehmet Ankara-ABL return-PAST think-PAST-1SG

b. Ben Mehmet-i [$_{VP[fin,unmarked]}$ Ankara'-dan dön-dü]
I Mehmet-ACC Ankara-ABL return-PAST
san-dı-m/zannet-ti-m.
think-PAST-1SG
'I thought that Mehmet has returned from Ankara.'

In the rest of this section, I investigate the word order variation in complex sentences. In Section 3.4.1, we have seen that scrambling of indefinite NPs in Turkish may be subject to certain restrictions, as a consequence of certain facts related to the information structure of Turkish sentences (Hoffman (1995)). Thus, in order to work out *purely* syntactic constraints on scrambling, I systematically use definite NPs in the examples to come.

Local Scrambling

In the context of complex sentences 'local scrambling' (i.e. scrambling of constituents within a clause, as before) may refer to both the scrambling of an embedded clause in the matrix sentence, and the scrambling of the constituents of an embedded clause within the boundaries of that clause.

[13]The unexpressed subject of the VP[*fin*] complement in this case is constrained to structure-share its SYNSEM value with the accusative object, *'Mehmet'i'*, of the matrix clause, via a constraint imposed by the lexical entry of the matrix verb *'san/zannet'*. In addition, it is assumed that the matrix verb doesn't assign a semantic role to its accusative object. Again, see Pollard and Sag (1994)[Chapters 4,7] for details of the HPSG analysis of unsaturated complements and complement control.

Local Scrambling of Embedded Clauses in the Matrix Sentence

Non-finite clauses, with the exception of object infinitive clauses, can occur anywhere in the matrix sentence just like any other constituent. In (3.36a), for example, the accusative S[*fact*] complement of the main verb *'söyledi'* occurs in the sentence-initial position, rather than the typical immediately preverbal position as in (3.33a). In (3.36b), on the other hand, the S[*ger*] sentential modifier occurs in the postverbal position of the main sentence (cf. the unmarked order in (3.33d)).[14]

(3.36) a. [Mehmet'-in Ankara'-dan dön-düğ-ü-nü] Berfu ban-a söyle-di.
Mehmet-GEN Ankara-ABL return-FACT-3sPoss-ACC Berfu I-DAT tell-PAST
'That Mehmet has returned from Ankara, Berfu told me.'

b. Berfu $\_\__i$ ben-i ara-dı [Mehmet Ankara'-ya gid-ince.]$_i$
Berfu I-ACC phone-PAST Mehmet Ankara-DAT go-GER
'Berfu phoned me when Mehmet went to Ankara.'

It should be noted that subject and object infinitive clauses exhibit different behaviour with respect to this kind of scrambling. Subject infinitive clauses can scramble in the matrix sentence just like other non-finite clauses, as seen in (3.37b,c), where the VP[*inf*] subject of (3.37a) occurs in the immediately preverbal position and the postverbal position, respectively.

(3.37) a. [$_{VP[inf]}$ Çocuğ-a bak-mak] kadın-ı yor-du.
child-DAT look after-INF woman-ACC make tired-PAST
'To look after the child has made the woman tired.'

b. Kadın-ı [çocuğ-a bak-mak] yor-du.
woman-ACC child-DAT look after-INF make tired-PAST
'As for the woman, to look after the child has made her tired.'

c. [$\_\__i$ Kadın-ı yor-du] [çocuğ-a bak-mak]$_i$.
woman-ACC make tired-PAST child-DAT look after-INF
'It has made the woman tired, to look after the child.'

Object infinitive clauses, on the other hand, are restricted to their unmarked immediately preverbal position, as in (3.38a), ruling out any scrambling, as seen in (3.38b,c).

[14] Recall that examples of post-predicate scrambling are assumed to be instances of *hd-fill-ph*. Accordingly, the unmarked position of the postverbal constituent in (3.36b) (and in the examples to come) is marked by an underscore, indicating a gap, linked to the constituent itself.

(3.38) a. Berfu [$_{\mathsf{S}[\mathit{inf}]}$ Mehmet'-i Ankara'-ya götür-mek] iste-di.
Berfu Mehmet-ACC Ankara-DAT take-INF want-PAST
'Berfu wanted to take Mehmet to Ankara.'

b. * [Mehmet'-i Ankara'-ya götür-mek] Berfu iste-di.
Mehmet-ACC Ankara-DAT take-INF Berfu want-PAST
'To take Mehmet to Ankara, Berfu wanted.'

c. * [Berfu $\_{i}$ iste-di] [Mehmet'-i Ankara'-ya götür-mek.]$_i$
Berfu want-PAST Mehmet-ACC Ankara-DAT take-INF
'Berfu wanted it, to take Mehmet to Ankara.'

This restriction on the scrambling of object infinitive clauses in fact resembles the restriction on the scrambling of nominative direct objects (see Section 3.4.1). Considering these two cases, one may argue that Turkish exhibits a restriction on scrambling of object complements unless they are case-marked.

Turning now to finite complement clauses, those can only occur in their unmarked positions in the matrix sentence. Thus, for example, the S[*ki*] complement in (3.34) cannot occur in any position other than the typical immediately postverbal position, ruling out (3.39a,b) below. Similarly, the S[*unmarked*] complement in (3.35a) cannot leave the immediately preverbal position, and occur elsewhere, as seen in (3.39c,d). (Note that this restriction on scrambling of S[*fin*] complements is again in line with the above observation that object complements are not allowed to leave their typical positions in Turkish sentences unless they are case-marked.)

(3.39) a. * [Ki [Mehmet Ankara'-dan dön-dü]] Berfu ban-a de-di.
that Mehmet Ankara-ABL return-PAST Berfu I-DAT tell-PAST
'That Mehmet has returned from Ankara, Berfu told me.'

b. * Berfu ban-a [ki [Mehmet Ankara'-dan dön-dü]] de-di.
Berfu I-DAT that Mehmet Ankara-ABL return-PAST tell-PAST
'Berfu told me that Mehmet has returned from Ankara.'

c. * [Mehmet Ankara'-dan dön-dü] ben san-dı-m.
Mehmet Ankara-ABL return-PAST I think-PAST-1SG
'That Mehmet has returned from Ankara, I thought.'

d. * [Ben $\_{i}$ san-dı-m] [Mehmet Ankara'-dan dön-dü]$\_{i}$.
I think-PAST-1SG Mehmet Ankara-ABL return-PAST
'I thought that Mehmet has returned from Ankara.'

Local Scrambling of Constituents within Embedded Clauses

Non-finite clauses, with the exception of infinitive and participle clauses (see below), let their constituents scramble within the clause in both the preverbal and the postverbal positions, in the same way as simple sentences. (3.40a,b) below exemplify the preverbal and postverbal scrambling within an embedded S[*fact*] clause, respectively (cf. (3.33a) for the unmarked order for these examples). (3.40c,d) provide examples for similar kinds of scrambling within an embedded S[*ger*] clause (cf. (3.33d) for the unmarked order for these examples).

(3.40) a. Berfu ban-a [Ankara'-dan Mehmet'-in dön-düğ-ü-nü] söyle-di.
Berfu I-DAT Ankara-ABL Mehmet-GEN return-FACT-3sPoss-ACC tell-PAST
'Berfu told me that MEHMET has returned from Ankara.'

b. Berfu ban-a [[$\_{i}$ Ankara'-dan dön-düğ-ü-nü] Mehmet'-in$_i$]
Berfu I-DAT Ankara-ABL return-FACT-3sPoss-ACC Mehmet-GEN
söyle-di.
tell-PAST
'Berfu told me that he, Mehmet, has returned from Ankara.'

c. Berfu [Ankara'-ya Mehmet gid-ince] ben-i ara-dı.
Berfu Ankara-DAT Mehmet go-GER I-ACC phone-PAST
'Berfu phoned me when MEHMET went to Ankara.'

d. Berfu [[$\_{i}$ Ankara'-ya gid-ince] Mehmet$_i$] ben-i ara-dı.
Berfu Ankara-DAT go-GER Mehmet I-ACC phone-PAST
'Berfu phoned me when he, Mehmet, went to Ankara.'

On the other hand, non-finite sentences headed by infinitives and participles are strictly head-final, and let their constituents scramble within the clause only in the preverbal position. For instance, (3.41a,b) show that the constituents of the VP[*inf*] complement in (3.33b) can scramble freely within the embedded clause in the preverbal position, but not otherwise. And (3.41d,e) show that the same case holds true for the S[*part*] clause in (3.41c).

(3.41) a. Berfu [Ankara'-ya Mehmet'-i götür-mek] iste-di.
Berfu Ankara-DAT Mehmet-ACC take-INF want-PAST
'Berfu wanted to take MEHMET to Ankara.'

b. * Berfu [[Mehmet'-i $\_{i}$ götür-mek] Ankara'-ya$_i$] iste-di.
Berfu Mehmet-ACC take-INF Ankara-DAT want-PAST
'Berfu wanted to take Mehmet there, to Ankara.'

c. Berfu [$_{S[part]}$ Mehmet'-in Ahmet'e $\_{i}$ ver-diğ-i] kitab-ı$_i$
Berfu Mehmet-GEN Ahmet-DAT give-PART-3sPoss book-ACC
bil-iyor.
know-PROG
'Berfu knows the book that Mehmet has given to Ahmet.'

d. Berfu [Ahmet'e Mehmet'-in $\_{i}$ ver-diğ-i] kitab-ı$_i$ bil-iyor.
Berfu Ahmet-DAT Mehmet-GEN give-PART-3sPoss book-ACC know-PROG
'Berfu knows the book that MEHMET has given to Ahmet.'

e. * Berfu [[$\_{j}$ Ahmet'e $\_{i}$ ver-diğ-i] Mehmet'-in$_j$] kitab-ı$_i$
Berfu Ahmet-DAT give-PART-3sPoss Mehmet-GEN book-ACC
bil-iyor.
know-PROG
'Berfu knows the book that he, Mehmet, has given to Ahmet.'

Unmarked finite embedded clauses let their constituents freely scramble within the clause both preverbally and postverbally, as seen in (3.42) (cf. (3.35a) for the unmarked order in this case).

(3.42) a. Ben [Ankara'-dan Mehmet dön-dü] san-dı-m.
I Ankara-ABL Mehmet return-PAST think-PAST-1SG
'I thought that MEHMET has returned from Ankara.'

b. Ben [[Mehmet $\_{i}$ dön-dü] Ankara'-dan$_i$] san-dı-m.
I Mehmet return-PAST Ankara-ABL think-PAST-1SG
'I thought that Mehmet has returned from there, Ankara.'

Finally, in the case of marked finite embedded clauses, constituents of the S[*unmarked, fin*] head daughter can again scramble within that daughter both preverbally and postverbally, as seen in (3.43) (cf. (3.34) for the unmarked order in this case).

(3.43) a. Berfu ban-a de-di [ki [Ankara'-dan Mehmet dön-dü.]]
Berfu I-DAT tell-PAST that Ankara-ABL Mehmet return-PAST
'Berfu told me that MEHMET has returned from Ankara.'

b. Berfu ban-a de-di [ki [[Mehmet $\_{i}$ dön-dü] Ankara'-dan$_i$.]]
Berfu I-DAT tell-PAST that Mehmet return-PAST Ankara-ABL
'Berfu told me that Mehmet has returned from there, Ankara.'

Long-distance Scrambling

In addition to local scrambling within the boundaries of embedded clauses, constituents of such clauses can also occur in certain positions of the matrix sentence, namely the sentence-inital and the post-predicate position, giving rise to long-distance dependencies (Erguvanlı (1984), Hoffman (1995)). I refer to the former case (where a constituent of an embedded clause occurs in the sentence-initial position of the matrix clause) as long-distance topicalization and the latter case (where a constituent of an embedded clause occurs in the post-predicate position of the matrix clause) as long-distance backgrounding, implying only the variation in the word order, rather than the specific discourse conditions conveyed by those cases. All three embedded clause types mentioned in Section 3.4.2 let their constituents occur in the matrix sentence positions in this way.

(3.44a,b), for example, exemplify the long-distance topicalization and long-distance backgrounding, respectively, of the genitive subject, *'Mehmet'in'*, of the embedded S[*fact*] complement.

(3.44) a. Mehmet'-in$_i$, [Berfu ban-a [$\_{i}$ Ankara'-dan dön-düğ-ü-nü]
Mehmet-GEN Berfu I-DAT Ankara-ABL return-FACT-3sPoss-ACC
söyle-di.]
tell-PAST
'As for Mehmet, Berfu told me that he has returned from Ankara.'

b. [Berfu ban-a [$\_{i}$ Ankara'-dan dön-düğ-ü-nü] söyle-di]
Berfu I-DAT Ankara-ABL return-FACT-3sPoss-ACC tell-PAST
Mehmet'-in$_i$.
Mehmet-GEN
'Berfu told me that he, Mehmet, has returned from Ankara.'

Similarly, (3.45a,b) are examples where the ablative complement, *'Ankara'dan'*, of the S[*ki*] complement occurs in the sentence-initial position and the postverbal position in the matrix sentence, respectively. (Note that long-distance backgrounding out of a marked finite clause, as in (3.45b), in fact coincides with local backgrounding within such a clause (cf. (3.43b)), since those clauses always immediately follow the matrix verb.)

(3.45) a. Ankara'-dan$_i$, [Berfu ban-a de-di [ki [Mehmet $\_{i}$ dön-dü]]]
Ankara-ABL Berfu I-DAT tell-PAST that Mehmet return-PAST
'As for Ankara, Berfu told me that Mehmet has returned from there.'

b. [Berfu ban-a de-di [ki [Mehmet $\_{i}$ dön-dü]]] Ankara'-dan$_i$.
Berfu I-DAT tell-PAST that Mehmet return-PAST Ankara-ABL
'Berfu told me that Mehmet has returned from there, Ankara.'

Finally, (3.46a,b) provide examples where the ablative NP complement of the embedded S[*fin,unmarked*] complement clause has been long-distance fronted and long-distance backgrounded in the matrix clause, respectively.

(3.46) a. Ankara'-dan$_i$, [ben [Mehmet $\_{i}$ dön-dü] san-dı-m.]
Ankara-ABL I Mehmet return-PAST think-PAST-1SG
'As for Ankara, I thought that Mehmet has returned from there.'

b. [Ben [Mehmet $\_{i}$ dön-dü] san-dı-m] Ankara'-dan$_i$.
I Mehmet return-PAST think-PAST-1SG Ankara-ABL
'I thought that Mehmet has returned from there, Ankara.'

Note, however, that in none of the embedded clause types can a constituent extracted out of an embedded clause occur in the immediately preverbal (focus) position in the matrix sentence (Erguvanlı (1984), Hoffman (1995)). In other words, long-distance focusing is not possible in Turkish.[15] This restriction is exemplified in (3.47a-c) for complex sentences with an S[*fact*], S[*ki*], and S[*fin, unmarked*] complement clause, respectively.

(3.47) a. * Berfu ban-a [$\_{i}$ Ankara'-dan dön-düğ-ü-nü] Mehmet'-in$_i$
Berfu I-DAT Ankara-ABL return-FACT-3sPoss-ACC Mehmet-GEN
söyle-di.
tell-PAST
'Berfu told me that MEHMET has returned from Ankara.'

b. * Berfu ban-a Mehmet$_i$ de-di [ki [$\_{i}$ Ankara'-dan dön-dü]]
Berfu I-DAT Mehmet tell-PAST that Ankara-ABL return-PAST
'Berfu told me that MEHMET has returned from Ankara.'

c. * Ben [Mehmet $\_{i}$ dön-dü] Ankara'-dan$_i$ san-dı-m.
I Mehmet return-PAST Ankara-ABL think-PAST-1SG
'I thought that Mehmet has returned from ANKARA.'

One point to note is that (3.47a,c) have the same surface form as the cases in (3.40b) and (3.42b) (both repeated below), respectively, which exemplify backgrounding within the clause boundaries of the respective complement kinds. However, these two cases convey pragmatic conditions that are completely different than the corresponding cases in (3.47), as indicated by the English translations in each case.

[15] Considering certain contrasts between the facts of 'focus projection' in Turkish and those in Hungarian, Vallduví and Engdahl (1996) conclude that the syntactic focus assignment in Turkish is achieved by removing the nonfocal elements from within the focus domain (as in Catalan), rather than moving the focal elements to a focus slot (as in Hungarian). One may suggest a relationship between their conclusion and the fact that long-distance focusing is not possible in Turkish.

(3.40b) Berfu ban-a [[$\_{i}$ Ankara'-dan dön-düğ-ü-nü] Mehmet'-in$_i$] söyle-di.
Berfu I-DAT Ankara-ABL return-FACT-3sPoss-ACC Mehmet-GEN tell-PAST
'Berfu told me that he, Mehmet, has returned from Ankara.'

(3.42b) Ben [[Mehmet $\_{i}$ dön-dü] Ankara'-dan$_i$] san-dı-m.
I Mehmet return-PAST Ankara-ABL think-PAST-1SG
'I thought that Mehmet has returned from there, Ankara.'

When it comes to long-distance topicalization and backgrounding, Turkish doesn't exhibit any island effects as to extraction of constituents out of relative clauses or sentential subjects (Hoffman (1995)).[16] Thus, for example, the genitive NP subject of the relative clause (S[*part*]) in (3.48a) can be extracted out of that clause, occurring in the sentence-initial position, as seen in (3.48b).

(3.48) a. Ben [$_{S[part]}$ Berfu'nun $\_{i}$ oku-duğ-u] kitab-ı$_i$ bil-iyor-um.
I Berfu-GEN read-PART-3sPoss book-3sPoss know-PROG-1SG
'I know the book Berfu is reading.'

b. Berfu'nun$_j$ [ben [$\_{j}$ $\_{i}$ oku-duğ-u] kitab-ı$_i$ bil-iyor-um.]
Berfu-GEN I read-PART-3sPoss book-3sPoss know-PROG-1SG
'As for Berfu, I know the book she is reading.'

Likewise, (3.49b) shows the long-distance backgrounding of the dative NP complement of the VP[*inf*] subject clause in (3.49a).

(3.49) a. [$_{VP[inf]}$ Çocuğ-a bak-mak] kadın-ı yor-du.
child-DAT look after-INF woman-ACC make tired-PAST
'To look after the child has made the woman tired.'

b. [[$\_{i}$ Bak-mak] kadın-ı yor-du] çocuğ-a$_i$.
look after-INF woman-ACC make tired-PAST child-DAT
'To look after him, the child, has made the woman tired.'

[16] In the case of relativization, however, subject infinitive phrases behave as islands; see Sezer (1986), and also Chapter 4 (Section 4.2.5) of this dissertation.

Long-distance Scrambling 'within' Embedded Clauses

Notice that the above characterization introduces long-distance scrambling in Turkish as a matrix sentence phenomenon, in that it only mentions the cases where the constituent extracted out of an embedded clause occurs in a matrix sentence position. However, this kind of scrambling is, in fact, not only limited to such cases in Turkish. It is also possible for a constituent extracted out of an inner clause to occur in the clause-initial or post-predicate position of a surrounding clause which is itself an embedded clause.

Consider, for example, (3.50a), with two S[*fact*] clauses one embedded within the other. (3.50b,c) are examples where the dative NP complement, *'Ankara'ya'*, of the inner S[*fact*] clause occurs in the sentence-initial and post-predicate positions of the outer S[*fact*] clause, respectively.

(3.50) a. Ben $[_{S[fact]}$ Berfu'-nun $[_{S[fact]}$ Mehmet'-in Ankara'-ya
I Berfu-GEN Mehmet-GEN Ankara-DAT
git-tiğ-i-ne] inan-dığ-ı-nı] san-dı-m.
go-FACT-3sPoss-DAT believe-FACT-3sPoss-ACC think-PAST-1SG
'I thought that, as for Ankara, Berfu believed that Mehmet went there.'

b. Ben [Ankara'-ya$_i$ [Berfu'-nun [Mehmet'-in $\__i$ git-tiğ-i-ne]
I Ankara-DAT Berfu-GEN Mehmet-GEN go-FACT-3sPoss-DAT
inan-dığ-ı-nı] san-dı-m.
believe-FACT-3sPoss-ACC think-PAST-1SG
'I thought that, as for Ankara, Berfu believed that Mehmet went there.'

c. Ben [[Berfu'-nun [Mehmet'-in $\__i$ git-tiğ-i-ne]
I Berfu-GEN Mehmet-GEN go-FACT-3sPoss-DAT
inan-dığ-ı-nı] Ankara'-ya$_i$] san-dı-m.
believe-FACT-3sPoss-ACC Ankara-DAT think-PAST-1SG
'I thought that Berfu believed that Mehmet went there, to Ankara.'

(3.51a) is an example with an S[*fact*] clause embedded in an S[*ki*] complement. (3.51b,c) show that a constituent of the embedded S[*fact*] clause (in this case the dative NP, *Ankara'ya*) can occur in the sentence-initial position of the S[*unmarked,fin*] head daughter of the S[*ki*] complement, but not in the initial position in the S[*ki*] complement itself. In the case of long-distance backgrounding, on the other hand, it is not quite clear whether the backgrounding takes place within the S[*unmarked, fin*] head daughter or the S[*ki*] complement itself. For the sake of argument, I assume that it takes place within the head daughter, rather than the *marked* clause.

(3.51) a. Ben san-dı-m [$_{S[ki]}$ ki [$_{S[unmarked]}$ Berfu [$_{S[fact]}$ Mehmet'-in
I think-PAST-1SG that Berfu Mehmet-GEN
Ankara'-ya git-tiğ-i-ne] inan-dı.]]
Ankara-DAT go-FACT-3sPoss-DAT believe-PAST
'I thought that Berfu believed that Mehmet went to Ankara.'

b. Ben san-dı-m [ki [Ankara'-ya$_i$ [Berfu [Mehmet'-in $\__i$
I think-PAST-1SG that Ankara-DAT Berfu Mehmet-GEN
git-tiğ-i-ne] inan-dı.]]]
go-FACT-3sPoss-DAT believe-PAST
'I thought that, as for Ankara, Berfu believed that Mehmet went there.'

c. * Ben san-dı-m [Ankara'-ya$_i$ [ki [Berfu [Mehmet'-in $\__i$
I think-PAST-1SG Ankara-DAT that Berfu Mehmet-GEN
git-tiğ-i-ne] inan-dı.]]]
go-FACT-3sPoss-DAT believe-PAST

d. Ben san-dı-m [ki [[Berfu [Mehmet'-in $\__i$ git-tiğ-i-ne]
I think-PAST-1SG that Berfu Mehmet-GEN go-FACT-3sPoss-DAT
inan-dı] Ankara'-ya$_i$.]]
believe-PAST Ankara-DAT
'I thought that Berfu believed that Mehmet went there, to Ankara.'

Finally, (3.52a) is an example with an S[*unmarked,fin*] complement which itself contains an embedded S[*fact*] complement. (3.52b,c) exemplify the long-distance topicalization and backgrounding, respectively, of the dative NP complement, *'Ankara'ya'*, of the inner S[*fact*] clause, in the S[*fin, unmarked*] complement clause.

(3.52) a. Ben [$_{S[fin]}$ Berfu [$_{S[fact]}$ Mehmet'-in Ankara'-ya git-tiğ-i-ne]
I Berfu Mehmet-GEN Ankara-DAT go-FACT-3sPoss-DAT
inan-dı] san-dı-m.
believe-PAST think-PAST-1SG
'I thought that Berfu believed that Mehmet went to Ankara.'

b. Ben [Ankara'-ya$_i$ [Berfu [Mehmet'-in $\__i$ git-tiğ-i-ne] inan-dı]]
I Ankara-DAT Berfu Mehmet-GEN go-FACT-3sPoss-DAT believe-PAST
san-dı-m.
think-PAST-1SG
'I thought that, as for Ankara, Berfu believed that Mehmet went there.'

c. Ben [[Berfu [Mehmet'-in $\__i$ git-tiğ-i-ne] inan-dı] Ankara'-ya$_i$]
I Berfu Mehmet-GEN go-FACT-3sPoss-DAT believe-PAST Ankara-DAT
san-dı-m.
think-PAST-1SG
'I thought that Berfu believed that Mehmet went there, to Ankara.'

Long-distance Scrambling 'within' Infinitive and Participle Clauses

We have seen in Section 3.4.2 (cf. example (3.41)) that local backgrounding cannot occur in infinitive and participle clauses. It is essential then to find out whether long-distance scrambling can occur within such clauses, in order to conclude whether they can be allowed to head head-filler phrases in Turkish.

Consider first (3.53a), with a VP[*inf*] complement which itself contains a dative S[*fact*] complement. (3.53b,c) show that a constituent of the embedded S[*fact*] complement (in this case the dative NP *'Ankara'ya'*) can be long-distance fronted, but not backgrounded within the VP[*inf*] clause.

(3.53) a. Ben [$_{\text{VP}[inf]}$ Berfu'-yu [$_{\text{S}[fact]}$ Mehmet'-in Ankara'-ya
I Berfu-ACC Mehmet-GEN Ankara-DAT
git-tiğ-i-ne] inan-dır-mak] iste-di-m.
go-FACT-3sPoss-DAT believe-CAUS-INF want-PAST-1SG
'I wanted to convince Berfu that Mehmet had gone to Ankara.'

b. Ben [Ankara'-ya$_i$ [Berfu'-yu [Mehmet'-in $\__i$ git-tiğ-i-ne]
I Ankara-DAT Berfu-ACC Mehmet-GEN go-FACT-3sPoss-DAT
inan-dır-mak]] iste-di-m.
believe-CAUS-INF want-PAST-1SG
'I wanted to convince Berfu that, as for Ankara, Mehmet had gone there.'

c. * Ben [[Berfu'-yu [Mehmet'-in $\__i$ git-tiğ-i-ne] inan-dır-mak]
I Berfu-ACC Mehmet-GEN go-FACT-3sPoss-DAT believe-CAUS-INF
Ankara'-ya$_i$] iste-di-m.
Ankara-DAT want-PAST-1SG
'I wanted to convince Berfu that Mehmet had gone there, to Ankara.'

Next, consider (3.54a), with a participle clause containing an S[*fact*] complement. Again, (3.54b,c) show that a constituent of the embedded S[*fact*] complement (in this case the genitive NP subject *'Mehmet'in'*) can be long-distance fronted, but not backgrounded within the S[*part*] clause.

(3.54) a. Ben [$_{\text{S}[part]}$ $\__i$ Berfu'-ya [$_{\text{S}[fact]}$ Mehmet'-in pasta-yı
I Berfu-DAT Mehmet-GEN cake-ACC
ye-diğ-i-ni] söyle-yen] çocuğ-u$_i$ tanı-yor-um.
eat-FACT-3sPoss-ACC tell-PART child-ACC know-PROG-1SG
'I know the child who told Berfu that Mehmet ate the cake.'

b. Ben [Mehmet'-in$_j$ [$\_{i}$ Berfu'-ya [$\_{j}$ pasta-yı ye-diğ-i-ni]
I Mehmet-GEN Berfu-DAT cake-ACC eat-FACT-3sPoss-ACC
söyle-yen]] çocuğ-u$_i$ tanı-yor-um.
tell-PART child-ACC know-PROG-1SG
'I know the child who told Berfu that, as for Mehmet, he ate the cake.'

c. * Ben [[$\_{i}$ Berfu'-ya [$\_{j}$ pasta-yı ye-diğ-i-ni] söyle-yen]
I Berfu-DAT cake-ACC eat-FACT-3sPoss-ACC tell-PART
Mehmet'-in$_j$] çocuğ-u$_i$ tanı-yor-um.
Mehmet-GEN child-ACC know-PROG-1SG
'I know the child who told Berfu that he, Mehmet, ate the cake.'

Thus, VP[*inf*] and S[*part*] clauses do allow long-distance topicalization within their clause boundaries (hence can head certain kinds of head-filler phrases), but not backgrounding (local or long-distance).

3.4.3 Scrambling of Possessors and Postpositional Objects

As well as the clausal constituents of complex sentences, possessive NPs too let their arguments (i.e. possessors, which I argue should be treated as subjects, see Section 3.3) be long-distance fronted, or local or long-distance backgrounded. Consider, for example, (3.55a) with a possessive NP complement, whose genitive NP subject (possessor), *'Mehmet'in'*, can be long-distance fronted as in (3.55b), occurring in the sentence-initial position. The same genitive NP can also be both backgrounded within the possessive NP itself (i.e. occur immediately after the possessed head noun) as in (3.55c), and long-distance backgrounded in the main sentence (i.e. occur after the matrix verb) as in (3.55d).

(3.55) a. Berfu [$_{NP}$ Mehmet'-in kitab-ı-nı] oku-yor.
Berfu Mehmet-GEN book-3sPoss-ACC read-PROG
'Berfu is reading Mehmet's book.'

b. Mehmet'-in$_i$ [Berfu [$\_{i}$ kitab-ı-nı] oku-yor.]
Mehmet-GEN Berfu book-3sPoss-ACC read-PROG
'As for Mehmet, Berfu is reading his book.'

c. Berfu [[$\_{i}$ kitab-ı-nı] Mehmet'-in$_i$] oku-yor.
Berfu book-3sPoss-ACC Mehmet-GEN read-PROG
'Berfu is reading his book, Mehmet.'

d. [Berfu [$\_{i}$ kitab-ı-nı] oku-yor] Mehmet'-in$_i$.
Berfu book-3sPoss-ACC read-PROG Mehmet-GEN
'Berfu is reading his book, Mehmet.'

(3.56) below is a similar example, where the genitive possessor, *'sahaflığın'*, of the subject possessive NP occurs sentence-initially.[17]

(3.56) Sahaflığ-ın$_i$ [eskiden [$\_{i}$ çok renkli
antique book selling business-GEN in the past very/many colourful
yan-lar-ı] var-dı.]
facet-PLU-3sPoss existent-PAST
'As for the antique book selling business, back in the old times, it had very/many colorful facets.'

In addition, objects of *obj-agr* postpositions (i.e. the ones that bear agreement morphology with their objects, see pages 5 and 34) are also allowed to scramble in exactly the same way as possessors, provided that they are genitive marked. So, for example, the object of *'hakkında'* in (3.57) can scramble in exactly the same way as the possessors above, if it is genitive marked, but not when it is unmarked.

(3.57) a. Biz [$_{PP}$ Berfu'-nun/Berfu hakk-ı-nda] konuş-uyor-du-k.
we Berfu-GEN/Berfu about(-3sPoss-LOC) talk-PROG-PAST-1PL
'We were talking about Berfu.'

b. Berfu'-nun$_i$/*Berfu$_i$ [biz [$\_{i}$ hakk-ı-nda] konuş-uyor-du-k.]
Berfu-GEN/Berfu we about(-3sPoss-LOC) talk-PROG-PAST-1PL
'As for Berfu, we were talking about her.'

c. Biz [[$\_{i}$ hakk-ı-nda] Berfu'-nun$_i$/*Berfu$_i$] konuş-uyor-du-k.
we about(-3sPoss-LOC) Berfu-GEN/Berfu talk-PROG-PAST-1PL
'We were talking about her, Berfu.'

d. [Biz [$\_{i}$ hakk-ı-nda] konuş-uyor-du-k] Berfu'-nun$_i$/*Berfu$_i$.
we about(-3sPoss-LOC) talk-PROG-PAST-1PL Berfu-GEN/Berfu
'We were talking about her, Berfu.'

3.4.4 Further Limitations on Scrambling

Local and long-distance scrambling may interact with each other in various combinations. Consider, for example, (3.58a), a complex sentence with an S[*fact*] complement (with every constituent in its unmarked position). (3.58b) shows the case where the nominative NP subject, *'Berfu'*, of the main sentence occurs in the immediately preverbal (focus) position, while the S[*fact*] complement, whose genitive subject *'Mehmet'in'*

[17]This example is taken from an article titled "Sahaflık neden can çekişiyor?" by Sami Önal, published in Cumhuriyet Kitap, August 19, 1993.

has been long-distance topicalized, itself occurs in the postverbal position of the main sentence. In addition, the ablative NP complement, *'Ankara'dan'*, of this complement clause occurs in the postverbal position of the clause. Perhaps surprisingly, this example is perfectly easy to interpret. (The feature structure for (3.58b) is presented below in (3.73), page 77.)

(3.58) a. Berfu ban-a [$_{S[fact]}$ Mehmet'-in Ankara-dan dön-düğ-ü-nü]
Berfu I-DAT Mehmet-GEN Ankara-ABL return-FACT-3sPoss-ACC
söyle-di.
tell-PAST
'Berfu told me that Mehmet has returned from Ankara.'

b. [Mehmet'-in$_i$ [[ban-a $\_{j}$ Berfu söyle-di] [$\_{i}$ $\_{k}$ dön-düğ-ü-nü]$_j$]]
Mehmet-GEN I-DAT Berfu tell-PAST return-FACT-3sPoss-ACC
Ankara-dan$_k$.
Ankara-ABL
'As for Mehmet, BERFU told me that he has returned from there, Ankara.'

One point to note is that long-distance scrambling of an NP into a clause that has another NP with the same case is in certain cases blocked. Consider, for example, (3.59), from Hoffman (1995). Although the dative NP complement *'eve'* of the S[*fin*] clause in (3.59a) can be long-distance topicalized in the matrix sentence, as seen in (3.59b), the same doesn't hold for the nominative NP subject, *'Ali'*, of the same clause, as seen in (3.59c). Hoffman notes that in such ambiguous cases speakers prefer the reading where each NP is interpreted as the argument of the closest centre-embedded verb, i.e. the reading in (3.59d).

(3.59) a. Fatma [$_{S[fin,unmarked]}$ Ali ev-e git-ti] san-dı.
Fatma Ali house-DAT go-PAST think-PAST
'Fatma thought Ali went home.'

b. Ev-e$_i$ [Fatma [Ali $\_{i}$ git-ti] san-dı.]
house-DAT Fatma Ali go-PAST think-PAST
'To the house, Fatma thought that Ali went there.'

c. * Ali$_i$ [Fatma [$\_{i}$ ev-e git-ti] san-dı.]
Ali Fatma house-DAT go-PAST think-PAST
'As for Ali, Fatma thought that he went home.'

d. Ali [Fatma ev-e git-ti] san-dı.
Ali Fatma house-DAT go-PAST think-PAST
'Ali thought that Fatma went home.'

One might argue that this is a syntactic restriction on long-distance scrambling. However, Hoffman notes the following example, which shows that long-distance scrambling of an NP into a clause which contains another NP with the same case is possible if the two NPs (in this case the dative NPs *'Esra'ya'* and *'Fatma'ya'*) are far enough apart.

(3.60) Esra'-ya$_i$, [Ahmet [ben-im $\_\_i$ yardım et-tiğ-im-i], Fatma'-ya söyle-di].
Esra-DAT Ahmet I-GEN help do-FACT-1sPoss-ACC Fatma-DAT say-PAST
'As for Esra, Ahmet told Fatma that I helped her.'

Moreover, consider the two cases in (3.61), where the nominative subject of the embedded S[*fin*] clause occurs sentence-initially just before the nominative subject of the main clause just like (3.59c) above, yet the agreement morphology on the verbs uniquely denotes the actual predicate-argument structures. It is interesting to note that (3.61a) is perfectly acceptable, whereas (3.61b) seems to be ruled out.

(3.61) a. Fatma$_i$ [ben [$\_\_i$ ev-e git-ti] san-dı-m.]
Fatma I house-DAT go-PAST think-PAST-1SG
'As for Fatma, I thought that she went home.'

b. * Sen$_i$ [Fatma [$\_\_i$ ev-e git-ti-n] san-dı.]
you Fatma house-DAT go-PAST-2SG think-PAST
'As for you, Fatma thought that you went home.'

Thus, I agree with Hoffman (1995) that this 'apparent' restriction on long-distance scrambling is actually a processing limitation (rather than a syntactic restriction), due to the significance of case-marking in disambiguation of the predicate-argument structures of clauses in Turkish.

Finally, note once again that in this section I have only concerned myself with the syntactic constraints on word order variation. Recall from Section 3.4.1 that referential status of NPs impose further constraints on scrambling due to the pragmatic aspect of word order variation in Turkish. We have seen, for example, that indefinite NPs in Turkish tend to occur in the immediately preverbal position, rather than the other sentence positions. Similar restrictions also apply to long-distance scrambling. So, for example, the genitive marked indefinite subject, *'bir paketin'*, of the S[*fact*] clause in (3.62a) cannot be long-distance topicalized or backgrounded, as seen in (3.62b,c), respectively.

(3.62) a.

Berfu	[masa-nın	üst-ü-nde	bir	paket-in	dur-duğ-u-nu]
Berfu	table-GEN	top-3sPoss-LOC	a	packet-GEN	lie-FACT-3sPoss-ACC

gör-dü.
see-PAST
'Berfu saw that a packet was lying on the table.'

b.

* Bir	paket-in$_i$	[Berfu	[masa-nın	üst-ü-nde	$\_{i}$ dur-duğ-u-nu]
a	packet-GEN	Berfu	table-GEN	top-3sPoss-LOC	lie-FACT-3sPoss-ACC

gör-dü.]
see-PAST
'As for a packet, Berfu saw that it was lying on the table.'

c.

* [Berfu	[masa-nın	üst-ü-nde	$\_{i}$ dur-duğ-u-nu]	gör-dü]	bir
Berfu	table-GEN	top-3sPoss-LOC	lie-FACT-3sPoss-ACC	see-PAST	a

paket-in$_i$.
packet-GEN
'Berfu saw that it was lying on the table, a packet.'

Lee (1993) notes similar effects of certain discourse notions on scrambling in Korean, another "free" word order language. Consider, for example, (3.63a) (from Lee), where the indefinite object can be interpreted as either specific or nonspecific, and only the specific reading is possible in the case of scrambling, as seen in (3.63b,c).

(3.63) a.

Minho-ka	cantipat-eyse	chayk-ul	ilkessta.
Minho-NOM	grass-LOC	book-ACC	read

'Minho read a (specific or non-specific) book on the grass.'

b.

Minho-ka	chayk-ul	cantipat-eyse	ilkessta.
Minho-NOM	book-ACC	grass-LOC	read

c.

chayk-ul	Minho-ka	cantipat-eyse	ilkessta.
book-ACC	Minho-NOM	grass-LOC	read

'Minho read a (specific) book on the grass.'

Similarly, in (3.64a) (again from Lee), the in-situ object *'yumyeng violinist'* is ambiguous between specific and non-specific readings, but only the specific reading is available in the case of both local and long-distance scrambling, as seen in (3.64b,c), respectively.[18]

[18]I must note, however, that on the basis of certain other cases, Lee concludes that the discourse notion that correctly characterizes the constraints on scrambling in Korean is in fact 'presuppositionality' (Diesing (1990)), rather than specificity.

(3.64) a.

Minho-ka	[lotte	hotel-eyse	yumyeng	violinist-lul	poassta-ko]
Minho-NOM	lotte	hotel-LOC	famous	violinist-ACC	saw-COMP

calanghayssta.
said proudly
'Minho said proudly that he saw a (specific or non-specific) famous violinist at Hotel Lotte.'

b.

Minho-ka	[yumyeng	violinist-lul	lotte	hotel-eyse	poassta-ko]
Minho-NOM	famous	violinist-ACC	lotte	hotel-LOC	saw-COMP

calanghayssta.
said proudly

c.

yumyeng	violinist-lul$_i$	Minho-ka	[$\_{i}$	lotte	hotel-eyse	poassta-ko]
famous	violinist-ACC	Minho-NOM		lotte	hotel-LOC	saw-COMP

calanghayssta.
said proudly
'Minho said proudly that he saw a (specific) famous violinist at Hotel Lotte.'

3.4.5 Word Order Constraints in HPSG

Taking into account the restrictions on word order variation in complex sentences, below I revise the LP constraints stated in Section 3.4.1, and impose further constraints on certain phrase types to capture the restrictions on local and long-distance scrambling in complex sentences.

Head-final Constraint

Considering that *marked* finite clauses (that is, S[*ki*] and S[*çünkü*] clauses) always (immediately) follow their head (cf. Section 3.4.2, page 57), I propose the following two LP constraints to restrict the order of the head and its sisters in a phrasal sign of Turkish, with respect each other.

(3.65) a. [*unmarked*] < HEAD

b. HEAD < [*marked*]

(3.65a) states that in a phrasal sign of Turkish, the head follows any of its *unmarked* sisters, and (3.65b) restricts the head to precede its *marked* sister(s), if any. Note that these two LP constraints together further ensure that a *marked* clause always *immediately* follows the head in a phrasal sign.

Non-case-marked Objects

We have observed above that non-case-marked object complements in Turkish are not allowed to scramble, 'non-case-marked' meaning complements with the morphologically unmarked *nominative* as their CASE value, or those with no appropriate CASE feature at all, i.e. VP[*inf*] and S[*fin*] complements.

Keeping in mind that *marked* finite clauses always (immediately) follow their head, we can formulate the above restriction on the order of non-case-marked object complements as in (3.66), where *inf-fin* has the obvious subtypes. (Note that (3.66) replaces the LP constraint in (3.32), Section 3.4.1, page 53; see that LP constraint for the status of NON-SUBJECT in (3.66).)

(3.66) a. COMPLEMENT[*unmarked*] < COMPLEMENT[*noun*[*nom*], NON-SUBJECT]

b. COMPLEMENT[*unmarked*] < COMPLEMENT[*verb*[*inf-fin*], NON-SUBJECT]

(3.66) states that in any phrasal sign of Turkish, a non-subject complement which has a HEAD value of type *noun* with the CASE value constrained as *nominative*, or of type *verb* with the VFORM value constrained as either *inf* or *fin*, should always be preceded by any other *unmarked* sister complements.

Moreover, we need to further constrain the complement extraction lexical rule (first presented in Chapter 2, page 22, and repeated below) in (3.67) in order to block the extraction of non-case-marked objects.

(3.67) $$\begin{bmatrix} \text{COMPS} & \langle \ldots, [\text{LOC}\ \boxed{1}], \ldots \rangle \\ \text{INHER|SLASH} & \boxed{2} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} & \langle \ldots \rangle \\ \text{INHER|SLASH} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

To that end, I propose the use of two separate extraction lexical rules, one for extraction of subjects and the other of non-subject complements. (Although I assume subjects are treated in the same way as the other complements in the sentence structure, such a distinction in terms of extraction in the lexicon seems to be essential to block the extraction of nominative objects, while letting nominative subjects be extracted at the same time.) The following lexical rule deals with the extraction of the subjects of the nouns and *non-base-inf* verbs. (Recall from page 47 that the type *non-base-inf* has the subtypes *fin*, *nomin*, *part* and *ger*.)

(3.68) $$\begin{bmatrix} \text{HEAD} & \textit{noun} \vee \textit{verb}[\textit{non-base-inf}] \\ \text{SUBJ} & \langle [\text{LOC}\ \boxed{1}] \rangle \\ \text{INHER|SLASH} & \boxed{2} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{SUBJ} & \langle \rangle \\ \text{INHER|SLASH} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

(3.68) simply removes the subject argument of a lexical entry which is a noun or a *non-base-inf* verb, including its LOCAL value in the INHER|SLASH set of the output.

The second extraction lexical rule (for complements) is presented in (3.69). Again, *non-base* has the subtypes *fin, nomin, inf, part* and *ger* (cf. page 35), and *non-nominative* (*non-nom*) has the subtypes *genitive, accusative, dative, ablative, locative* and *instrumental*.

$$(3.69)\quad \begin{bmatrix} \text{HEAD} & postp[obj\text{-}agr] \vee verb[non\text{-}base] \\ \text{COMPS} & \langle \ldots, [\text{LOC}\ \boxed{1}[\text{CASE}\ non\text{-}nom]], \ldots \rangle \\ \text{INHER|SLASH} & \boxed{2} \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} & \langle \ldots \rangle \\ \text{INHER|SLASH} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

(3.69) extracts a non-nominative complement of a lexical entry which is an *obj-agr* postposition or a *non-base* verb, placing its LOCAL value in the INHER|SLASH set of the output, as before. (Note that in the actual implementation of this rule the input entry can further be constrained to have a non-empty SUBJ value, if it is a verb, in order to block the application of this lexical rule to the outputs of the lexical rule in (3.19) – which moves the subject of a *non-base-inf* verb to its COMPS list – and to those of (3.68) above, thereby eliminating redundant lexical ambiguity.)

Constraints on Head-filler Phrases

The HPSG grammar in Chapter 2, Table 2.1 constrains the head daughter of a head-filler phrase to be an S[*fin*]. Note, however, that this constraint is too strong for head-filler phrases in Turkish, in two respects.

First, we have seen in Section 3.4.2 that non-finite clauses too can be head daughters of head-filler phrases in Turkish. In addition, NPs and PPs with locally backgrounded arguments are also considered instances of *hd-fill-ph*. Thus, I assume the HEAD value of the head daughter of a head-filler phrase in Turkish grammar is constrained as the disjunction of *noun*, *postp*[*obj-agr*] and *verb*[*non-base*], rather than a *finite* verb.

Second, we have seen in Section 3.4.2 (page 65) that long-distance topicalization is also possible within 'unsaturated' VP[*inf*] complements in Turkish, which suggests that constraining the head daughter of a head-filler phrase to have an empty SUBJ value is too restrictive for Turkish, since it would rule out such cases. I therefore assume that in Turkish grammar, the head daughter of a head-filler phrase is only constrained to have empty COMPS and SPR values.

We have seen in Section 3.4.2 (cf. examples (3.41), (3.53), and (3.54)) that postverbal scrambling (local or long-distance) cannot occur within infinitive and participle clauses in Turkish. To capture this restriction, I assume that the type *hd-fill-ph* in Turkish grammar has the two subtypes *head-topic-phrase* (*hd-topic-ph*) and *head-backg-phrase* (*hd-backg-ph*). Moreover, I introduce an additional type, *vform-backg*, as a supertype for *fin*, *nomin* and *ger*. The above restriction can then be imposed by assuming the following constraint on phrases of type *hd-backg-ph*, which constrains the head daughter of any instance of type *hd-backg-ph* to have a HEAD value constrained as *noun*, *postp*[*obj-agr*] or *verb*[*vform-backg*].

(3.70) *hd-backg-ph* ⇒ [HD-DTR [HEAD *noun* ∨ *postp*[*obj-agr*] ∨ *verb*[*vform-backg*]]]

In addition, since the assumption that head-filler phrases can be headed by nouns and postpositions is made merely on the grounds of accounting for the NPs and PPs with locally backgrounded arguments (see above), I further assume the following constraint on the type *hd-topic-ph*, which constrains the head daughter of any instance of that type to have a HEAD value constrained as *verb*[*non-base*]:

(3.71) *hd-topic-ph* ⇒ [HD-DTR [HEAD *verb*[*non-base*]]]

We have also seen in Section 3.4.2 (cf. example (3.51)) that long-distance scrambling cannot occur within *marked* finite clauses in Turkish. This restriction can simply be captured by (3.72), which constrains the head daughter of any head-filler phrase to have a MARKING value of type *unmarked*.

(3.72) *hd-fill-ph* ⇒ [HD-DTR [MARKING *unmarked*]]

Finally, recall from Section 3.4.2 (cf. example (3.47)) that long-distance focusing is not possible in Turkish. This restriction is trivially captured in the grammar by the fact that the head daughter of any head-filler phrase is constrained to have an empty COMPS value, i.e. to have 'consumed' all its complements. That prevents a constituent extracted out of an embedded clause from intervening between the complements of a head and the head itself.

Table 3.1 summarizes the constraints on the type *hd-fill-ph* and its subtypes in Turkish grammar, and also repeats for convenience the ones on the types *hd-ph* and *hd-nexus-ph* (which are also inherited by *hd-fill-ph* and its subtypes). Note that the head daughter of any instance of *hd-fill-ph* is constrained to have an empty COMPS value by the Empty COMPS Constraint (ECC; cf. (2.22) in Chapter 2 (Section 2.3), page 26), which it inherits from *hd-ph* (cf. Table 3.1).

TYPE	CONSTRAINTS	ISA
phrase		*sign*
hd-ph	HFP, VALP, ECC, NFP, Marking Principle	*phrase*
hd-nexus-ph	[CONTENT [1] HD-DTR [CONTENT [1]]]	*hd-ph*
hd-fill-ph	[HD-DTR [*phrase* HEAD *noun* ∨ *postp*[*obj-agr*] ∨ *verb*[*non-base*] SPR <> MARKING *unmarked* INHER|SLASH {. . . , [1], . . . } TO-BIND|SLASH {[1]}] NON-HD-DTRS ⟨[LOCAL [1]]⟩]	*hd-nexus-ph*
hd-topic-ph	[HD-DTR [HEAD *verb*[*non-base*]]]	*hd-fill-ph*
hd-backg-ph	[HD-DTR [HEAD *noun* ∨ *postp*[*obj-agr*] ∨ *verb*[*vform-backg*]]]	*hd-fill-ph*

Table 3.1: The constraints on type *hd-fill-ph*, and its subtypes and superstypes, in Turkish grammar.

Let us now consider again (3.58b) (first presented in Section 3.4.4, page 68, and repeated below) which illustrates the interaction of different kinds of scrambling. The feature structure for this case is presented in (3.73). (Small 's' and 'np' in (3.73) are used as abbreviations for *local* objects with HEAD values of type *verb* and *noun*, respectively, and with empty values for all three valence features SUBJ, SPR, and COMPS.)

(3.58b)	[Mehmet'-in$_i$	[[ban-a	$\_{j}$	Berfu	söyle-di]	[$\_{i}$ $\_{k}$	dön-düğ-ü-nü]$_j$]]
	Mehmet-GEN	I-DAT		Berfu	tell-PAST		return-FACT-3sPoss-ACC
	Ankara-dan$_k$.						
	Ankara-ABL						

'As for Mehmet, BERFU told me that he has returned from there, Ankara.'

Notice that the nonlocal dependency due to the backgrounded S[*fact*] clause is introduced in the structure in (3.73) by the *local* object (constrained as s[*fact*], tag [8]) in the INHER|SLASH set of the lexical head *'söyledi'*. That dependency is bound off by the TO-BIND|SLASH element of the head daughter of the inner head-backg phrase. Also, the nonlocal dependencies due to the long-distance topicalized genitive subject, *'Mehmet'in'*, and backgrounded ablative complement, *'Ankara'dan'*, of the S[*fact*] clause are introduced by the corresponding *local* objects in the INHER|SLASH set of the verbal head, *'döndüğünü'*, of that clause (tags [10] and [9], respectively). The former dependency is bound off by the TO-BIND|SLASH element of the head daughter of the head-topic phrase (i.e. the inner head-backg phrase), and the latter is bound off by the TO-BIND|SLASH element of the head daughter of the outer head-backg phrase (i.e. the head-topic phrase).

(3.73)

```
[hd-backg-ph
 PHON         ⟨1, 2, 3, 4, 5, 6⟩
 SYNSEM       [HEAD 7
               INHER|SLASH {}]
 NON-HD-DTRS  ⟨[hd-comp-ph
                PHON ⟨6 Ankara'dan⟩
                SYNSEM|LOC 9 np[dat]]⟩
 HEAD-DTR     [hd-topic-ph
               PHON            ⟨1, 2, 3, 4, 5⟩
               SYNSEM          [HEAD 7
                                INHER|SLASH {9}
                                TO-BIND|SLASH {9}]
               NON-HEAD-DTRS   ⟨[hd-comp-ph
                                 PHON ⟨1 Mehmet'in⟩
                                 SYNSEM|LOC 10 np[gen]]⟩
               HEAD-DTR        [hd-backg-ph
                                PHON           ⟨2, 3, 4, 5⟩
                                SYNSEM         [HEAD 7
                                                INHER|SLASH {10, 9}
                                                TO-BIND|SLASH {10}]
                                NON-HEAD-DTRS  ⟨[hd-comp-ph
                                                 PHON ⟨5 döndüğünü⟩
                                                 SYNSEM [LOC 8 s[fact]
                                                         INHER|SLASH {10, 9}]
                                                 NON-HD-DTRS ⟨⟩
                                                 HEAD-DTR [word
                                                           COMPS ⟨⟩
                                                           INHER|SLASH {10, 9}]]⟩
                                HEAD-DTR       [hd-comp-ph
                                                PHON           ⟨2, 3, 4⟩
                                                SYNSEM         [HEAD 7
                                                                COMPS ⟨⟩
                                                                INHER|SLASH {8}
                                                                TO-BIND|SLASH {8}]
                                                NON-HEAD-DTRS  ⟨[hd-comp-ph
                                                                 PHON ⟨3 Berfu⟩
                                                                 SYNSEM 11 NP[nom]],
                                                                [hd-comp-ph
                                                                 PHON ⟨2 bana⟩
                                                                 SYNSEM 12 NP[dat]]⟩
                                                HEAD-DTR       [word
                                                                PHON ⟨4 söyledi⟩
                                                                SYNSEM [HEAD 7 verb[fin]
                                                                        COMPS ⟨11, 12⟩
                                                                        INHER|SLASH {8}]]]]]]
```

3.4.6 Use of Pragmatic Notions in the LP Constraints

As mentioned earlier in this section, the LP constraint in (3.65a), which constrains the head to follow any of its *unmarked* sisters in any phrasal sign of Turkish, is in fact too restrictive, since it excludes cases of backgrounding. So far, I haven't actually proposed an LP constraint that accounts for cases of backgrounding, that is, that constrains the head to precede its sister in such phrasal signs. To that end, it seems appropriate to exploit pragmatic, as well as syntactic, notions in the LP constraints for Turkish.[19] So, for example, considering both the facts related to the information structure of Turkish sentences and the syntactic constraints on word order, one could propose the LP constraints in (3.74) for Turkish, where TOPIC, FOCUS and BACKG abbreviate whatever analysis one adopts for the pragmatic notions of topic, focus and background in Turkish, respectively. Considering that in Turkish information structure significantly interacts with both word order and prosody, such an analysis will need to involve both prosodic and contextual constraints, as mentioned before. (Note that the use of this particular set of pragmatic notions, proposed by Erguvanlı (1984), in these LP constraints is merely tentative at this point.)

(3.74)
a. [TOPIC +] < [ ]
b. [ ] < [BACKG +]
c. [BACKG -, *unmarked*] < HEAD
d. HEAD < [*marked*]
e. COMPLEMENT[*unmarked*] < COMPLEMENT[$noun[nom] \vee verb[inf \vee fin]$, NON-SUBJECT]
f. COMPLEMENT[*unmarked*] < COMPLEMENT[FOCUS +]

(3.74a) states that in any phrasal sign of Turkish, the topic, if any, is constrained to precede any of its sisters, and (3.74b) constrains the background to follow any of its sisters. (3.74c) constrains the head, in any phrasal sign of Turkish, to follow any of its *unmarked* sisters that is *not* constrained to be the background. (3.74d) constrains the head to precede its *marked* sister(s), if any, and (3.74e) constrains any non-case-marked complement to be preceded by any other *unmarked* sister complements, as before. Note that the status of focus is rather complicated in Turkish, in that it can be marked either i) via word order variation, by placing the focal constituent in the immediately preverbal position (as we have seen so far); or ii) by prosodically marking the focal constituent with nuclear stress without changing its position in the sentence (cf. Erguvanlı (1984),[20] and Vallduví and Engdahl (1996)). Although one can adopt the LP constraint in (3.74f) for the former case, the latter apparently shouldn't be constrained by principles of linear ordering.

[19] Recall from Section 3.4.1 that the idea is also independently motivated by the discussion on the linear order of indefinite NPs in Turkish sentences; cf. pages 50–52.

[20] Erguvanlı argues that the latter strategy signals a 'focus of contrast', and is possible for only certain kinds of constituents.

It should be noted that the use of pragmatic notions in the linear ordering of constituents has previously been proposed in the literature in various frameworks and for several languages. For example, in their Functional Unification Grammar (Kay (1985)) analysis of Finnish, Karttunen and Kay (1985) exploit discourse roles such as 'topic' and 'contrast' as well as syntactic notions such as 'subject' and 'finite verb', in the linear ordering of constituents. Similarly, the GPSG analysis of German word order by Uszkoreit (1987) proposes certain LP rules that make use of the notion of 'focus', in addition to the case marking of NPs and their syntactic category (i.e. +/- PRONOUN). Another example is the Focus Rule of Pollard and Sag (1987)[Chapter 7], proposed for 'heavy constituent shift' in English. Also, along these lines comes the LFG analysis of Russian phrase structure by King (1995)[Chapters 6-8], who proposes a hierarchical phrase structure for Russian, where certain positions are associated with particular discourse functions, to account for the interaction between word order and discourse function interpretation in Russian. In addition, the distribution of grammatical functions in the phrase structure is governed by the interaction of 'functional uncertainty' (Kaplan and Zaenen (1988)) with the well-formedness conditions on the f-structure. Within the HPSG framework, Vallduví and Engdahl (1995) propose an LP constraint for Catalan to constrain the order in which 'link', 'focus' and 'tail' are realized in Catalan phrase structure. Finally, in her Multiset-CCG analysis of Turkish, Hoffman (1995) makes use of certain pragmatic word order constraints which associate information structure components such as topic and focus with the appropriate sentence positions.

3.5 Summary

In this chapter, I have analyzed certain phenomena in Turkish grammar, within the HPSG framework. In Section 3.1, I summarized certain assumptions concerning the type hierarchy in Turkish grammar. In Section 3.2, I proposed a way of dealing with the obligatoriness of accusative case-marking on 'inherently specific' NP objects in Turkish, using a *syntactic* SPECIFICITY feature appropriate for nouns and determiners only. Then in Section 3.3, I argued that possessors in Turkish should be treated as subjects, and proposed an HPSG analysis along these lines.

The main contribution of this chapter is Section 3.4, where I examine the issue of word order variation in Turkish, focussing mainly on the syntactic aspects of that variation. In that section, I presented a detailed characterization of the word order variation in simple and complex Turkish sentences, which may involve both local and long-distance scrambling. I argued that long-distance scrambling in Turkish is not only a matrix sentence phenomenon, in that it may as well occur *within* embedded clauses; that is, a constituent extracted out of an embedded clause may occur in certain syntactic positions of a surrounding clause which is itself an embedded clause. Furthermore, I outlined the assumptions I make, in HPSG, to deal with the "free" word order in Turkish, and proposed ways of capturing certain restrictions on local and long-distance scrambling, either in the form of LP constraints, in the usual way, or as constraints imposed on certain phrase types in the type hierarchy. I finished off the section by suggesting the use of pragmatic, as well as syntactic, notions in the LP constraints for Turkish, and proposed a (tentative) set of such constraints.

In the next chapter, I turn my attention to relativization in Turkish, and propose an account of the phenomenon, within the framework of HPSG.

Chapter 4

An HPSG Analysis of Relativization in Turkish

As we have seen in Chapter 1 (Section 1.3.1), relative clauses in Turkish are prenominal, and have verbal heads that are morphologically marked with participle suffixes. There are two different strategies of relativization in Turkish, distinguished by the morphological marking on the verbal head of the clause. Several accounts have been proposed in the literature which try to formulate the distribution of the two relativization strategies, all accounts so far being within the tradition of transformational grammar; for example, Underhill (1972), Hankamer and Knecht (1976), Dede (1978), Csató (1985), and Barker *et al.* (1990). (See Knecht (1979) for an overview of the first three accounts.) In this chapter, I propose a purely lexical account of the phenomenon within HPSG, which I claim is empirically more adequate than the previous accounts, as well as being computationally more attractive.

The first part of the chapter (Sections 4.1-4.3) focuses on empirical data, in an attempt to come up with an adequate descriptive account of relativization in Turkish, and also to characterize the nature of certain restrictions on relativization. The second part, Section 4.4, then proceeds to propose an HPSG analysis, relying on the observations of the first part.

4.1 Bounded Relativization

The two relativization strategies in Turkish have traditionally been called *subject participle (SPc)*, with the suffix *'-(y)En'*, and *object participle (OPc)*, with either of the suffixes

'-dIk' and *'-(y)EcEk'*, reflecting the correlation between the grammatical role of the relativized constituent and the choice of the relativization strategy (Knecht (1979), and Sezer (1986)).[1] This correlation is quite strong and seems to determine the choice of the strategy in cases like (4.1), with no long-distance dependency.

(4.1) a. Adam kadın-ı gör-dü. 'The man saw the woman.'
man woman-ACC see-PAST

b. [$\_i$ kadın-ı gör-en] adam$_i$ 'the man who saw the woman'
woman-ACC see-SPc man

c. [adam-ın $\_i$ gör-düğ-ü] kadın$_i$ 'the woman that the man saw'
man-GEN see-OPc-3sPoss woman

(4.1b) is a case of subject relativization and the SPc strategy is used, with the corresponding suffix on the verbal head *'gör'*. And (4.1c) exemplifies object relativization, with an OPc suffix on the verbal head as well as a possessive suffix which agrees, in person and number, with the genitive marked subject of the clause. This pattern of object relativization in fact applies to the relativization of any non-subject constituent, including adjuncts, as some of the forthcoming examples in this chapter will reveal.

There are nevertheless cases where non-subject constituents are relativized using the SPc, apparently violating the above generalization concerning the correlation between the grammatical function of the relativized constituent and the strategy to be used. Consider, for example, (4.2a), whose locative adjunct NP *'evde'* can be relativized using both the OPc, as seen in (4.2b), and the SPc, as seen in (4.2c).

(4.2) a. Her gece ev-de bir çocuk ağlı-yor.
every night house-LOC a child cry-PROG
'A child cries in the house every night.'

b. [her gece $\_i$ bir çocuğ-un ağla-dığ-ı] ev$_i$
every night a child-GEN cry-OPc-3sPoss house

c. [her gece $\_i$ bir çocuk ağla-yan] ev$_i$
every night a child cry-SPc house
'the house where a child cries every night'

[1] There are a number of other less common suffixes in both classes (SPc and OPc), which I won't consider here. Note also that I do not deal with 'headless relative clauses' here, briefly mentioned in Chapter 1 (Section 1.3.1).

It is important to note that (4.2b,c) have different interpretations, with specific and non-specific subject readings, respectively. This is due to the fact that in Turkish genitive marking on subjects of relative clauses (and also nominalization phrases) generally correlates with a specific reading of the subject, just as accusative marking on direct objects relates to a specific reading of the object, as discussed in Chapter 3 (Section 3.2).[2,3] Thus, one may argue that it is in fact the existence of a genitive marked subject in the clause that determines the particular relativization suffix on its verbal head (rather than the grammatical function of the relativized constituent), and that genitive subjects always induce the use of OPc, whereas the lack of such a subject results in the SPc. Further evidence supporting this argument comes from Turkish 'impersonal passives', that is, 'subjectless' constructions formed by passivized intransitive predicates, as in (4.3).

(4.3) Bu hava-da deniz-e gir-il-ir.
this weather-LOC sea-DAT enter-PASS-AOR
'This weather is good to swim in the sea.'

It was Hankamer and Knecht (1976) who first observed that relativization out of such constructions is only possible by using the SPc (although it is – naturally – always a non-subject constituent that is relativized). (4.4), for instance, shows that the dative object NP *'denize'*, and the locative adjunct NP *'bu havada'* in (4.3) above are both relativized using the SPc.

(4.4) a. [bu hava-da $\_{i}$ gir-il-en] $deniz_i$
this weather-LOC enter-PASS-SPc sea
'the sea that this weather is good to swim in'

b. [$\_{i}$ deniz-e gir-il-en] bu $hava_i$
sea-DAT enter-PASS-SPc this weather
'this weather which is good to swim in the sea'

It is clear from the discussion so far that in the case of bounded relativization, the existence/non-existence of a genitive-marked subject does play the main role in the choice of the relativization strategy in a clause. Next, I consider examples of long-distance relativization, before proposing an adequate pattern of relativization in Turkish.

[2]For the interested reader, Nilsson (1985) provides a thorough discussion of the function of case marking in Turkish.

[3]Consequently, exactly the same class of 'inherently specific' NPs mentioned in Chapter 3 need to be genitive case-marked when they appear as subjects of relative clauses (and nominalization phrases).

4.2 Long-distance Relativization

In Turkish relativization is also possible out of embedded phrases of certain kinds such as relative clauses, possessive phrases, postpositional phrases, nominalization phrases and non-subject infinitive phrases, resulting in structures with long-distance dependencies.

4.2.1 Relativization out of Relative Clauses

(4.5b) below is an example of relativization out of a relative clause, where the genitive subject, *'adamın'*, of the clause in (4.5a) has been relativized out of that clause, appearing as the head noun of a second relative clause surrounding the first one. The inner clause consequently has two gaps while the outer one has none. Note that in the outer clause, which has a genitive-marked subject, the OPc is used, and that the inner clause retains the OPc marker and the possessive suffix on its verbal head although it no longer has a genitive subject.

(4.5) a. Kadın [adam-ın $\_{i}$ oku-duğ-u] kitab-ı$_{i}$ bil-iyor.
woman man-GEN read-OPc-3sPoss book-ACC know-PROG
'The woman knows the book that the man is reading.'

b. [kadın-ın [$\_{j}$ $\_{i}$ oku-duğ-u] kitab-ı$_{i}$ bil-diğ-i] adam$_{j}$
woman-GEN read-OPc-3sPoss book-ACC know-OPc-3sPoss man
'the man that the woman knows the book he reads'

Consider now (4.6b), which is another example of relativization out of a relative clause. Note that in this case the accusative object of the clause in (4.6a) has been relativized, leaving again two gaps in the inner clause and leading to no further gaps in the outer one. The outer clause, as before, has a genitive subject and the OPc has been used.

(4.6) a. Kadın [$\_{i}$ kitab-ı oku-yan] adam-ı$_{i}$ gör-dü.
woman book-ACC read-SPc man-ACC see-PAST
'The woman saw the man who was reading the book.'

b. [kadın-ın [$\_{i}$ $\_{j}$ oku-yan] adam-ı$_{i}$ gör-düğ-ü] kitap$_{j}$
woman-GEN read-SPc man-ACC see-OPc-3sPoss book
'the book such that the woman saw the man who was reading it'

There are three points to note about these two examples. First, the relativization of a second constituent out of the inner clause hasn't changed the strategy used in that clause before, in either of these examples (leaving it as OPc in (4.5b) and SPc in (4.6b)). Second, the choice of the strategy in the outer clause hasn't been affected by the strategy used in the inner clause (resulting in OPc in the outer clause in both cases). And finally, neither has it been affected by the grammatical function of the constituent that has been long-distance relativized (i.e. subject in (4.5b) and object in (4.6b)).

Let us now consider the two examples of long-distance relativization given in (4.7b) and (4.8b). Note that in neither of these examples has the outer clause a genitive subject, and also that the strategy used in the outer clause in both cases is SPc. Note further that all three observations made above for (4.5) and (4.6) also hold true for (4.7) and (4.8).

(4.7) a. [$\__i$ Bitki-yi yi-yen] insan-lar-da$_i$ alerji tespit ed-il-di.
plant-ACC eat-SPc person-PLU-LOC allergy determine-PASS-PAST
'Allergy was diagnosed in the people who ate the plant.'

b. [[$\__i$ $\__j$ yi-yen] insan-lar-da$_i$ alerji tespit ed-il-en] bitki$_j$
eat-SPc person-PLU-LOC allergy determine-PASS-SPc plant
'the plant which allergy was diagnosed in the people who ate (it)'

(4.8) a. [İnsan-lar-ın $\__i$ ye-dik-leri] bitki-de$_i$ zehir tespit ed-il-di.
person-PLU-GEN eat-OPc-3pPoss plant-LOC poison detect-PASS-PAST
'Poison was detected in the plant that the people ate.'

b. [[$\__j$ $\__i$ ye-dik-leri] bitki-de$_i$ zehir tespit ed-il-en] insan-lar$_j$
eat-OPc-3pPoss plant-LOC poison detect-PASS-SPc person-PLU
'the people that poison was detected in the plant they ate'

Examples (4.5)-(4.8) all seem to support the previous claim on the correlation between the existence of a genitive-marked subject and the choice of the relativization strategy in the (outer) clause. Next, I provide some further examples which suggest that the choice of the relativization strategy in the outer clause is in fact constrained by an additional factor in certain cases of long-distance relativization.

The first case is exemplified in (4.9). The genitive subject, *'kadının'*, of the relative clause in (4.9a) has been relativized in (4.9b) using the SPc in the outer clause. The point to note about this case is that the genitive marking on the subject in the outer clause and the use of OPc strategy is ruled ungrammatical, as shown in (4.9c), although the subject (*'child'*) *does* have a specific reading.

(4.9) a. [Kadın-ın $\_{i}$ sev-diğ-i] çocuk$_i$ ağla-ma-ya başla-dı.
woman-GEN cuddle-OPc-3sPoss child cry-ACT-DAT start-PAST
'The child that the woman was cuddling started crying.'

b. [[$\_{j}$ $\_{i}$ sev-diğ-i] çocuk$_i$ ağla-ma-ya başla-yan] kadın$_j$
cuddle-OPc-3sPoss child cry-ACT-DAT start-SPc woman
'the woman who the child that she was cuddling started crying'

c. * [[$\_{j}$ $\_{i}$ sev-diğ-i] çocuğ-un$_i$ ağla-ma-ya başla-dığ-ı] kadın$_j$
cuddle-OPc-3sPoss child-GEN cry-ACT-DAT start-OPc-3sPoss woman

The second case is exemplified in (4.10b), where the accusative object, *'uçağı'*, of (4.10a) has been relativized using the OPc in the outer clause, together with a genitive marking on the subject of that clause. Observe that (4.10c), which lacks a genitive marking on the subject and makes use of the SPc in the outer clause, is ungrammatical although the subject in this case is no further specific than the (nominative) one in (4.9b) above, and hence its obligatory genitive-marking doesn't seem to be due to semantic reasons.

(4.10) a. [$\_{i}$ Uçağ-ı kullan-an] pilot$_i$ çıldır-dı.
plane-ACC fly-SPc pilot go crazy-PAST
'The pilot who was flying the plane went crazy.'

b. [[$\_{i}$ $\_{j}$ kullan-an] pilot-un$_i$ çıldır-dığ-ı] uçak$_j$
fly-SPc pilot-GEN go crazy-OPc-3sPoss plane
'the plane which the pilot who was flying it went crazy'

c. * [[$\_{i}$ $\_{j}$ kullan-an] pilot$_i$ çıldır-an] uçak$_j$
fly-SPc pilot go crazy-SPc plane

I claim that the choice of the relativization strategy in the outer clause in these two examples is determined by the grammatical function of the gap in the inner clause which corresponds to the head noun of the outer clause. More specifically, I claim that the strategy used in the outer clause is SPc if that grammatical function is subject, as in (4.9b), and it is OPc otherwise, as in (4.10b).

What then is the difference between the (b) examples in (4.5)-(4.8) and the ones in (4.9)-(4.10) that makes the choice of the relativization strategy in the outer clause rely on different factors? Note that in all cases of long-distance relativization, the inner clause is a part (modifier in the examples so far) of one of the constituents of the outer clause. Hereafter, I refer to that constituent of the outer clause as the 'gap host'.[4] Note further

[4]I borrow the term 'gap host' from Barker *et al.* (1990). They define a gap host as the highest *nominal* in the relative clause dominating the gap. However, I use it in a broader sense here which also includes postpositional phrases, nominalization phrases and infinitive phrases (as will become clear later in the chapter).

that in all the (b) examples in (4.5)-(4.8), the gap host is a non-subject constituent of the outer clause (i.e. an accusative object in (4.5b) and (4.6b), and a locative adjunct in (4.7b) and (4.8b)), whereas in (4.9b) and (4.10b), the gap host is the subject of the outer clause. It is exactly this difference in the grammatical function of the gap host, I claim, that determines which one of the two factors mentioned above plays a role in the choice of the strategy in the outer clause (i.e. the existence of a genitive-marked subject in the outer clause, as in (4.5b)-(4.8b), where the gap host is a non-subject constituent, or the grammatical function of the gap,[5] as in (4.9b) and (4.10b), where the gap host is the subject.

One can then formalize the above discussion on the choice of the relativization strategy in long-distance relativization in the following way:[6]

(4.11) **Long-distance relativization pattern in Turkish:**

(a) if the gap host is a *non-subject* constituent then
  (i) if there is a *genitive* subject in the clause[7] then
      the OPc strategy is used
  (ii) else
      the SPc strategy is used
(b) else if the gap host is the *subject* then
  (i) if the grammatical role of the gap is *subject* then
      the SPc strategy is used
  (ii) else
      the OPc strategy is used

So, for example, in (4.5b)-(4.8b), the gap host is a non-subject constituent of the outer clause, and the OPc is used in the outer clause in (4.5b) and (4.6b), since there is a genitive subject in that clause (cf. (4.11ai)), and the SPc is used in (4.7b) and (4.8b), since there is no such subject (cf. (4.11aii)). On the other hand, in both (4.9b) and (4.10b) the gap host is the subject of the outer clause, and the SPc is used in (4.9b), since the

[5]In the rest of the chapter, in cases with more than one gap, the word 'gap' always refers to the gap that corresponds to the long-distance relativized constituent (that is, the head noun of the outer clause).

[6]I must note that this pattern is based on a compilation of grammaticality judgements of 12 native Turkish speakers on 60 examples of Turkish relative clauses with long-distance dependencies.

[7]Note that Turkish is a pro-drop language, and that the genitive subject in this case does not need to be an overt one.

grammatical function of the gap is subject (cf. (4.11bi)) – and furthermore the use of OPc is ruled out as shown in (4.9c) – and the OPc is used in (4.10b), since the gap is a non-subject constituent of the inner clause, that is, the accusative object (cf. (4.11bii)) – and the use of SPc is ruled out as shown in (4.10c).[8]

Until now, there have been two main (independent) proposals in the literature as to what determines the relativization strategy in the outer clause in the case of long-distance relativization in Turkish: i) the grammatical function of the gap (e.g. Csató (1985)), and ii) the grammatical function of the gap host (e.g. Barker *et al.* (1990)). Notice that (4.11) takes both these factors into account as well as a third one, namely the existence of a genitive-marked subject in the outer clause. It may prove useful to make a comparison between (4.11) and the account suggested by Barker *et al.* (1990). (4.11a), where the gap host is a non-subject constituent, is quite straightforward and is in line with the account by Barker *et al.* (1990), except they analyze clauses with nominative subjects as *subjectless* (just like impersonal passives), claiming that such subjects undergo 'subject incorporation'. As for (4.11b), where the gap host is subject, I disagree with Barker *et al.* (1990) on empirical grounds. They claim that there are two dialects with respect to the distribution of the OPc. In one of the dialects (their Dialect A) the OPc is ruled out in this case, hence the SPc is the only strategy to use, whatever the grammatical role of the gap is. In the other dialect (Dialect B), however, both strategies can be used again independent of the grammatical role of the gap. The grammaticality judgements of my informants (cf. fn. 6 above) have led me to reject the claim that the SPc can be used in this case when the grammatical role of the gap is non-subject (except for the cases in which the gap host is a nominalization phrase as I further discuss in Section 4.2.3). Turning to the possibility of the OPc when the grammatical role of the gap is subject, I have encountered a number of judgements in favour of this. I do not however see myself in the position of claiming the existence of two different dialects with respect to this particular case only, since those judgements are outnumbered by the judgements from the very same speakers on structurally similar examples that rule out the use of OPc in this case.

Note that (4.11) can further be generalized to cover the cases of bounded relativization as well. The only crucial point in this case is to assume that the gap coincides with the gap host. The case in (4.11bii) would then never arise since the gap/gap host cannot

[8]One may argue that what renders (4.10c) ungrammatical is in fact the fact that the dependencies between the gaps and the corresponding head nouns is intersecting in this case (cf. the Nested Dependency Constraint of Fodor (1978); see fn. 11 in Chapter 5, page 162). Notice however that both (4.6b) and (4.10b) are grammatical although they exhibit the same kind of intersecting dependencies.

both be the subject and a non-subject constituent at the same time. Hence, if the gap is the subject then only the SPc can be used (cf. (4.11bi)); and if the gap host is a non-subject constituent then either the OPc or the SPc can be used depending on whether there is a genitive subject in the clause or not (cf. (4.11ai) and (4.11aii), respectively). In the case of impersonal passives, (4.11aii) is the only case that arises (since there is no subject in the clause), and hence the SPc is the only possible strategy – in line with the empirical facts.

In this section, I presented several examples of relativization out of relative clauses in Turkish, and proposed a relativization pattern, (4.11), that covers all the examples of bounded and long-distance relativization considered so far. In the following sections, I turn to the cases of relativization out of possessive NPs, nominalization and infinitive phrases, and show that they are also in line with the predictions made by (4.11).

4.2.2 Relativization out of Possessive Phrases

(4.12b) and (4.13b) below are examples of relativization out of possessive phrases, where the genitive possessor, *'adamın'*, of the corresponding (a) example in each case has been relativized, appearing as the head noun of the relative clause surrounding the possessive phrase. Note that in (4.12b) there is a genitive subject, *'senin'*, in the clause and the OPc is used (cf. (4.11ai)), and in (4.13b) the subject is nominative, *'arı'*, and the SPc is used (cf. (4.11aii)).

(4.12) a. Sen [adam-ın kız-ı-nı] gör-dü-n.
you man-GEN daughter-3sPoss-ACC see-PAST-2SG
'You saw the man's daughter.'

b. [sen-in [$\_i$ kız-ı-nı] gör-düğ-ün] adam$_i$
you-GEN daughter-3sPoss-ACC see-OPc-2sPoss man
'the man whose daughter you saw'

(4.13) a. [Adam-ın bacağ-ı-nı] arı sok-tu.
man-GEN leg-3sPoss-ACC bee sting-PAST
'Some bee/bees stung the man's leg.'

b. [[$\_i$ bacağ-ı-nı] arı sok-an] adam$_i$
leg-3sPoss-ACC bee sting-SPc man
'the man whose leg some bee/bees stung'

To cover the case of possessor relativization out of subject possessive phrases using the relativization pattern in (4.11), I rely on my argument in Chapter 3 (Section 3.3) that possessors in Turkish should treated as subjects. Hence, in (4.14) (4.11bi) predicts the use of SPc (and rules out the OPc), which is indeed the case as seen in (4.14b) (and in (4.14c)).

(4.14) a. [Adam-ın kız-ı] sen-i gör-dü.
man-GEN daughter-3sPoss you-ACC see-PAST
'The man's daughter saw you.'

b. [[$\_{i}$ kız-ı] sen-i gör-en] adam$_i$
daughter-3sPoss you-ACC see-SPc man
'the man whose daughter saw you'

c. * [[$\_{i}$ kız-ı-nın] sen-i gör-düğ-ü] adam$_i$
daughter-3sPoss-GEN you-ACC see-OPc-3sPoss man

4.2.3 Relativization out of Nominalization Phrases

All four cases considered in (4.11) arise in the case of relativization out of nominalization phrases, since relativization of both subject and non-subject constituents is possible out of both subject and non-subject nominalization phrases.

Let us first consider the examples of relativization out of non-subject nominalization phrases given in (4.15) and (4.16). Note that independent of the grammatical function of the gap (which is subject in the (b) examples and accusative object in the (c) examples), the relativization strategy used is OPc in (4.15b,c), where there is a genitive subject in the relative clause, and SPc in (4.16b,c), where the subject is nominative, as predicted by (4.11ai) and (4.11aii), respectively.

(4.15) a. [Başbakan-ın bu söz-ü söyle-diğ-i-ni] gazete
prime minister-GEN this word-ACC say-FACT-3sPoss-ACC newspaper
yaz-dı.
write-PAST
'The newspaper reported that the prime minister said these words.'

b. [[$\_{i}$ bu söz-ü söyle-diğ-i-ni] gazete-nin yaz-dığ-ı]
this word-ACC say-FACT-3sPoss-ACC newspaper-GEN write-OPc-3sPoss
başbakan$_i$
prime minister
'the prime minister who the newspaper reported to have said these words'

c.

[[başbakan-ın	$\_{i}$	söyle-diğ-i-ni]	gazete-nin
prime minister-GEN		say-FACT-3sPoss-ACC	newspaper-GEN

yaz-dığ-ı]	bu	söz$_i$
write-OPc-3sPoss	this	word

'these words which the newspaper reported the prime minister said'

(4.16) a.

[Başbakan-ın	bu	söz-ü	söyle-diğ-i-ni]	gazete
prime minister-GEN	this	word-ACC	say-FACT-3sPoss-ACC	newspaper

yaz-dı.
write-PAST
'Some newspaper/newspapers reported that the prime minister said these words.'

b.

[[$\_{i}$	bu	söz-ü	söyle-diğ-i-ni]	gazete	yaz-an]
	this	word-ACC	say-FACT-3sPoss-ACC	newspaper	write-SPc

başbakan$_i$
prime minister
'the prime minister such that some newspaper/newspapers reported that he said these words'

c.

[[başbakan-ın	$\_{i}$	söyle-diğ-i-ni]	gazete	yaz-an]	bu
prime minister-GEN		say-FACT-3sPoss-ACC	newspaper	write-SPc	this

söz$_i$
word
'these words which some newspaper/newspapers reported the prime minister said'

Relativization out of subject nominalization phrases is exemplified by (4.17) and (4.18) below. (4.17b,c) reveal that in the case of subject relativization out of a subject nominalization phrase, the strategy to be used is SPc and that the use of OPc is ruled out, as is predicted by (4.11bi). The case of non-subject relativization out of such phrases, however, constitutes a problem for the relativization pattern in (4.11). Recall that (4.11bii) predicts the use of OPc in this case, and rules out the SPc. Yet, only some of my informants have found (4.18b) (and similar examples), with the OPc, grammatical, whereas all of them have agreed that (4.18c) (and similar examples), with the SPc, is perfectly grammatical.[9]

[9]Recall from page 88 that according to Barker *et al.* (1990), (4.18c) would be grammatical in both Dialect A and Dialect B, while (4.18b) would be grammatical only in Dialect B. Hence, their account makes the correct predictions in this particular case.

(4.17) a. [Adam-ın kadın-ı tanı-ma-sı] bekle-n-iyor.
man-GEN woman-ACC know-ACT-3sPoss expect-PASS-PROG
'It is expected that the man knows the woman.'

b. [[$\_i$ kadın-ı tanı-ma-sı] bekle-n-en] adam$_i$
woman-ACC know-ACT-3sPoss expect-PASS-SPc man
'the man who is expected to know the woman'

c. * [[$\_i$ kadın-ı tanı-ma-sı-nın] bekle-n-diğ-i] adam$_i$
woman-ACC know-ACT-3sPoss-GEN expect-PASS-OPc-3sPoss man

(4.18) a. [Bu gösteri-ye 500 kişi-nin katıl-ma-sı]
this demonstration-DAT person-GEN participate-ACT-3sPoss
bekle-n-iyor.
expect-PASS-PROG
'It is expected that 500 people will participate in this demonstration.'

b. ? [[$\_i$ 500 kişi-nin katıl-ma-sı-nın] bekle-n-diğ-i]
person-GEN participate-ACT-3sPoss-GEN expect-PASS-OPc-3sPoss
bu gösteri$_i$
this demonstration
'this demonstration in which it is expected that 500 people will participate'

c. [[$\_i$ 500 kişi-nin katıl-ma-sı] bekle-n-en] bu
person-GEN participate-ACT-3sPoss expect-PASS-SPc this
gösteri$_i$
demonstration
'this demonstration in which it is expected that 500 people will participate'

4.2.4 Relativization out of Postpositional Phrases

Recall from Chapter 1 (Section 1.3.1) that certain postpositions in Turkish form PPs that are structurally similar to possessive NPs. The object of the postposition is either genitive marked or unmarked (i.e. nominative), and the postposition *contains* a possessive suffix that agrees (in person and number) with its object and a case suffix (that is fixed for each postposition), as exemplified in (4.19) below.

(4.19) a. Bu iş [adam taraf-ı-ndan] yap-ıl-dı.
this job man by(-3sPoss-ABL) do-PASS-PAST
'This job was done by the man.'

b. Biz [çocuk hakk-ı-nda] konuş-tu-k.
we child about(-3sPoss-LOC) talk-PAST-1PL
'We have talked about the child.'

Certain other postpositions form PPs that have a rather simpler structure, in that although the object is again either genitive or nominative, the postposition appears without any agreement or case suffix, as seen in (4.20).

(4.20) Berfu [diğer çocuk-lar gibi] okul-a git-ti.
Berfu other child-PLU like school-DAT go-PAST
'Berfu has gone to school like the other children.'

Only the postpositions that bear agreement morphology let their object be relativized, as seen in (4.21). Note that in both (4.21a,b) the OPc is used, as predicted by (4.11ai).

(4.21) a. [bu iş-in [$\_{i}$ taraf-ı-ndan] yap-ıl-dığ-ı] adam$_i$
this job-GEN by(-3sPoss-ABL) do-PASS-OPc-3sPoss man
'the man by whom the job was done'

b. [biz-im [$\_{i}$ hakk-ı-nda] konuş-tuğ-umuz] çocuk$_i$
we-GEN about(-3sPoss-LOC) talk-OPc-1pPoss child
'the child whom we have talked about'

c. * [Berfu-nun [$\_{i}$ gibi] okul-a git-tiğ-i] diğer çocuk-lar$_i$
Berfu like school-DAT go-OPc-3sPoss other child-GEN
'the other children like whom Berfu has gone to school'

4.2.5 Relativization out of Infinitive Phrases

Sezer (1986) observes that relativization is possible also out of infinitive phrases in Turkish, but only non-subject ones. Thus, for example, it is not possible to relativize the accusative object, *'çocuğu'*, of the subject infinitive phrase in (4.22a) (cf. (4.22b)), while this is perfectly possible in (4.23a), where the same infinitive phrase acts as an object of the subject equi verb *'iste'*. Note the genitive subject *'kadının'*, and the use of OPc in (4.23b) (cf. (4.11ai)).

(4.22) a. [Çocuğ-u gör-mek] kadın-ı sevin-dir-di.
child-ACC see-INF woman-ACC be happy-CAUS-PAST
'To see the child made the woman happy.'

b. * [[$\_{i}$ gör-mek] kadın-ı sevin-dir-en] çocuk$_i$
see-INF woman-ACC be happy-CAUS-SPc child

(4.23) a. Kadın [çocuğ-u gör-mek] iste-di.
woman child-ACC see-INF want-PAST
'The woman wanted to see the child.'

b. [kadın-ın [$\_i$ gör-mek] iste-diğ-i] çocuk$_i$
woman-GEN see-INF want-OPc-3sPoss child
'the child that the woman wanted to see'

So far, I have been concerned with proposing a descriptive account of relativization in Turkish that correctly characterizes the distribution of the two relativization strategies. The next section discusses a number of restrictions on relativization in Turkish.

4.3 Restrictions on Relativization in Turkish

We saw in Section 4.2.5 that relativization is not possible out of subject infinitive phrases in Turkish. A number of further restrictions are discussed below.

4.3.1 Restriction on Multiple Non-subject Relativization

In Section 4.2, we considered several examples of relativization in Turkish with two constituents relativized out of the same clause (i.e. either the same relative clause as in Section 4.2.1 or the same nominalization phrase as in Section 4.2.3). Note that in all those examples, one of the relativized constituents was the subject of the clause, and the other a non-subject constituent. We haven't, in other words, seen any examples with more than one non-subject gap in the same clause, and in fact it turns out that this is not possible in Turkish.[10] Consider, for example, (4.24) and (4.25) below, which show that there is such a restriction in the case of relativization out of relative clauses.[11]

(4.24) a. Ben [çocuğ-un okul-dan $\_i$ git-tiğ-i] ev-i$_i$ gör-dü-m.
I child-GEN school-ABL go-OPc-3sPoss house-ACC see-PAST-1SG
'I saw the house where the child went from the school.'

b. * [ben-im [çocuğ-un $\_j$ $\_i$ git-tiğ-i] ev-i$_i$ gör-düğ-üm] okul$_j$
I-GEN child-GEN go-OPc-3sPoss house-ACC see-OPc-1sPoss school
'the school such that I saw the house where the child went from there'

[10] Note however that it is in general possible to extract two non-subject constituents out of the same clause in the case of other UDCs, such as topicalization and backgrounding, in Turkish.

[11] The same restriction also holds true for other types of non-finite sentences where object relativization is possible, such as nominalization and infinitive phrases.

(4.25) a. [Adam-ın kitab-ı $\_{i}$ oku-duğ-u] ev$_i$ yan-ıyor.
man-GEN book-ACC read-OPc-3sPoss house burn-PROG
'The house where the man is reading the book is on fire.'

b. * [[adam-ın $\_{i}$ $\_{j}$ oku-duğ-u] ev-in$_i$ yan-dığ-ı] kitap$_j$
man-GEN read-OPc-3sPoss house-GEN burn-OPc-3sPoss book
'the book such that the house where the man is reading it is on fire'

c. * [[adam-ın $\_{i}$ $\_{j}$ oku-duğ-u] ev$_i$ yan-an] kitap$_j$
man-GEN read-OPc-3sPoss house burn-SPc book
'the book such that the house where the man is reading it is on fire'

The relative clause in (4.24a) already has a dative object gap, and the relativization of the second object, the ablative NP *'okuldan'*, is blocked as shown in (4.24b).[12] Similarly, the relativization of the accusative NP *'kitabı'* out of the relative clause in (4.25a), which has already had its locative adjunct relativized, is blocked, whichever strategy is used (cf. (4.25b,c)).

4.3.2 Restriction on Relativization of Nominative Subjects of Non-finite Sentences

In Section 4.1, we saw that subjects of relative clauses (and nominalization phrases) in Turkish can be genitive marked or unmarked, and that genitive marking usually correlates with a specific reading of the subject. It should be noted that nominative subjects of such clauses cannot be relativized. So, for example, the relativization of the nominative subject, *'arı'*, of the relative clause in (4.26a) is blocked, whichever strategy is used (cf. (4.26b,c)).

(4.26) a. [[$\_{i}$ Bacağ-ı-nı] arı sok-an] kız$_i$ ağla-dı.
leg-3sPoss-ACC bee sting-SPc girl cry-PAST
'The girl whose leg was stung by a bee cried.'

b. * [[[$\_{i}$ bacağ-ı-nı] $\_{j}$ sok-an] kız$_i$ ağla-yan] arı$_j$
leg-3sPoss-ACC sting-SPc girl cry-SPc bee
'some bee such that the girl whose leg was stung by it cried'

c. * [[[$\_{i}$ bacağ-ı-nı] $\_{j}$ sok-an] kız-ın$_i$ ağla-dığ-ı] arı$_j$
leg-3sPoss-ACC sting-SPc girl-GEN cry-OPc-3sPoss bee
'some bee such that the girl whose leg was stung by it cried'

[12] Contrast (4.24b) with (i) below, which is perfectly grammatical.

(i) [ben-im [çocuğ-un $\_{i}$ git-tiğ-i] ev-i$_i$ $\_{j}$ gör-düğ-üm] okul$_j$
I-GEN child-GEN go-OPc-3sPoss house-ACC see-OPc-1sPoss school
'the school from where I saw the house that the child went'

4.3.3 Restriction on Relativization across Finite Sentences

No relativized constituent in Turkish can cross the boundaries of a *finite* sentence, whereas an extracted (e.g. topicalized or backgrounded) constituent can. In (4.27a), for example, the S[*fin*] complement of *'sandı'* contains an embedded S[*fact*] complement, whose accusative object can be long-distance topicalized as seen in (4.27b), but not be relativized, cf. (4.27c).

(4.27) a. Ben [$_{S[fin]}$ adam-a kadın [$_{S[fact]}$ çocuğ-un kitab-ı
I man-DAT woman child-GEN book-ACC
oku-duğ-u-nu] söyle-di] san-dı-m.
read-FACT-3sPoss-ACC tell-PAST think-PAST-1SG
'I thought that the woman told the man that the child had read the book.'

b. Kitab-ı$_i$ [ben [$_{S[fin]}$ adam-a kadın [$_{S[fact]}$ çocuğ-un $\_i$
book-ACC I man-DAT woman child-GEN
oku-duğ-u-nu] söyle-di] san-dı-m.]
read-FACT-3sPoss-ACC tell-PAST think-PAST-1SG
'As for the book, I thought that the woman told the man that the child had read it.'

c. * [$_{S[part]}$ ben-im [$_{S[fin]}$ adam-a kadın [$_{S[fact]}$ çocuğ-un $\_i$
I-GEN man-DAT woman child-GEN
oku-duğ-u-nu] söyle-di] san-dığ-ım] kitap$_i$
read-FACT-3sPoss-ACC tell-PAST think-OPc-1sPoss book
'the book that I thought that the woman told the man that the child had read'

Since we know that relativization is in general possible out of nominalization phrases, the most likely explanation for the contrast between the grammaticality of (4.27b) and (4.27c) is that crossing of relativized constituents across the boundaries of finite sentences is blocked in Turkish. Only to make sure, let us also consider (4.28), where that S[*fin*] complement has been replaced by an S[*fact*] one. Note that *'kitabı'* in this case can *indeed* be both topicalized and relativized, as seen in (4.28b,c), respectively.

(4.28) a. Ben [$_{S[fact]}$ adam-a kadın-ın [$_{S[fact]}$ çocuğ-un kitab-ı
I man-DAT woman-GEN child-GEN book-ACC
oku-duğ-u-nu] söyle-diğ-i-ni] san-dı-m.
read-FACT-3sPoss-ACC tell-FACT-3sPoss-ACC think-PAST-1SG
'I thought that the woman told the man that the child had read the book.'

b. Kitab-ı$_i$ [ben [$_{S[fact]}$ adam-a kadın-ın [$_{S[fact]}$ çocuğ-un $\_i$
book-ACC I man-DAT woman-GEN child-GEN
oku-duğ-u-nu] söyle-diğ-i-ni] san-dı-m.]
read-FACT-3sPoss-ACC tell-FACT-3sPoss-ACC think-PAST-1SG
'As for the book, I thought that the woman told the man that the child had read it.'

c. [$_{S[part]}$ ben-im [$_{S[fact]}$ adam-a kadın-ın [$_{S[fact]}$ çocuğ-un $\_i$
I-GEN man-DAT woman-GEN child-GEN
oku-duğ-u-nu] söyle-diğ-i-ni] san-dığ-ım] kitap$_i$
read-FACT-3sPoss-ACC tell-FACT-3sPoss-ACC think-OPc-1sPoss book
'the book that I thought that the woman told the man that the child had read'

4.4 An Analysis within HPSG

This section is concerned with providing an HPSG analysis of relativization in Turkish, taking into account both the relativization pattern proposed in Section 4.2, and also the restrictions on relativization discussed in Section 4.3.[13] I start out by emphasizing the important characteristics of Turkish relative clauses that shape the main features of any prospective HPSG analysis.

The first point to note is that Turkish relative clauses have verbal heads with identifying morphology, which gives one the opportunity to specify the MOD value of a relative clause, in the lexical entry of its verbal head that would then be passed on to the clause via the HFP.[14]

The second point is related to the status of the NONLOCAL feature RELATIVE (REL) in standard HPSG, in the case of Turkish. The function of REL is to encode the relative dependency in a relative clause, that is, the dependency between the relative word and the head noun with which it shares an index (see Pollard and Sag (1994)[pages 210–220] for details). Note however, from the examples we have seen so far, that there is no relative pronoun in Turkish. Hence, I assume that there is no need to use the REL feature in the analysis of Turkish relative clauses, since there is no such dependency in Turkish.

[13]Note that the analysis presented here deals with only argument relativization, leaving relativization of adjuncts for further consideration.

[14]The idea of having such an analysis for languages like Korean, where the verbal head of a relative clause bears identifying morphology, is suggested by Pollard and Sag (1994)[page 57].

Third, recall from Section 4.3.3 that no relativized constituent in Turkish can cross the boundaries of a *finite* sentence, while an extracted (topicalized or backgrounded) constituent can. It therefore makes sense to use two separate NONLOCAL features in the analyses of relativization and other kinds of unbounded dependencies, since that would let the above restriction be readily formalized as a constraint on only one of those features of any *finite* sentence. One can then introduce a new NONLOCAL feature, say RELATIVIZED, in addition to SLASH, that takes values of type *set(local)*, and further assume the following parochial constraint for Turkish, which simply states that any *phrase* object with an S[*fin*] SYNSEM value must have an empty INHER|RELATIVIZED value:

(4.29) Finite Sentence Relativized Constraint (preliminary version):

$$\begin{bmatrix} phrase \\ \text{SYNSEM} \quad \text{S}[fin] \end{bmatrix} \Rightarrow \begin{bmatrix} \text{INHER|RELATIVIZED} \quad \{\} \end{bmatrix}$$

Finally, it is clear from the previous sections that Turkish relative clauses should be analysed as *weak* UDCs, requiring the gap and the head noun structure-share only their INDEX values (since they need not, for example, have the same case, as can be seen from the examples so far).

Having highlighted these general points, I now turn to the particular HPSG account that I propose here for relativization in Turkish, and outline the main features of the analysis (in addition to the ones mentioned above).

Note that in all examples of relative clauses where a *bounded* dependency is bound off, the head noun of the relative clause corresponds to (i.e. is coindexed with) a gap in the clause itself. On the other hand, in clauses where a *long-distance* dependency is bound off, the head noun doesn't correspond to a gap in the clause itself, but rather to one that is embedded somewhere within one of the constituents of the clause (i.e. the gap host). I make a distinction, in the analysis, between the *part* verbs[15] that head relative clauses of those two kinds. To that end, I make use of two separate sets of lexical rules to derive those two kinds of *part* verbs from *base* verbs in the lexicon. The first set of rules – ignoring the details for the time being – can be schematized as follows:

[15] Recall from Chapter 3 (Section 3.1) that *part* is assumed to be a subtype of the type *vform* in Turkish grammar.

(4.30)
$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{HEAD} & verb\begin{bmatrix}\text{VFORM} & bse \\ \text{MOD} & none\end{bmatrix} \\ \text{F} & \langle \ldots, \text{NP}[\text{LOC}\,\boxed{2}], \ldots\rangle \\ \text{CONTENT} & \boxed{3}\end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet <\textit{part-suffix}> \\ \text{HEAD} & verb\begin{bmatrix}\text{VFORM} & part \\ \text{MOD} & \text{NP}\left[\text{TO-BIND}|\text{RELATIVIZED}\left\{\boxed{2}_{\boxed{4}}\right\}\right] : \begin{bmatrix}\text{INDEX} & \boxed{4} \\ \text{RESTR} & \boxed{5}\end{bmatrix}\end{bmatrix} \\ \text{F} & \langle \ldots \rangle \\ \text{CONTENT} & nom\text{-}obj\begin{bmatrix}\text{INDEX} & \boxed{4} \\ \text{RESTR} & \{\boxed{3}\} \cup \boxed{5}\end{bmatrix} \\ \text{INHER}|\text{RELATIVIZED} & \{\boxed{2}\}\end{bmatrix}$$

where $\text{F} \in \{\text{SUBJ}, \text{COMPS}\}$

(4.30) derives a *part* verb from a *base* verb, affixing one of the participle suffixes to the PHON value of the *base* verb. Note that it also changes the MOD value of the verb from *none* to an NP (cf. the paragraph above on lexically specified MOD values), as well as 'relativizing' one of its arguments, that is, removing it from one of the valence lists (i.e. SUBJ or COMPS), and placing its LOCAL value within the INHER|RELATIVIZED set of the output.[16] It also changes the CONTENT value of the input entry (which is of type *psoa*) to an object of type *nom-object* with a restricted INDEX structure-shared with that of the NP being modified, and the restriction set (the RESTR value) being determined by adding the CONTENT value of the input verb to the restrictions imposed by the modified NP. Futhermore, the TO-BIND|RELATIVIZED value of the NP is constrained to have a single element structure-shared with the only element in the INHER|RELATIVIZED value of the *part* verb itself, to ensure the discharge of this dependency in a head-adjunct phrase with a relative clause headed by this *part* verb. Finally, the coindexation of the modified NP and the element in its TO-BIND|RELATIVIZED value, hence the element in the INHER|RELATIVIZED value of the *part* verb, guarantees that the NP will be assigned

[16]The idea behind these rules is similar, in this respect, to that behind the extraction lexical rules proposed by Pollard and Sag (1994)[pages 376–384] in the traceless account of UDCs; see Chapter 2, page 22, of this dissertation.

the same semantic role that is assigned to the argument that is relativized by this rule, within the CONTENT value of the input *base* verb.

The second set of rules mentioned above can be schematized as in (4.31). The main difference between (4.30) and (4.31) is that the latter does not relativize any of the arguments of the input, but instead constrains one of its arguments (i.e. the one to function as the gap host) to have a non-empty INHER|RELATIVIZED value. Accordingly, it is an element of this INHER|RELATIVIZED value that is structure-shared with the only one in the TO-BIND|RELATIVIZED value of the modified NP. (Note that there may be other dependencies stored in the INHER|RELATIVIZED value of the gap host, which would then be passed on to the mother NP of the relative clause headed by the *part* verb, via the NFP in the usual way.)

(4.31)

$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{HEAD} & verb\begin{bmatrix} \text{VFORM} & bse \\ \text{MOD} & none \end{bmatrix} \\ \text{F} & \langle \ldots, \text{XP}, \ldots \rangle \\ \text{CONTENT} & \boxed{2} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{part-suffix} > \\ \text{HEAD} & verb\begin{bmatrix} \text{VFORM} & part \\ \text{MOD} & \text{NP}\left[\text{TO-BIND|RELATIVIZED}\left\{\boxed{5}_{\boxed{3}}\right\}\right] : \begin{bmatrix} \text{INDEX} & \boxed{3} \\ \text{RESTR} & \boxed{4} \end{bmatrix} \end{bmatrix} \\ \text{F} & \langle \ldots, \text{XP}\,[\text{INHER|RELATIVIZED}\ \{\ldots, \boxed{5}, \ldots\}], \ldots \rangle \\ \text{CONTENT} & \textit{nom-obj}\begin{bmatrix} \text{INDEX} & \boxed{3} \\ \text{RESTR} & \{\boxed{2}\} \cup \boxed{4} \end{bmatrix} \end{bmatrix}$$

where $\text{F} \in \{\text{SUBJ}, \text{COMPS}\}$ and
$\text{XP} \in \{\ \text{NP}, \text{PP}, \text{S}[\textit{nomin}], \text{VP}[\textit{inf}]\ \}$

In addition to these two sets of lexical rules, there is one more set, schematized in (4.32), that deals with argument relativization out of embedded phrases of the kinds discussed in Section 4.2, by simply relativizing one of the arguments of the input entry, which is to function as the head of such an embedded phrase.

(4.32)
$$\begin{bmatrix} \text{F} & \langle \ldots, \text{NP}[\text{LOC}\,\boxed{1}], \ldots \rangle \\ \text{INHER}|\text{RELATIVIZED} & \boxed{2} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{F} & \langle \ldots \rangle \\ \text{INHER}|\text{RELATIVIZED} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

where F $\in$ {SUBJ , COMPS}

To sum up, the first set of rules, (4.30), introduce a dependency which is to be bound off immediately by the NP being modified by a clause headed by the derived *part* verb itself. The second set, (4.31) doesn't introduce a dependency, but only binds off one inherited from an argument of the *part* verb. And finally, the third set, (4.32), only introduces a dependency that is to be bound off as a long-distance dependency.

Another important feature of the analysis comes up in the implementation of the long-distance relativization pattern (4.11). Note that (4.11b) requires a mechanism which, at the outer clause level, differentiates between the different grammatical roles the gap might have, namely subject/non-subject distinction. Perhaps the most straightforward way of realizing this idea is to assume that subject and non-subject dependencies introduced by relativized constituents are passed up in the structure using different RELATIVIZED features. Hence, I assume that RELATIVIZED takes values of type *relativized*, a new type with two appropriate features, say SUBJ-REL and NON-SUBJ-REL, both of which take values of type *set(local)*. What is essential is then to make sure that whenever a subject dependency is introduced (by a lexical rule in (4.30) or (4.32)), it is stored in the RELATIVIZED|SUBJ-REL value of the output, and a non-subject dependency is always stored in the RELATIVIZED|NON-SUBJ-REL value. That way, it would be possible to choose the right strategy to bind off a long-distance dependency by an output of (4.31), depending on whether the dependency inherited from the gap host is passed on by its RELATIVIZED|SUBJ-REL or RELATIVIZED|NON-SUBJ-REL value. How this could be achieved will become clear in Section 4.4.2, where I discuss the details of the rules in the set (4.31).

Note that the above modification to the RELATIVIZED feature further requires us to revise the (parochial) Finite Sentence Relativized Constraint, (4.29), in the following way:

(4.33) Finite Sentence Relativized Constraint (final version):

$$\begin{bmatrix} \textit{phrase} & \\ \text{SYNSEM} & \text{S}[fin] \end{bmatrix} \Rightarrow \begin{bmatrix} \text{INHER}|\text{RELATIVIZED} & \begin{bmatrix} \text{SUBJ-REL} & \{\} \\ \text{NON-SUBJ-REL} & \{\} \end{bmatrix} \end{bmatrix}$$

Finally, recall from Table 2.1 in Chapter 2 (Section 2.3) the following constraints on the type *hd-adjunct-ph*, which guarantee that in any head-adjunct phrase, the adjunct daughter's MOD specification is structure-shared with the SYNSEM value of the head daughter, and also that the CONTENT value of any such phrase is structure-shared with that of its adjunct daughter.

(4.34)

$$\textit{hd-adjunct-ph} \Rightarrow \begin{bmatrix} \text{CONTENT} & \boxed{1} \\ \text{HD-DTR} & \begin{bmatrix} \textit{phrase} \\ \text{SYNSEM} \ \boxed{2} \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \text{HEAD}|\text{MOD} & \boxed{2} \\ \text{CONTENT} & \boxed{1} \end{bmatrix} \right\rangle \end{bmatrix}$$

In this section, I outlined the main features of the HPSG analysis I propose for relativization in Turkish. It is now time to discuss in detail the lexical rules in each of the three sets schematized in (4.30)-(4.32).

4.4.1 Participle Derivation for Bounded Relativization

There are three rules in this set, with the following functions: i) participle derivation for subject relativization, ii) participle derivation for object relativization for cases where the subject is genitive marked; and iii) participle derivation for object relativization for cases where the subject is unmarked (i.e. nominative). With the latest modification to the RELATIVIZED feature, we need to revise the schematization in (4.30) for these rules, in the following way:

(4.35)

$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{HEAD} & \textit{verb}\begin{bmatrix} \text{VFORM} & \textit{bse} \\ \text{MOD} & \textit{none} \end{bmatrix} \\ \text{F} & \langle \ldots, \text{NP}[\text{LOC}\ \boxed{2}], \ldots \rangle \\ \text{CONTENT} & \boxed{3} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{part-suffix} > \\ \text{HEAD} & \textit{verb}\begin{bmatrix} \text{VFORM } \textit{part} \\ \text{MOD} \quad \text{NP}\left[\text{TO-BIND|RELATIVIZED|Y}\left\{\boxed{2}_{\boxed{4}}\right\}\right] : \begin{bmatrix}\text{INDEX } \boxed{4} \\ \text{RESTR } \boxed{5}\end{bmatrix} \end{bmatrix} \\ \text{F} & \langle \ldots \rangle \\ \text{CONTENT} & \textit{nom-obj}\begin{bmatrix}\text{INDEX } \boxed{4} \\ \text{RESTR } \{\boxed{3}\} \cup \boxed{5}\end{bmatrix} \\ \text{INHER|RELATIVIZED|Y} & \{\boxed{2}\} \end{bmatrix}$$

where $\text{F} \in \{\text{SUBJ}, \text{COMPS}\}$ and
$\text{Y} \in \{\text{SUBJ-REL}, \text{NON-SUBJ-REL}\}$

Below I present the three rules mentioned above in detail, together with example structures which illustrate the use of the outputs of those rules. For expository reasons, I omit the parts that are exactly the same as in (4.35).

Subject Relativization:

The main point to note about (4.36), the subject relativization rule, is that it is the only element in the SUBJ list of the input that gets relativized, i.e. that is removed from that list, and gets its LOCAL value placed in the INHER|RELATIVIZED|SUBJ-REL set of the output, and hence a subject dependency is introduced. Consequently, the PHON value of the output gets the SPc suffix *'-(y)En'*, signalling the choice of the SPc strategy.

(4.36)
$$\begin{bmatrix} \text{PHON } \boxed{1} \\ \text{MOD } \textit{none} \\ \text{SUBJ } \langle \text{NP}[\text{LOC } \boxed{2}] \rangle \end{bmatrix}$$

$$\Downarrow$$

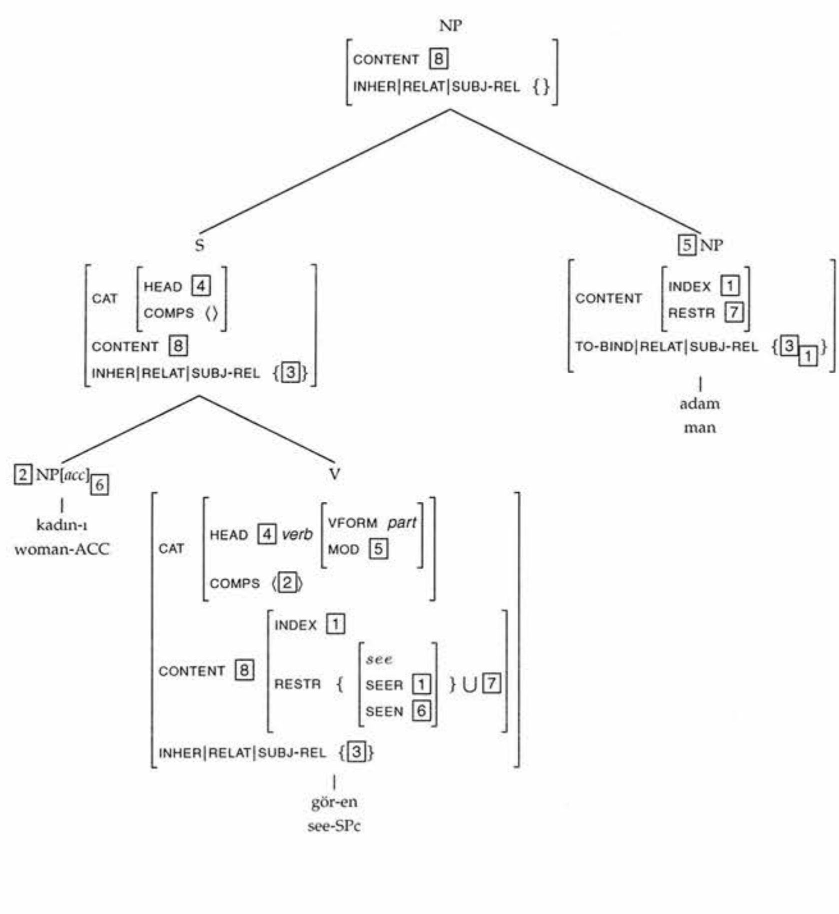
$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{yEn} > \\ \text{MOD} & \text{NP}_{\boxed{3}}\left[\text{TO-BIND|RELATIVIZED|SUBJ-REL}\left\{\boxed{2}_{\boxed{3}}\right\}\right] \\ \text{SUBJ} & \langle\rangle \\ \text{INHER|RELATIVIZED|SUBJ-REL} & \{\boxed{2}\} \end{bmatrix}$$

Let us now consider an example that illustrates the use of an output of (4.36). The structure of (4.1b), repeated below, is given in (4.37). Note that the lexical entry for the *part* verb *'gören'* in this structure is the output of (4.36). The structure-sharing (tag $\boxed{1}$) of the

INDEX values of the modified NP and the element in the TO-BIND|RELATIVIZED|SUBJ-REL set of the NP (hence, the element in the INHER|RELATIVIZED|SUBJ-REL set of the *part* verb), ensures that the SEER role of the *'see'* relation (in the CONTENT|RESTR value of the *part* verb) is filled by this index, since it is the index of the subject NP in the original lexical entry for the *base* verb *'gör'*, which has been relativized by (4.36). The non-empty INHER|RELATIVIZED|SUBJ-REL value introduced by the lexical entry of the participle is passed on to the mother S node, via the Nonlocal Feature Principle (NFP), and then bound off by the TO-BIND|RELATIVIZED|SUBJ-REL value of the modified NP again via the NFP, leaving the mother NP node with an empty INHER|RELATIVIZED|SUBJ-REL value. Note also that the CONTENT value of the *part* verb, which is structure-shared with that of the relative clause, is further structure-shared with the CONTENT value of the mother NP (via the relevant constraints on the types *hd-comp-ph* and *hd-adjunct-ph*, respectively).

(4.1b) [$\_{i}$ kadın-ı gör-en] adam$_i$ 'the man who saw the woman'
woman-ACC see-SPc man

(4.37)



Object Relativization when the Subject is Genitive-marked:

(4.38) relativizes one of the arguments in the COMPS list of the input by removing it from that list, and placing its LOCAL value in the INHER|RELATIVIZED|NON-SUBJ-REL set of the output, thereby introducing a non-subject dependency. The strategy chosen is OPc, hence the OPc suffix *'-dIk'* affixed to the PHON value of the output, together with a possessive suffix that agrees with the subject, which is further restricted to have a CASE value of type *genitive*.

(4.38)

$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{MOD} & \textit{none} \\ \text{SUBJ} & \langle \text{NP} \rangle \\ \text{COMPS} & \langle \ldots, \text{NP}[\text{LOC}\,\boxed{2}], \ldots \rangle \end{bmatrix}$$

$$\Downarrow$$

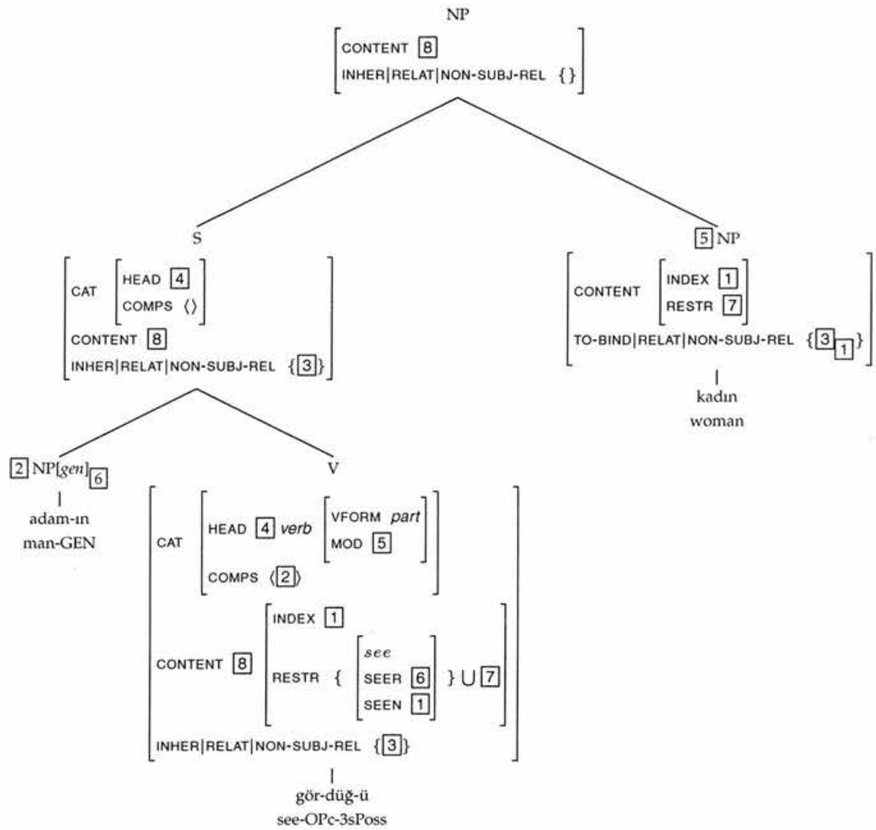
$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{dIk} > \bullet < \textit{possess-suffix}(\boxed{3}) > \\ \text{MOD} & \text{NP}_{\boxed{4}} \left[\text{TO-BIND|RELATIVIZED|NON-SUBJ-REL} \left\{ \boxed{2}_{\boxed{4}} \right\} \right] \\ \text{SUBJ} & \left\langle \text{NP}_{\boxed{3}}[\textit{gen}] \right\rangle \\ \text{COMPS} & \langle \ldots \rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\boxed{2}\} \end{bmatrix}$$

Consider now the structure of (4.1c), given in (4.39), where the lexical entry for the *part* verb *'gördüğü'* is an output of (4.38).[17] Notice that since the index of the modified NP is structure-shared with that of the relativized object of the *base* verb *'gör'*, in this case, it is the SEEN role of the *see* relation (in the CONTENT|RESTR value of the *part* verb) that is assigned to that index.

(4.1c) [adam-ın $\_{i}$ gör-düğ-ü] kadın$_i$
man-GEN see-OPc-3sPoss woman
'the woman that the man saw'

[17] Recall from Chapter 3 (Section 3.4.1) that I assume a flat sentence structure for Turkish, where the verbal head selects its subject via the COMPS feature, rather than SUBJ; cf. the lexical rule in (3.19) in that section. Hence, *'gördüğü'* in (4.39) selects its subject NP *'adamın'*, via its COMPS feature (tag $\boxed{2}$).

(4.39)



Object Relativization when the Subject is Nominative:

(4.40) is similar to (4.38), in that it also introduces a non-subject dependency, by relativizing an argument in the COMPS list of its input entry. In this case, however, the subject of the output is constrained as *nominative* and *nonspecific* (cf. Chapter 3, Section 3.2, and also fn. 3 in this chapter). Accordingly, the relativization strategy chosen is SPc, hence the SPc suffix '*-(y)En*' affixed to the PHON value of the output.

(4.40)

PHON	[1]
MOD	*none*
SUBJ	⟨NP⟩
COMPS	⟨…, NP[LOC [2]], …⟩

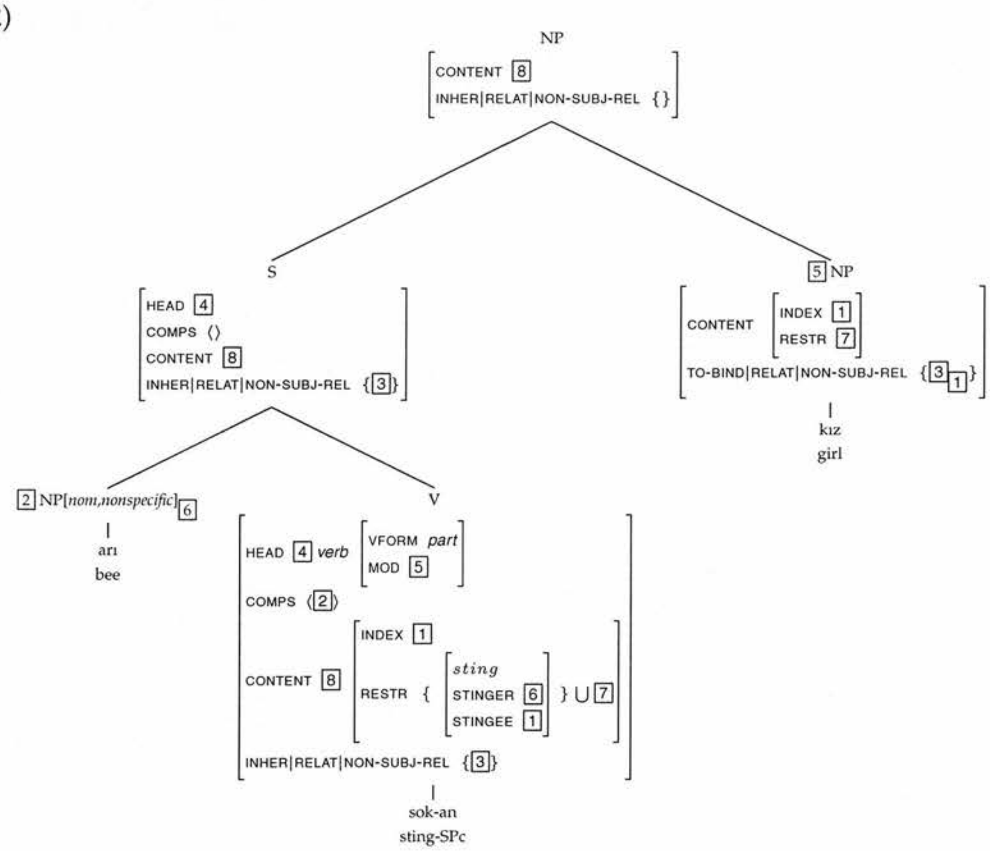
⇓

PHON	[1] • < *yEn* >
MOD	$NP_{[3]}$ [TO-BIND|RELATIVIZED|NON-SUBJ-REL {[2][3]}]
SUBJ	⟨NP[*nom, nonspecific*]⟩
COMPS	⟨…⟩
INHER|RELATIVIZED|NON-SUBJ-REL	{[2]}

The structure of (4.41), an example of object relativization where the subject is nominative, is given in (4.42). The lexical entry for the *part* verb *'sokan'* in this structure is an output of (4.40), hence the subject is *nominative* and *nonspecific*.

(4.41) [$\_{i}$ arı sok-an] kız$_i$ 'the girl whom some bee/bees stung'
bee sting-SPc girl

(4.42)



4.4.2 Participle Derivation for Long-distance Relativization

The lexical rules in this section derive *part* verbs (from *base* verbs) which are to function as heads of relative clauses where a long-distance dependency is bound off. There are four rules in this set, each of which corresponds to one of the cases in the long-distance relativization pattern (4.11), repeated below:

(4.11) **Long-distance relativization pattern in Turkish:**

(a) if the gap host is a *non-subject* constituent then

  (i) if there is a *genitive* subject in the clause then

   the OPc strategy is used

  (ii) else

   the SPc strategy is used

(b) else if the gap host is the *subject* then

  (i) if the grammatical role of the gap is *subject* then

   the SPc strategy is used

  (ii) else

   the OPc strategy is used

The schematization in (4.31), given for these four rules, should be revised as in (4.43) to adapt to the modification in the RELATIVIZED feature.

(4.43)

$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{HEAD} & verb\begin{bmatrix} \text{VFORM} & bse \\ \text{MOD} & none \end{bmatrix} \\ \text{F} & \langle \ldots, \text{XP}, \ldots \rangle \\ \text{CONTENT} & \boxed{2} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{part-suffix} > \\ \text{HEAD} & verb\begin{bmatrix} \text{VFORM} & part \\ \text{MOD} & \text{NP}\left[\text{TO-BIND}|\text{RELATIVIZED}|\text{Y}\left\{\boxed{5}_{\boxed{3}}\right\}\right] : \begin{bmatrix} \text{INDEX} & \boxed{3} \\ \text{RESTR} & \boxed{4} \end{bmatrix} \end{bmatrix} \\ \text{F} & \langle \ldots, \text{XP}\left[\text{INHER}|\text{RELATIVIZED}|\text{Y}\ \{\ldots, \boxed{5}, \ldots\}\right], \ldots \rangle \\ \text{CONTENT} & \textit{nom-obj}\begin{bmatrix} \text{INDEX} & \boxed{3} \\ \text{RESTR} & \{\boxed{2}\} \cup \boxed{4} \end{bmatrix} \end{bmatrix}$$

where F $\in$ {SUBJ , COMPS} ,

Y $\in$ {SUBJ-REL , NON-SUBJ-REL} , and

XP $\in$ { NP , PP , S[*nomin*] , VP[*inf*] }

Let us now discuss the details of these four rules, together with examples, again omitting the parts that are exactly the same as in (4.43).

Non-subject Gap Host – Genitive Subject (4.11ai)

The lexical rule in (4.44) deals with the case in (4.11ai), where the gap host is a non-subject constituent, and there is a *genitive* subject in the clause. Accordingly, one of the arguments in the COMPS list is constrained to have a non-empty value for either of the INHER|RELATIVIZED features, and the subject is constrained as *genitive*. Notice the OPc suffix affixed to the PHON value of the output, together with a possessive suffix that agrees with the subject, signalling the choice of the OPc strategy.

(4.44)

$$\left[\begin{array}{ll} \text{PHON} & \boxed{1} \\ \text{MOD} & \textit{none} \\ \text{SUBJ} & \langle \text{NP} \rangle \\ \text{COMPS} & \langle \ldots, \text{XP}, \ldots \rangle \end{array}\right]$$

$$\Downarrow$$

$$\left[\begin{array}{ll} \text{PHON} & \boxed{1} \bullet < \textit{dlk} > \bullet < \textit{possess-suffix}(\boxed{2}) > \\ \text{MOD} & \text{NP}_{\boxed{3}} \left[\text{TO-BIND|RELATIVIZED|Y} \left\{ \boxed{4}_{\boxed{3}} \right\}\right] \\ \text{SUBJ} & \left\langle \text{NP}_{\boxed{2}}[\textit{gen}] \right\rangle \\ \text{COMPS} & \langle \ldots, \text{XP}\,[\text{INHER|RELATIVIZED|Y}\ \{\ldots, \boxed{4}, \ldots\}], \ldots \rangle \end{array}\right]$$

where $\text{Y} \in \{\text{SUBJ-REL}, \text{NON-SUBJ-REL}\}$ and
$\text{XP} \in \{\text{NP}, \text{PP}, \text{S}[\textit{nomin}], \text{VP}[\textit{inf}]\}$

Consider now the example of long-distance relativization given in (4.45), where the possessor of the embedded NP has been relativized, and its structure in (4.46). The lexical entry for the *part* verb *'gördüğün'* in this structure is an output of (4.44), where the gap host (XP) is an NP (a possessive phrase) and the grammatical role of the gap is subject (hence, Y is SUBJ-REL). The lexical entry for the noun *'kitabını'* is the output of the rule in (4.65) in Section 4.4.3,[18] which deals with relativization of possessors out of possessive phrases.[19] The non-empty INHER|RELATIVIZED|SUBJ-REL value

[18] The examples in this section refer to certain lexical rules in Section 4.4.3, which deal with relativization out of embedded phrases. I opt for presenting these two sections in this order, since the following section also has references to this section, and since it would be much more difficult to follow the discussion in the other order.

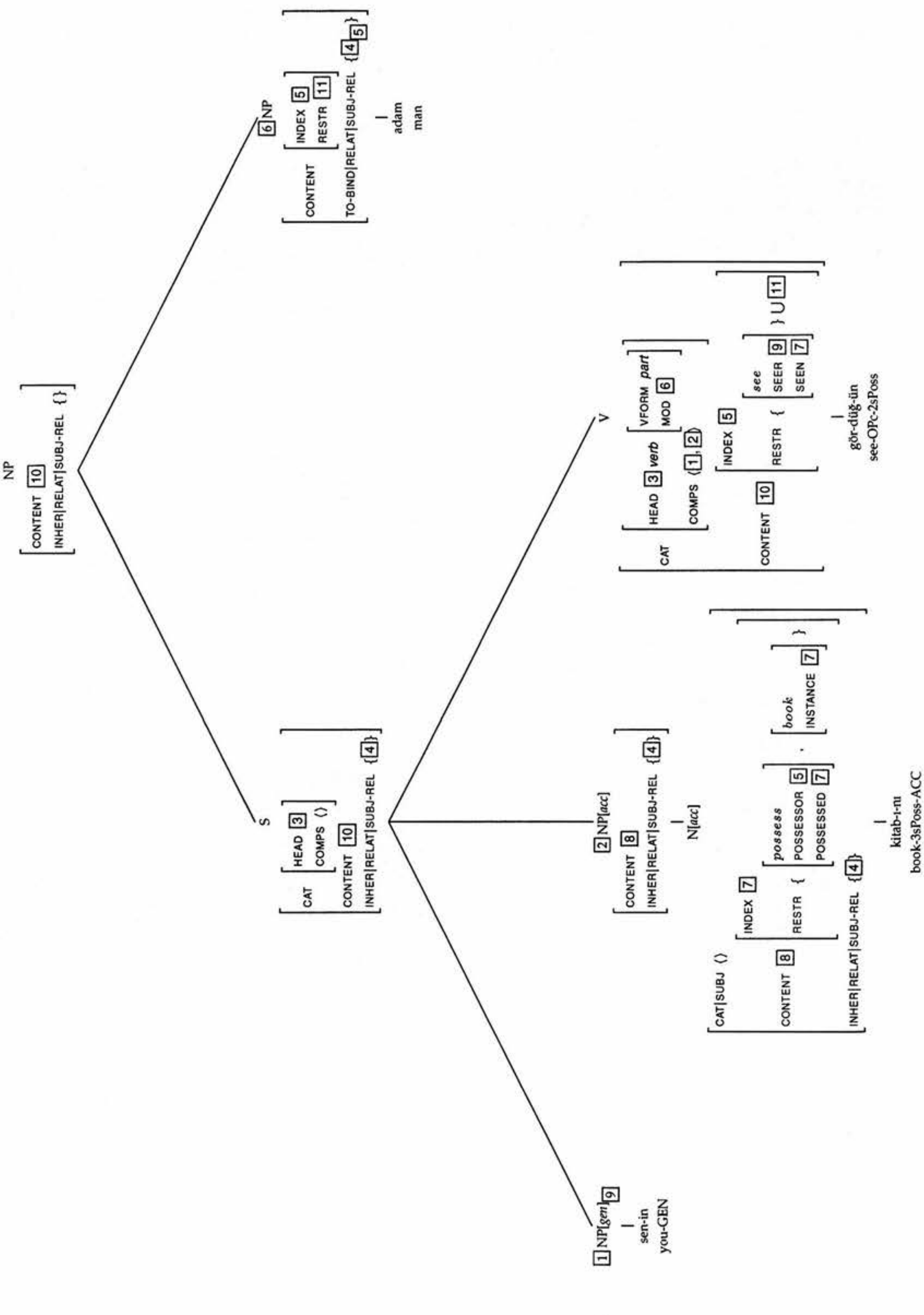
[19] In fact, this entry has also gone through an additional lexical rule, which has affixed the accusative case suffix to its PHON value and changed its CASE value from *nominative* to *accusative*. I prefer to omit such inessential details in the discussion here.

introduced by this lexical entry is passed on to its mother NP, via the NFP, rendering the (non-subject) complement of the *part* verb *'gördüğün'* to have a non-empty value for one of the INHER|RELATIVIZED features (SUBJ-REL in this case), in accordance with the relevant constraint imposed by the rule in (4.44). The CONTENT|RESTR value of the lexical entry for *'kitabını'* contains two psoas, one of them being a *possess* relation.[20] The structure-sharing (tag 5) of the INDEX values of the modified NP and the element its TO-BIND|RELATIVIZED|SUBJ-REL value (hence, the element in the INHER|RELATIVIZED|SUBJ-REL value of the lexical entry for *'kitabını'*) makes sure that the POSSESSOR role of the *possess* relation is filled by this index, since it is the index of the subject (possessor) NP in the original lexical entry for *'kitabını'*, which has been relativized by the rule in (4.65).

(4.45) [sen-in [$\_{i}$ kitab-ı-nı] gör-düğ-ün] adam$_i$
you-GEN book-3sPoss-ACC see-OPc-2sPoss man
'the man whose book you saw'

XP = NP (possessive phrase)
Y = SUBJ-REL

[20]Recall from Chapter 3 (Section 3.3) that this relation is introduced by a lexical rule that deals with possessive suffix affixation to nouns.


NP
S
6 NP
adam
man
1 NP[gen] 9
sen-in
you-GEN
2 NP[acc]
N[acc]
kitab-ı-nı
book-3sPoss-ACC
V
gör-düğ-ün
see-OPc-2sPoss

(4.46)

Let us now consider another example, (4.47), where the gap host (XP) is an NP modified by a relative clause. (4.48) shows the structure of this clause. The lexical entry of the *part* verb *'okuduğu'* in this structure is the output of the consecutive applications of the following lexical rules of relativization:

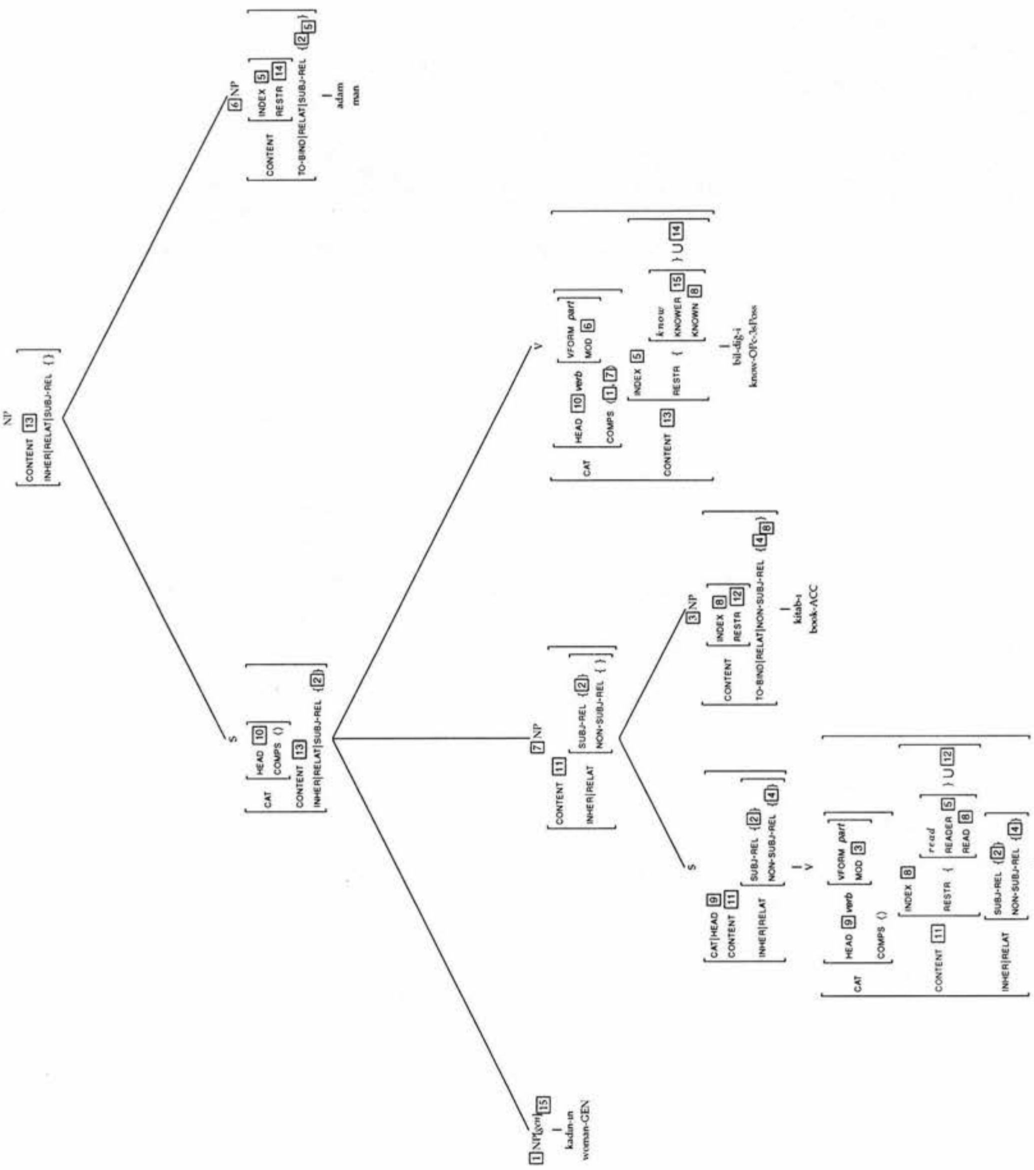
(i) (4.38) has applied to the lexical entry of the *base* verb *'oku'*, deriving a *part* verb with its direct object relativized. Notice the constraints this lexical rule imposes on the VFORM, MOD, CONTENT, COMPS and INHER|RELATIVIZED|NON-SUBJ-REL values of *'okuduğu'*, illustrated in the structure of the embedded relative clause (which is in fact very similar to the structure of (4.1c) given in (4.39) as far as these features are concerned).

(ii) The output *part* verb of the above application has then undergone the rule in (4.67) in Section 4.4.3, which has relativized its subject (placing its LOCAL value in the INHER|RELATIVIZED|SUBJ-REL value of the final output, that is, the one we see as the lexical entry of *'okuduğu'* in (4.48)).

On the mother NP node of the embedded relative clause, the non-empty INHER|RELATIVIZED|NON-SUBJ-REL value introduced by the lexical entry of the *part* verb *'okuduğu'*, no longer shows up, since it has been bound off by the TO-BIND|RELATIVIZED|NON-SUBJ-REL value of the head noun of this relative clause. However, the non-empty INHER|RELATIVIZED|SUBJ-REL value, which has been introduced by the same lexical entry, is still there. Hence, the (non-subject) complement of the *part* verb *'bildiği'* (which is the output of the lexical rule in (4.44)) has a non-empty value for one of the INHER|RELATIVIZED features (SUBJ-REL in this case) satisfying the relevant constraint imposed by (4.44). Note that the READER role of the *read* relation in the CONTENT|RESTR value of the embedded participle *'okuduğu'* is filled by the INDEX of the head noun of the outer relative clause (tag 5), since this is the index of the subject NP of the output *'okuduğu'* of (4.38) (cf. (i) above), which has then been relativized by (4.67) (cf. (ii) above).

(4.47) [kadın-ın [$\_{j}$ $\_{i}$ oku-duğ-u] kitab-ı$_i$ bil-diğ-i] adam$_j$
woman-GEN read-OPc-3sPoss book-ACC know-OPc-3sPoss man
'the man that the woman knows the book he reads'

XP = NP (modified by a relative clause) , Y = SUBJ-REL



(4.48)

Non-subject Gap Host – Nominative Subject (4.11aii)

The lexical rule in (4.49) deals with the case in (4.11aii), where the gap host is a non-subject constituent, and there is no *genitive* subject in the clause. Notice the SPc suffix *'-(y)En'* affixed to the PHON value of the output, since in this case the SPc is to be used. Note that I do not provide an account for relativization out of *impersonal passives* here, hence the SUBJ value of the output (which in the case of *impersonal passives* would be empty) is constrained to contain an NP which is *nominative* and *nonspecific*.

(4.49)
$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{MOD} & none \\ \text{SUBJ} & \langle \text{NP} \rangle \\ \text{COMPS} & \langle \ldots, \text{XP}, \ldots \rangle \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < yEn > \\ \text{MOD} & \text{NP}_{\boxed{2}} \left[\text{TO-BIND}|\text{RELATIVIZED}|\text{Y} \left\{ \boxed{3}_{\boxed{2}} \right\} \right] \\ \text{SUBJ} & \langle \text{NP}[nom, nonspecific] \rangle \\ \text{COMPS} & \langle \ldots, \text{XP}\,[\text{INHER}|\text{RELATIVIZED}|\text{Y}\ \{\ldots, \boxed{3}, \ldots\}], \ldots \rangle \end{bmatrix}$$

where $\text{Y} \in \{\text{SUBJ-REL}, \text{NON-SUBJ-REL}\}$ and
$\text{XP} \in \{\text{NP}, \text{PP}, \text{S}[nomin], \text{VP}[inf]\}$

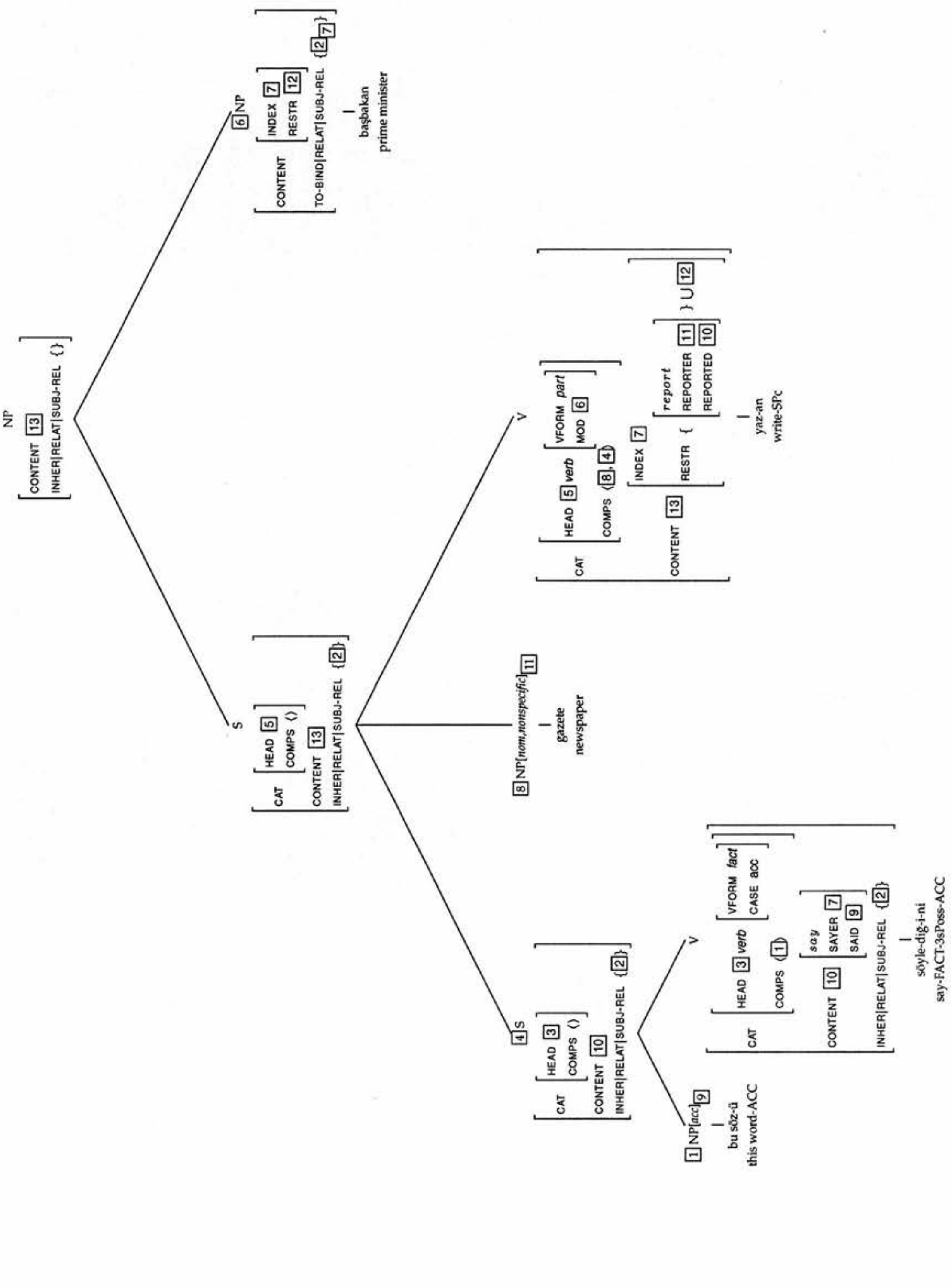
Let us now consider a couple of examples to illustrate the use of (4.49). The first example, given in (4.50), is one where the gap host is a *fact* type nominalization phrase that functions as the accusative object of the relative clause (XP is S[*fact*]), and the grammatical role of the gap is subject (Y is SUBJ-REL). (4.51) shows the structure for this example. The lexical entry for the *fact* verb *'söylediğini'* is the output of the rule in (4.69) in Section 4.4.3, which has relativized its subject. The non-empty INHER|RELTIVIZED|SUBJ-REL value introduced by this entry is passed on to the mother S node. Hence, the (non-subject) complement of the *part* verb *'yazan'* has a non-empty value for one of the INHER|RELTIVIZED features (SUBJ-REL in this case), satisfying the relevant constraint imposed by (4.49). The rest of the structure is very similar to the ones in the previous case, except that the subject of the relative clause is *nominative* and *nonspecific* in this case.

(4.50) [[$\_\_i$ bu söz-ü söyle-diğ-i-ni] gazete yaz-an] başbakan$_i$
this word-ACC say-FACT-3sPoss-ACC newspaper write-SPc prime minister
'the prime minister who some newspaper/newspapers reported to have said these words'

XP = S

Y = SUBJ-REL

(4.51)


NP
CONTENT 13
INHER|RELAT|SUBJ-REL { }
S
6 NP
başbakan
prime minister
8 NP[nom,nonspecific] 11
gazete
newspaper
V
yaz-an
write-SPc
4 S
1 NP[acc] 9
bu söz-ü
this word-ACC
söyle-diğ-i-ni
say-FACT-3sPoss-ACC

In the next example, (4.52), the gap host is the accusative object of the outer relative clause, modified by an embedded relative clause (XP is NP), and the grammatical role of the gap is object (Y is NON-SUBJ-REL). In the structure for this example, given in (4.53), the lexical entry for the *part* verb *'yiyen'* is the output of the following consecutive applications of two of the lexical rules of relativization:

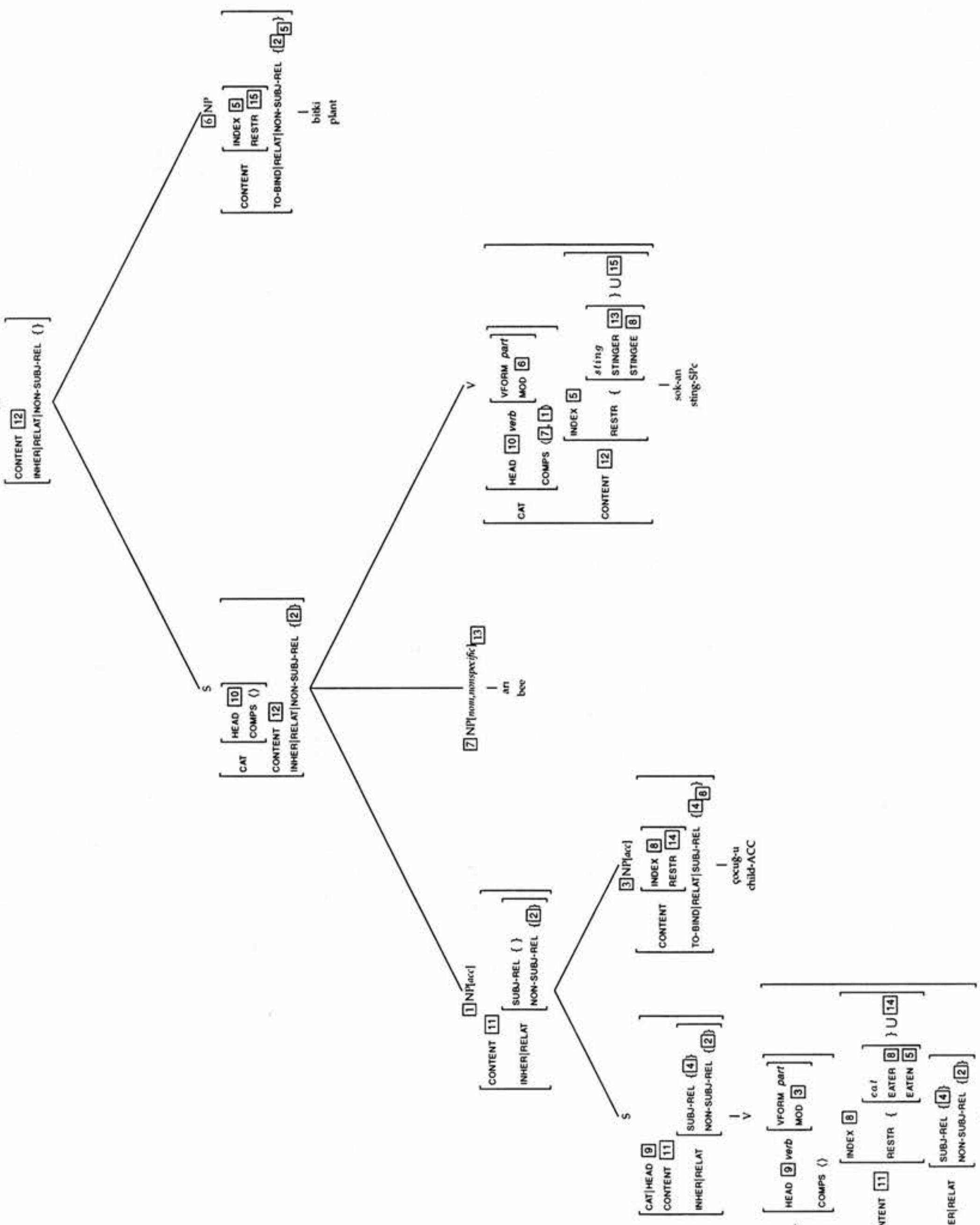
(i) (4.36) has derived the *part* verb *'yiyen'* from the lexical entry of the *base* verb *'ye'*, relativizing its subject. Note how the constraints imposed by this rule on the VFORM, MOD, CONTENT, SUBJ and INHER|RELATIVIZED|SUBJ-REL values of the output have been illustrated in the structure of the embedded relative clause (which is in fact very similar to the structure of (4.1b) given in (4.37) as far as these features are concerned).

(i) This *part* verb has then undergone the lexical rule in (4.68), which has relativized the element in its COMPS list, placing it in the INHER|RELATIVIZED|NON-SUBJ-REL value of the output.

On the mother NP node of the embedded relative clause, the INHER|RELATIVIZED|SUBJ-REL value is empty, since the non-empty INHER|RELATIVIZED|SUBJ-REL value introduced by the lexical entry of *'yiyen'* has been bound off by the TO-BIND|RELATIVIZED|SUBJ-REL value of the head noun of this relative clause. The INHER|RELATIVIZED|NON-SUBJ-REL value, however, is still non-empty. The (non-subject) complement of the *part* verb *'sokan'* (which is an output of (4.49) therefore has a non-empty value for one of its INHER|RELATIVIZED features (NON-SUBJ-REL in this case).

(4.52) [[$\_{i}$ $\_{j}$ yi-yen] çocuğ-u$_i$ arı sok-an] bitki $_j$
eat-SPc child-ACC bee sting-SPc plant
'the plant such that the child who ate it was stung by a bee'

XP = NP (modified by a relative clause)
Y = NON-SUBJ-REL


bitki
plant
arı
bee
sok-an
sting-SPc
çocuğ-u
child-ACC
yi-yen
eat-SPc

(4.53)

Subject Gap Host – Subject Gap (4.11bi)

Let us now discuss the formulation of the third case in (4.11), i.e. the case where the gap host is the subject of the relative clause in question, and the grammatical role of the gap is also subject. The lexical rule in (4.54), which derives *part* verbs to function as heads of clauses that fit in this case, therefore constrains its output to have a subject with a non-empty INHER|RELATIVIZED|SUBJ-REL value. The fact that the SPc is to be used in this case determines the SPc suffix *'-(y)En'* affixed to the PHON value of the output. Notice that VP[*inf*], which appears in the list of possible syntactic categories for the gap host in the lexical rules (4.44) and (4.49), is left out in (4.54), since relativization is not possible out of subject infinitive phrases (cf. Section 4.2.5).

(4.54)
$$\begin{bmatrix} \text{PHON} & \boxed{1} \\ \text{MOD} & \textit{none} \\ \text{SUBJ} & \langle \text{XP} \rangle \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON} & \boxed{1} \bullet < \textit{yEn} > \\ \text{MOD} & \text{NP}_{\boxed{2}} \left[\text{TO-BIND|RELATIVIZED|SUBJ-REL} \left\{ \boxed{3}_{\boxed{2}} \right\} \right] \\ \text{SUBJ} & \langle \text{XP} [\textit{nom}, \text{INHER|RELATIVIZED|SUBJ-REL} \{ \ldots, \boxed{3}, \ldots \}] \rangle \end{bmatrix}$$

where XP $\in$ { NP , S[*nomin*] }

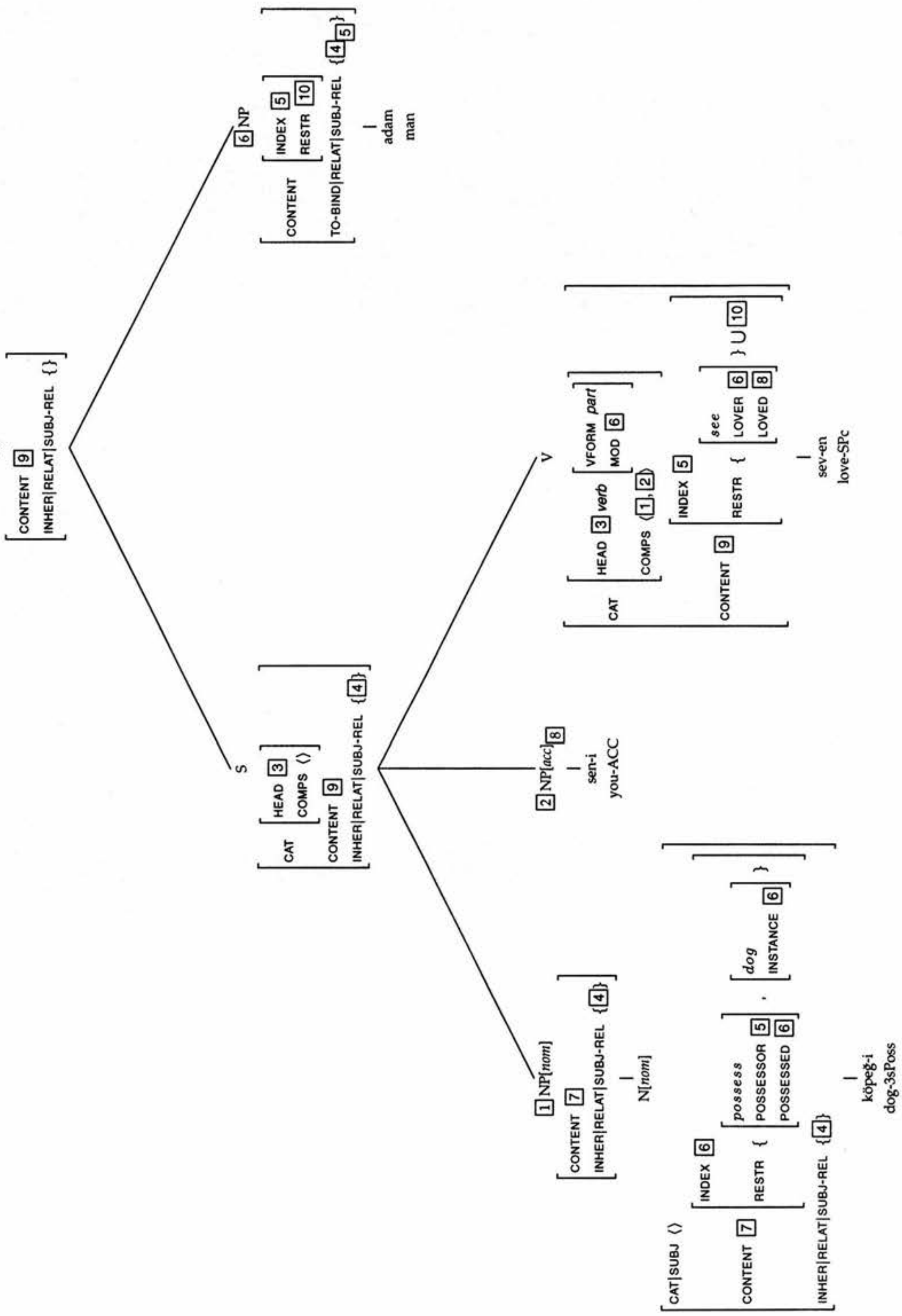
Consider now the example in (4.55), where the possessor of the embedded subject NP has been relativized. In the structure for this example, given in (4.56), the lexical entry of the *part* verb *'seven'* is the output of the rule in (4.54), where the gap host is an NP. And the lexical entry of the noun *'köpeği'* is the output of the rule in (4.65), which has relativized the subject (possessor) of the input, placing it in the INHER|RELATIVIZED|SUBJ-REL value of this output entry. This non-empty INHER|RELATIVIZED|SUBJ-REL value is passed on to the mother NP node. The subject of the *part* verb *'seven'* (which it selects via its COMPS feature)[21] therefore has a non-empty INHER|RELATIVIZED|SUBJ-REL value, in line with the relevant constraint imposed by the rule (4.54).

(4.55) [[$\_{i}$ köpeğ-i] sen-i sev-en] adam$_i$
dog-3sPoss you-ACC love-SPc man
'the man whose dog loves you'

XP = NP (possessive phrase)

[21] Note, once again, that I assume a flat sentence structure for Turkish, where the verbal head selects its subject via the COMPS feature, rather than SUBJ; see Chapter 3 (Section 3.4.1), and in particular the lexical rule in (3.19).

(4.56)

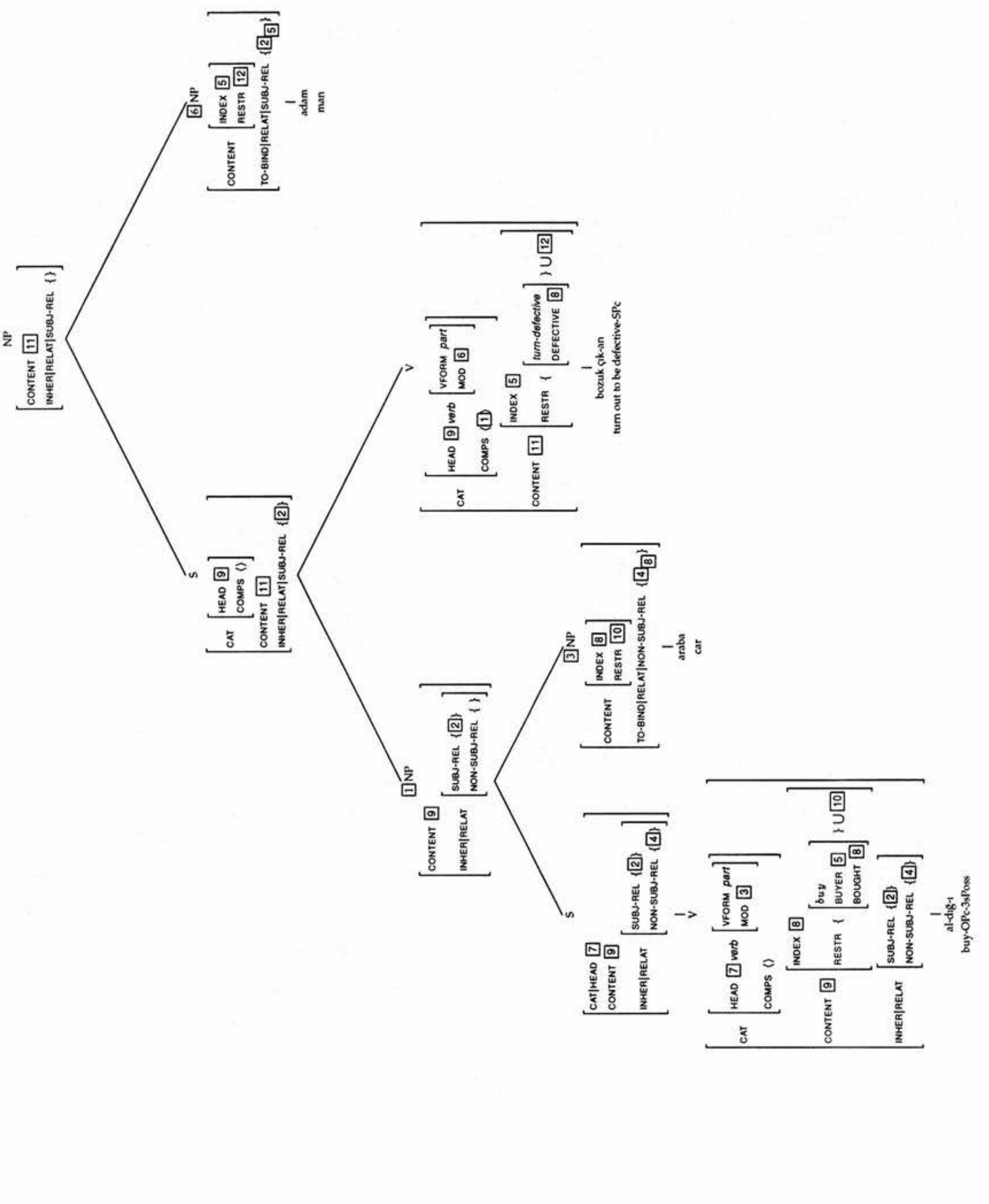

NP
CONTENT 9
INHER|RELAT|SUBJ-REL { }
S
CAT HEAD 3 COMPS ⟨⟩
CONTENT 9
INHER|RELAT|SUBJ-REL {4}
6 NP
CONTENT INDEX 5 RESTR 10
TO-BIND|RELAT|SUBJ-REL {4 5}
adam
man
1 NP[nom]
CONTENT 7
INHER|RELAT|SUBJ-REL {4}
N[nom]
CAT|SUBJ ⟨⟩
CONTENT 7 INDEX 6 RESTR { possess POSSESSOR 5 POSSESSED 6 , dog INSTANCE 6 }
INHER|RELAT|SUBJ-REL {4}
köpeğ-i
dog-3sPoss
2 NP[acc] 8
sen-i
you-ACC
V
CAT HEAD 3 verb VFORM part MOD 6
COMPS ⟨1, 2⟩
CONTENT 9 INDEX 5 RESTR { see LOVER 6 LOVED 8 } ∪ 10
sev-en
love-SPc

Another example in relation to the rule (4.54) is given in (4.57), and its structure in (4.58). The gap host in this case is the subject NP, modified by a relative clause. The *part* verb *'aldığı'*, the head of that embedded relative clause, is the output of the consecutive applications of the rules (4.38) and (4.67), exactly like the *part* verbal head of the embedded relative clause in (4.48) (page 113). And the *part* verb *'bozuk çıkan'* in the outer relative clause is the output of the rule in (4.54).

(4.57) [[$\_{j}$ $\_{i}$ al-dığ-ı] araba$_i$ bozuk çık-an] adam$_j$
buy-OPc-3sPoss car defective turn out-SPc man
'the man who the car that he bought turned out to be defective'

XP = NP (modified by a relative clause)

(4.58)


adam
man
araba
car
bozuk çık-an
turn out to be defective-SPc
al-dığ-ı
buy-OPc-3sPoss

Subject Gap Host – Non-subject Gap (4.11bii)

We now come to the last case in (4.11), i.e. the one with a subject gap host and a non-subject gap. Recall from Section 4.2 that the subject of the relative clause (the gap host) in this case is marked genitive, and the OPc is used. Accordingly, (4.59), which handles this case, affixes the OPc suffix *'-dIk'* to the PHON value of the output, together with a possessive suffix which agrees with the *genitive* subject, and constrains the INHER|RELATIVIZED|NON-SUBJ-REL value of the element in the SUBJ list of the output to be non-empty (and consequently the TO-BIND|RELATIVIZED|NON-SUBJ-REL value of the head noun). VP[*inf*] is again excluded from the list of possible syntactic categories for the gap host (XP) as in the case of the lexical rule in (4.54).

(4.59)

$$\begin{bmatrix} \text{PHON } \boxed{1} \\ \text{MOD } \textit{none} \\ \text{SUBJ } \langle \text{XP} \rangle \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{PHON } \boxed{1} \bullet < \textit{dlk} > \bullet < \textit{possess-suffix}(\boxed{2}) > \\ \text{MOD } \text{NP}_{\boxed{3}} \left[\text{TO-BIND|RELATIVIZED|NON-SUBJ-REL} \left\{ \boxed{4}_{\boxed{3}} \right\} \right] \\ \text{SUBJ } \left\langle \text{XP}_{\boxed{2}} [\textit{gen}, \text{INHER|RELATIVIZED|NON-SUBJ-REL } \{\ldots, \boxed{4}, \ldots\}] \right\rangle \end{bmatrix}$$

where XP ∈ { NP , S[*nomin*] }

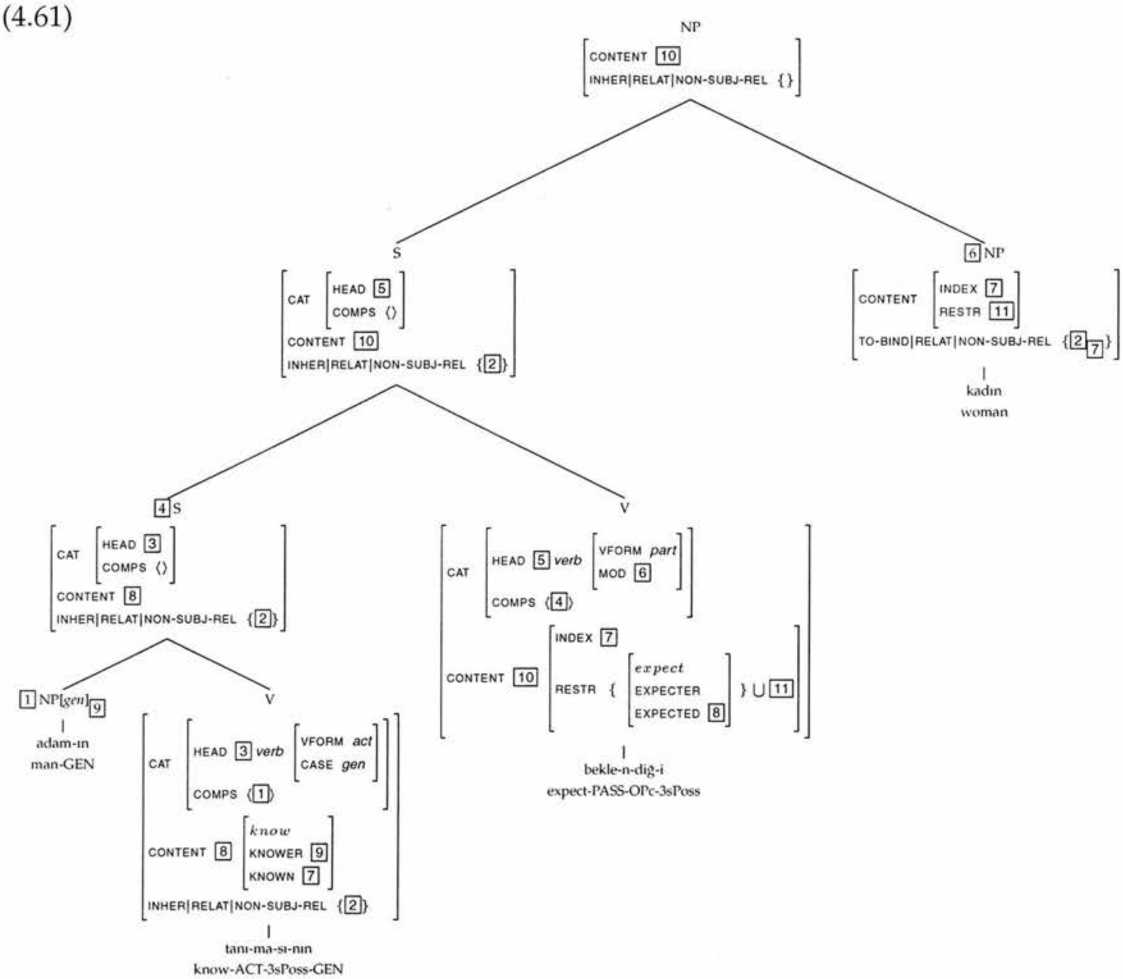
Consider now the example in (4.60), and its structure in (4.61). Note that the gap host in this case is an *act* type nominalization phrase, which acts as the *genitive* subject of the relative clause. The verbal head *'tanımasının'* of this subject is an output of the rule in (4.70) in Section 4.4.3, which has relativized its object. And the *part* verb *'beklendiği'* in the relative clause is an output of (4.59).

(4.60) [[adam-ın $\underline{\ \ }_i$ tanı-ma-sı-nın] bekle-n-diğ-i] kadın$_i$
man-GEN know-ACT-3sPoss-GEN expect-PASS-OPc-3sPoss woman
'the woman such that it is expected that the man knows her'[22]

XP = S

[22]Recall from Section 4.2.3 that in the case of relativization out of subject nominalization phrases, the SPc may be used even when the grammatical role of the gap is non-subject (cf. (4.18)). Recall also that this is a problem with the relativization pattern in (4.11). In order to get around this problem within the HPSG analysis, I propose an additional lexical rule that derives a *part* verb from a *base* verb, affixing the SPc suffix to its PHON value, as well as constraining the subject to be a *nominative* nominalization phrase with a non-empty INHER|RELATIVIZED|NON-SUBJ-REL value. Such a rule then derives *part* verbs like *'beklenen'*, which, for instance, licences a structure for (i) that is exactly the same as (4.61), except that the S[*act*] gap

(4.61)



A second example with regard to the rule in (4.59) is given in (4.62), and its structure in (4.63). The *part* verb *'kullanan'* in this structure (the head of the embedded relative clause) is the output of the consecutive applications of the two rules in (4.36) and (4.68), just like the *part* verb *'yiyen'* in (4.53) (page 117). And the *part* verb *'çıldırdığı'* in the outer clause is the output of the rule in (4.59).

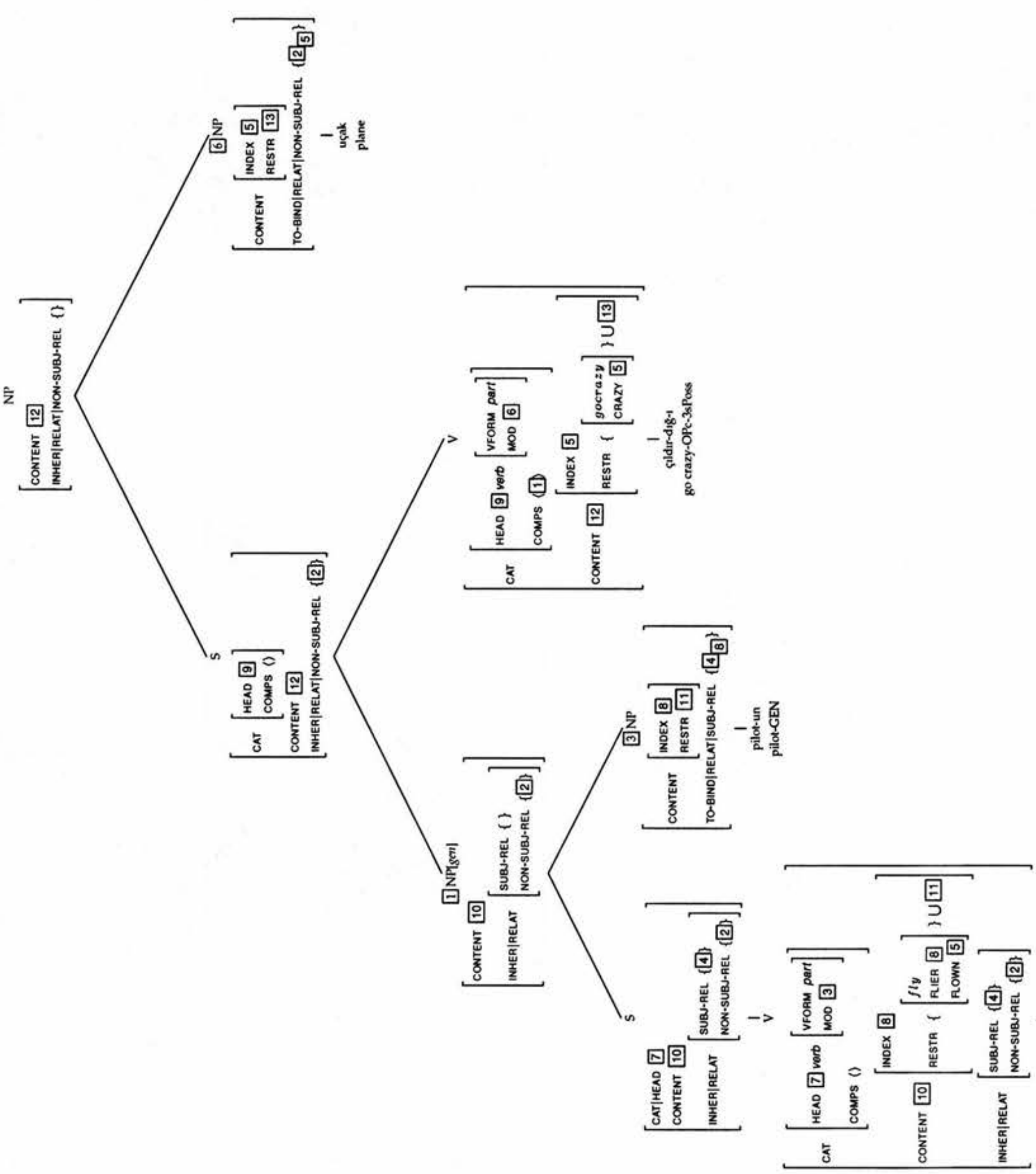
(4.62) [[$\_i$ $\_j$ kullan-an] pilot-un$_i$ çıldır-dığ-ı] uçak$_j$
fly-SPc pilot-GEN go crazy-OPc-3sPoss plane
'the plane which the pilot who was flying it went crazy'

---

host (headed by the *act* verb *'tanıması'*) is *nominative* and the head of the relative clause is the *part* verb *'beklenen'* (derived by the above-mentioned lexical rule).

(i) [[adam-ın $\_i$ tanı-ma-sı] bekle-n-en] kadın$_i$
man-GEN know-ACT-3sPoss expect-PASS-SPc woman
'the woman such that it is expected that the man knows her'

XP = NP (modified by a relative clause)



(4.63)

4.4.3 Relativization out of Embedded Phrases

Let us now discuss the final set of relativization lexical rules, that is, the rules that deal with argument relativization out of embedded phrases of the kinds discussed in Section 4.2, by simply relativizing one of the arguments of the input entry, which is to function as the head of such an embedded phrase. These rules, which differ from each other only with minimal variations, can be schematized as in (4.64).

$$(4.64)\quad \begin{bmatrix} \text{F} & \langle \ldots, \text{NP}[\text{LOC}\,\boxed{1}], \ldots \rangle \\ \text{INHER}|\text{RELATIVIZED}|\text{Y} & \boxed{2} \end{bmatrix}$$
$$\Downarrow$$
$$\begin{bmatrix} \text{F} & \langle \ldots \rangle \\ \text{INHER}|\text{RELATIVIZED}|\text{Y} & \{\boxed{1}\} \cup \boxed{2} \end{bmatrix}$$

where $\text{F} \in \{\text{SUBJ}, \text{COMPS}\}$ and
$\text{Y} \in \{\text{SUBJ-REL}, \text{NON-SUBJ-REL}\}$

Below I discuss the details of each of these rules.

Possessor (Subject) Relativization out of Noun Phrases

The lexical rule in (4.65) deals with relativization of possessors (subjects) out of noun phrases, by simply relativizing the possessor of an input noun, i.e. removing the only element in the SUBJ list of the noun from that list, and placing its LOCAL value within the INHER|RELATIVIZED|SUBJ-REL value of the output. The input noun is constrained to have *nominative* CASE to eliminate redundant lexical ambiguity (since there are other lexical rules that operate on the outputs of this rule to affix different case suffixes, at the same time changing the CASE value of the input noun). For an example of the rule of an output of this rule, see the relative clause in (4.45) and its structure in (4.46), and the one in (4.55) and its structure in (4.56).

$$(4.65)\quad \begin{bmatrix} \text{HEAD} & noun\,[\text{CASE}\ nom] \\ \text{SUBJ} & \langle \text{NP}[\text{LOC}\,\boxed{1}] \rangle \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{SUBJ} & \langle\rangle \\ \text{INHER}|\text{RELATIVIZED}|\text{SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

Object Relativization out of Postpositional Phrases

In Section 4.2.4, we saw that postpositions that bear agreement morphology (i.e. those that have a PFORM value of type *obj-agr*; cf. Chapter 3, Section 3.1) let their objects be relativized. The lexical rule in (4.66) deals with this case.

$$(4.66)\quad \begin{bmatrix} \text{HEAD} & \textit{postp}\ [\text{PFORM}\ \textit{obj-agr}] \\ \text{COMPS} & \langle \text{NP}[\text{LOC}\ \boxed{1}] \rangle \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{COMPS} & \langle\rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

Relativization out of Relative Clauses

The lexical rule in (4.67) deals with subject relativization out of relative clauses. The constraint that the subject of the input must have *genitive* CASE value prevents (4.67) from applying to an output of any of the lexical rules (4.40), (4.49) or (4.54). In the first two cases, the subject cannot be relativized because of the constraint discussed in Section 4.3.2, and in the third case, since it is a gap host. Furthermore, the constraint that the subject must have an empty INHER|RELATIVIZED|NON-SUBJ-REL value prevents it from applying to an output of (4.59), since the subject in this case is again a gap host, hence cannot be relativized. See, for instance, the two relative clauses in (4.47) and (4.57), and their structures in (4.48) and (4.58), respectively, which illustrate the use of the outputs of (4.67).

$$(4.67)\quad \begin{bmatrix} \text{HEAD}\ \textit{verb}\ [\text{VFORM}\ \textit{part}] \\ \text{SUBJ}\quad \langle \text{NP}\ [\text{LOC}\ \boxed{1}\ [\textit{gen}], \text{INHER|RELATIVIZED|NON-SUBJ-REL}\ \{\}] \rangle \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{SUBJ} & \langle\rangle \\ \text{INHER|RELATIVIZED|SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

Object relativization out of relative clauses is handled by (4.68), which constrains the relativized object to have empty values for both of the INHER|RELATIVIZED features, to ensure that it is not a gap host. It furthermore constrains the INHER|RELATIVIZED|NON-SUBJ-REL value of the input to be empty, to rule out relativization of more than one non-subject constituent out of the same relative clause (cf. the constraint discussed in Section 4.3.1). See the relative clauses in (4.52) and (4.62), and their structures in (4.53) and (4.63), respectively, for an example of the use of (4.68).

$$(4.68)\quad \begin{bmatrix} \text{HEAD} & \textit{verb}\ [\text{VFORM}\ \textit{part}] \\ \text{COMPS} & \langle \ldots, \text{NP}\ [\text{LOC}\ \boxed{1}, \text{INHER|RELATIVIZED|Y}\ \{\}], \ldots \rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{COMPS} & \langle \ldots \rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

where Y $\in$ {SUBJ-REL , NON-SUBJ-REL}

Relativization out of Nominalization Phrases

The lexical rule in (4.69) relativizes *genitive* subjects of *nominalization* verbs (since *nominative* subjects of such verbs cannot be relativized). The CASE value of the input entry is constrained as *nominative*, to eliminate redundant lexical ambiguity (as in the case of possessor relativization of nouns in (4.65)). The structure of the relative clause in (4.50), given in (4.51), exemplifies the use of an output of this rule.

(4.69)
$$\begin{bmatrix} \text{HEAD} & verb\ [\text{VFORM}\ nomin\ [\text{CASE}\ nom]] \\ \text{SUBJ} & \langle \text{NP}\,[\text{LOC}\,\boxed{1}\,[gen]]\rangle \end{bmatrix}$$
$$\Downarrow$$
$$\begin{bmatrix} \text{SUBJ} & \langle\rangle \\ \text{INHER|RELATIVIZED|SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

Object relativization out of nominalization phrases is dealt with by (4.70), which constrains the INHER|RELATIVIZED|NON-SUBJ-REL value of the input as empty, to prevent more than one object gap in the same nominalization phrase (cf. again Section 4.3.1). See the relative clause in (4.60) and its structure in (4.61), for an example of the use of (4.70).

(4.70)
$$\begin{bmatrix} \text{HEAD} & verb\ [\text{VFORM}\ nomin\ [\text{CASE}\ nom]] \\ \text{COMPS} & \langle \ldots, \text{NP}\,[\text{LOC}\,\boxed{1}], \ldots\rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\} \end{bmatrix}$$
$$\Downarrow$$
$$\begin{bmatrix} \text{COMPS} & \langle \ldots\rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

Relativization out of Infinitive Phrases

Recall from Section 4.2.5 that relativization is possible only out of non-subject infinitive phrases in Turkish. Note that this restriction is dealt with by the *part* verb derivation rules for the second part of the long-distance relativization pattern, that is, (4.11b), which exclude VP[*inf*] from being a possible category for a subject gap host. Turning to relativization out of complement infinitive phrases, such phrases always appear as complements of 'equi' verbs, hence their subjects never get expressed. Consequently, only object relativization is possible in this case, handled by (4.71), which constrains the INHER|RELATIVIZED|NON-SUBJ-REL value of the input as empty, to rule out more than one non-subject gap in the same infinitive phrase (cf. once again Section 4.3.1).

(4.71)

$$\begin{bmatrix} \text{HEAD} & \textit{verb}\,[\text{VFORM}\ \textit{inf}] \\ \text{COMPS} & \langle ..., \text{NP}[\text{LOC}\ \boxed{1}], ... \rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \text{COMPS} & \langle ... \rangle \\ \text{INHER|RELATIVIZED|NON-SUBJ-REL} & \{\boxed{1}\} \end{bmatrix}$$

4.5 Summary

In this chapter, I have examined the issue of relativization in Turkish. In Sections 4.1 and 4.2, I focussed on empirical data, and proposed a descriptive account of relativization in Turkish, which I claim adequately characterizes the distribution of the two relativization strategies in Turkish (SPc and OPc), in cases of both bounded and long-distance relativization. Then in Section 4.3, I discussed a number of restrictions on relativization.

In Section 4.4, I presented an HPSG analysis of relativization in Turkish. The analysis exploits the identifying morphology on verbal heads of Turkish relative clauses, and is based on an assumption that those clauses have lexically specified MOD values (encoded in the lexical entry of the verbal head of the clause). In addition, it makes use of a number of relativization lexical rules which deal with participle derivation for bounded and long-distance relativization, and relativization of subjects and complements out of embedded phrases. The restrictions on relativization discussed in Section 4.3 are captured in the analysis mostly by constraints imposed on the relativization lexical rules, and in one case by a parochial constraint imposed on *finite* sentences in Turkish (cf. (4.33), page 101).

Chapter 5

Incremental Constraint-based Parsing: An Informal View

In the rest of the dissertation, I explore the computational effectiveness of an incremental processing mechanism for HPSG grammars. Section 5.1 introduces the motivation for providing incremental processing mechanisms for grammars within any linguistic theory, in general, and also for investigating the issue for a constraint-based theory of competence like HPSG, in particular. Section 5.2 then presents an introductory example, which illustrates the incremental processing of language, relying on an HPSG grammar. Then in Section 5.3, I discuss a fundamental problem that faces any incremental processing mechanism for HPSG grammars, which suggests that any such processing mechanism has to employ some sort of nonmonotonicity in order to guarantee both 'completeness' and 'termination'. In Section 5.4, I briefly mention a number of proposals made in certain unification-based grammar formalisms, which have provided a motivation for several researchers to investigate possible nonmonotonic extensions of the unification operation. I also present an overview of several such extensions proposed in the literature. Then in Section 5.5, I present an informal introduction to an incremental parsing approach for HPSG grammars, which retains unification as the underlying constraint satisfaction operation, and exploits underspecification in structure and nonmonotonicity in processing, to overcome the processing problem discussed in Section 5.3.

In the final two sections of the chapter, I turn my attention to incremental processing of a "free" word order, head-final language, specifically Turkish. In Section 5.6, I investigate how case values of constituents in a head-final language like Turkish can be used to improve the incrementality of structuring, by foreseeing clause boundaries while processing embedded constructions. And in Section 5.7, I illustrate how incremental processing enables one to capture certain preferences that humans exhibit in processing certain constructions with unbounded dependencies in a "free" word order language like Turkish.

5.1 Motivation

Sag (1995) investigates certain design properties of prospective grammars of competence that are to be directly embedded within realistic models of language processing. His observations, relying on purely intuitive evidence, suggest that such grammars should be in line with the following characteristics of human language processing:

1. Human language processing is highly incremental, which has also been shown by psycholinguistic evidence (e.g. Marslen-Wilson (1973)), as Sag points out. Humans construct a word-by-word partial representation of an utterance as they hear each word.

2. Human language processing is highly integrative. Linguistic information is integrated with non-linguistic information (e.g. world knowledge, context) anytime during processing an utterance. Again, related psycholinguistic evidence for this point comes from, for example, Marslen-Wilson and Tyler (1980), who have shown in a number of experiments that contextual information influences word recognition.

3. Human language processing is order-independent. There is no fixed order in which particular kinds of information (e.g. morphological, syntactic, contextual) are considered.[1]

[1] In relation to this point, one must note that there is actually an ongoing debate in modern psycholinguistics concerning the way different knowledge sources are invoked during human language processing: The 'modular' view claims that there is a *strict* order in which humans have access to different sources of information during language processing (cf., e.g., Frazier (1984)), whereas the 'interactive' view (which Sag agrees with in proposing the above-mentioned point) argues that language processing is one single process which *combines* information from different sources without any particular order (cf., e.g., Trueswell and Tanenhaus (1994)). (See Crocker (1996b) for an overview of certain aspects of mechanisms for human sentence processing.)

In the light of the last two points, Sag argues that performance compatibility is most likely to be achieved by a *constraint-based* grammar of competence. Clearly, HPSG, being a constraint-based theory of grammar, is in line with those two criteria: Linguistic descriptions in HPSG are represented by sign objects with a highly-integrated architecture that brings together phonological, syntactic, semantic and contextual information concerning a certain linguistic description, within one single linguistic object. In addition, the theory is purely declarative, only embodying grammars that merely consist of order-independent constraints (with no particular order to be satisfied). However, concerning the first point mentioned above, one must note that HPSG (like the other constraint-based theories such as LFG) has not yet been shown to exhibit efficient incremental satisfiability (Sag (1995)).

5.2 Incremental HPSG Parsing: An Introductory Example

Let us first consider a simple example, (5.1), to illustrate the incremental parsing of language, relying on an HPSG grammar. The structure of (5.1) is presented in (5.2).[2]

(5.1) Scully loves cats.

What we aim to achieve is the following:

i) Start the parse with an underspecified global structure, say an object of type *headed-phrase* (*hd-ph*).[3]

ii) Attach every word in the input string, (5.1), one by one – from left to right – to that global structure, thereby constraining the structure further and further as the parse progresses.

iii) At the end, come up with the *fully* specified structure in (5.2), once all the words in (5.1) are consumed (i.e. encountered and attached to the structure).

[2]To improve the readibility – and to keep the feature structures at a reasonable size – in this structure, and the ones to come, I only show the most essential feature values; note in particular that I ignore the *elist*-valued features, that is, NON-HD-DTRS or the valence features SPR, SUBJ, and COMPS with an empty value, unless they are essential to the ongoing discussion.

[3]Here I concern myself with parsing headed phrases only. In other words, I do not deal with certain structures that are assumed to be non-headed in HPSG, such as coordinate structures.

$$(5.2)\quad \left[\begin{array}{ll} \textit{hd-subj-ph} & \\ \text{PHON} & \langle \boxed{1}, \boxed{2}, \boxed{3} \rangle \\ \text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle\rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\ \text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\ \textit{Scully} \rangle \\ \text{SYNSEM} & \boxed{6}\ \text{NP}_{\boxed{8}}[nom] \end{array}\right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{2}, \boxed{3} \rangle \\ \text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle \boxed{6} \rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\ \text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{3}\ \textit{cats} \rangle \\ \text{SYNSEM} & \boxed{7}\ \text{NP}_{\boxed{9}}[acc] \end{array}\right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{word} & \\ \text{PHON} & \langle \boxed{2}\ \textit{loves} \rangle \\ \text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4}\ verb[fin] \\ \text{SUBJ} & \langle \boxed{6} \rangle \\ \text{COMPS} & \langle \boxed{7} \rangle \\ \text{CONTENT} & \boxed{5} \left[\begin{array}{ll} \textit{love} & \\ \text{LOVER} & \boxed{8} \\ \text{LOVED} & \boxed{9} \end{array}\right] \end{array}\right] \end{array}\right] \end{array}\right] \end{array}\right]$$

The global structure (henceforth, STRUCT) at the beginning of the parse is given in (5.3). The only constraints imposed on STRUCT at that stage are the ones imposed on the type *hd-ph* in the grammar. Thus, for example, the feature value types are constrained in the way they are specified in the grammar for the type *hd-ph*. In addition, the HFP, for example, constrains the HEAD value of STRUCT to be structure-shared with that of its head daughter (tag $\boxed{1}$). (Ignore for the time being the impact of the other constraints on the type *hd-ph* in the grammar, on STRUCT here.)

$$(5.3)\quad \left[\begin{array}{ll} \textit{hd-ph} & \\ \text{PHON} & \textit{list(phonstring)} \\ \text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{1}\ \textit{head} \end{array}\right] \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{sign} & \\ \text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{1} \end{array}\right] \end{array}\right] \end{array}\right]$$

When the first word, *'Scully'*, is encountered, a head-complement phrase is constructed headed by that word (and with an empty NON-HD-DTRS value), and is attached to STRUCT as a non-head daughter, resulting in the structure given in (5.4).

$$
(5.4)\quad
\begin{bmatrix}
\textit{hd-ph} & \\
\text{PHON} & \langle \boxed{1} \rangle \bullet \boxed{8} \\
\text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{4} \end{bmatrix} \\
\text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \boxed{1}\ \textit{Scully} \rangle \\ \text{SYNSEM} & \boxed{6}\ \text{NP} \end{bmatrix} \right\rangle \\
\text{HD-DTR} & \begin{bmatrix} \textit{sign} & \\ \text{PHON} & \boxed{8} \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{4} \end{bmatrix} \end{bmatrix}
\end{bmatrix}
$$

After that, the second word, *'loves'*, is again attached as the head daughter of a newly constructed head-complement phrase (this time with one non-head daughter), which is then attached as the head daughter of STRUCT, as shown in (5.5). STRUCT is now further constrained as of type *hd-subj-ph*. Note that all phrase objects in (5.5) are constrained according to the constraints imposed by the grammar on the respective types, and the constraints imposed on the lexical entries of the words in the input sentence, (5.1), that have so far been processed (i.e. *'Scully'* and *'loves'*). Thus, for example, the relevant constraint on the type *hd-subj-ph* in the grammar imposes the structure-sharing (tag $\boxed{6}$) between the SYNSEM value of the non-head daughter of the head-subject phrase (STRUCT), and the element in the SUBJ list of the head daughter of that phrase. Similarly, the respective constraint on the type *hd-comp-ph* imposes the structure-sharing (tag $\boxed{7}$) between the SYNSEM value of the non-head daughter of the head-complement phrase headed by *'loves'*, and the element in the COMPS list of the word *'loves'* – which in effect constrains the SYNSEM value of that non-head daughter (which has't yet been encountered) as NP[*acc*] at this stage of the parse.

Finally, when the last word, *'cats'*, is encountered, it is again attached as the head daughter of a newly constructed head-complement phrase (with no non-head daughters), which is then attached to STRUCT as the non-head daughter of the head-complement phrase headed by *'loves'* (constrained at the previous stage to have an NP[*acc*] SYNSEM value), yielding the final structure (5.2) presented above (cf. page 132).

This provides a fairly informal introduction to the present parsing approach. Certain details need to be further clarified, including i) what triggers the construction of a new phrase object with a certain type, or further specification of a previously constructed underspecified phrase object in the structure; and ii) what determines where a newly constructed phrase object is to be attached.

(5.5)

$$\left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{hd-subj-ph}} \\
\text{PHON} & \langle \boxed{1}, \boxed{2}, \boxed{3} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle\rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\
\text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{hd-comp-ph}} \\ \text{PHON} & \langle \boxed{1}\, \textit{Scully} \rangle \\ \text{SYNSEM} & \boxed{6}\, \text{NP}[\textit{nom}] \end{array}\right] \right\rangle \\
\text{HD-DTR} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{hd-comp-ph}} \\
\text{PHON} & \langle \boxed{2}, \boxed{3} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4} \\ \text{SUBJ} & \langle \boxed{6} \rangle \\ \text{COMPS} & \langle\rangle \\ \text{CONTENT} & \boxed{5} \end{array}\right] \\
\text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{hd-ph}} \\ \text{PHON} & \langle \boxed{3} \rangle \\ \text{SYNSEM} & \boxed{7}\, \text{NP}[\textit{acc}] \end{array}\right] \right\rangle \\
\text{HD-DTR} & \left[\begin{array}{ll}
\multicolumn{2}{l}{\textit{word}} \\
\text{PHON} & \langle \boxed{2}\, \textit{loves} \rangle \\
\text{SYNSEM} & \left[\begin{array}{ll} \text{HEAD} & \boxed{4}\, verb[fin] \\ \text{SUBJ} & \langle \boxed{6} \rangle \\ \text{COMPS} & \langle \boxed{7} \rangle \\ \text{CONTENT} & \boxed{5} \left[\begin{array}{ll} \multicolumn{2}{l}{\textit{love}} \\ \text{LOVER} & \boxed{8} \\ \text{LOVED} & \boxed{9} \end{array}\right] \end{array}\right]
\end{array}\right]
\end{array}\right]
\end{array}\right]$$

5.3 Processing Certain Recursive Constructions

In this section, I discuss a fundamental problem that faces any incremental processing mechanism for HPSG grammars (and in fact any phrase structure grammar), which essentially relates to processing certain kinds of recursive structures in a language. Consider first the English NPs in (5.6).

(5.6)
a. $[_{NP}\ [_{DetP}$ the $]\ [_{N'}$ policeman $]]$
b. $[_{NP}\ [_{DetP}$ the $]\ [_{N'}\ [_{N'}$ policeman $]\ [_{PP}$ with glasses $]]]$
c. $[_{NP}\ [_{DetP}$ the $]\ [_{N'}\ [_{N'}\ [_{N'}$ policeman $]\ [_{PP}$ with glasses $]]\ [_{PP}$ in uniform $]]]$

The problem comes from the fact that at the time of processing the N$'$ *'policeman'*, it is not possible to guess the number of PPs that may follow – and modify – that N$'$:

- It may be zero, as in (5.6a), in which case the N$'$ is simply to be attached as the head daughter of the head-specifier phrase:

$$(5.7)\quad \begin{bmatrix} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SYNSEM} & \text{DetP} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \text{N}' \end{bmatrix} \end{bmatrix}$$

- It may be one, as in (5.6b), in which case the N′ is to be attached as the head daughter of a head-adjunct phrase that is to be attached as the head daughter of the head-specifier phrase:

$$(5.8)\quad \begin{bmatrix} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SYNSEM} & \text{DetP} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \text{N}' \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

- It may be two, as in (5.6c), in which case the N′ is to be attached as the head daughter of a head-adjunct phrase that is to be attached as the head daughter of another head-adjunct phrase that is to be attached as the head daughter of the head-specifier phrase:

$$(5.9)\quad \begin{bmatrix} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SYNSEM} & \text{DetP} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \text{N}' \end{bmatrix} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

Evidently, there is potentially no limit on the level of embedding where the N′ *'policeman'* may end up being attached in the final structure, since there is no potential limit on the number of PPs that may modify that N′. Clearly, one could not assume during processing that there were an infinite number of potential attachment sites for the N′ in such cases (that is, by assuming a non-deterministic choice point with infinite branching), since one could not in that case guarantee that processing would always terminate (at least on finite input).

Let us next consider the complex Turkish sentences in (5.10), with embedded *fact* clauses. Note that the level of embedding in (5.10a,b) is two and three, respectively, and in both cases the second NP, *'Güneş'in'*, belongs to the innermost clause. Hence, the problem is the same as before: Given that there is potentially no limit on the level of embedding, an incremental parser would run into trouble while processing such examples as to where (i.e. at what level of embedding) to attach a particular constituent that is just encountered.

(5.10) a. Berfu'-ya $[_{S[fact]}$ $[_{S[fact]}$ Güneş'-in uyu-duğ-u-nu] ben-im
Berfu-DAT Güneş-GEN sleep-FACT-3sPoss-ACC I-GEN
gör-düğ-üm-ü] Yasemin söyle-di.
see-FACT-1sPoss-ACC Yasemin tell-PAST
'YASEMIN has told Berfu that I have seen that Güneş was asleep.'

b. Berfu'-ya $[_{S[fact]}$ $[_{S[fact]}$ $[_{S[fact]}$ Güneş'-in uyu-duğ-u-nu] ben-im
Berfu-DAT Güneş-GEN sleep-FACT-3sPoss-ACC I-GEN
gör-düğ-üm-ü] Mehmet'-in bil-diğ-i-ni] Yasemin
see-FACT-1sPoss-ACC Mehmet-GEN know-FACT-3sPoss-ACC Yasemin
söyle-di.
tell-PAST
'YASEMIN has told Berfu that Mehmet knows that I have seen that Güneş was asleep.'

In phrase structure grammars, such structures are licensed (derived) by rules that are traditionally called 'left-recursive', due to the fact that the category on the left hand side is exactly the same as the leftmost category on the right hand side, as in (5.11),[4] for example, which derives the recursive N' structures in (5.6) above.

(5.11) $N' \rightarrow N'\ PP$

It is a well-known fact that the possibility of left recursion constitutes a basic problem for incremental (word by word) parsing in general. Milward (1994) notes that the solution required is a way of encoding an infinite number of tree fragments, and presents an overview of different approaches in the literature to tackle the problem in that way.

[4]This is in fact the definition of 'directly' left-recursive rules. One can also have 'indirectly' left-recursive rules, as in (i):

(i) a. $NP \rightarrow DetP\ N'$

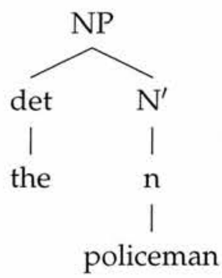
b. $DetP \rightarrow NP$'s

For example, the D-Theory parsers (Marcus *et al.* (1983)) construct descpriptions of trees rather than trees themselves. The descriptions are sets of predicates such as precedence and domination, but not immediate domination, hence the parsers are capable of capturing a certain amount of ambiguity in the structures they build.

Here I focus on a proposal made in the framework of HPSG. Konieczny (1996) presents a model of human sentence processing called the Semantics-Oriented Unification-based Language (SOUL) system, which takes the HPSG formalism as the competence base. (See also Konieczny and Hemforth (1994), and Konieczny and Strube (1995); and also Chapter 7 – Section 7.2 – of this dissertation.) Being proposed as a model of human sentence processing, SOUL has a processing mechanism that is left-to-right incremental. To deal with examples such as the ones in (5.6), the SOUL system makes use of a non-monotonic operation called 'adjoining'.[5] Let us consider, for example, the processing of (5.6b), repeated below, to illustrate how adjoining works. (I use the tree notation, preferred by Konieczny, for the structures below.) The NP *'the policeman'*, once processed, is attached to the global structure as a sign node, whose structure is given in (5.12).

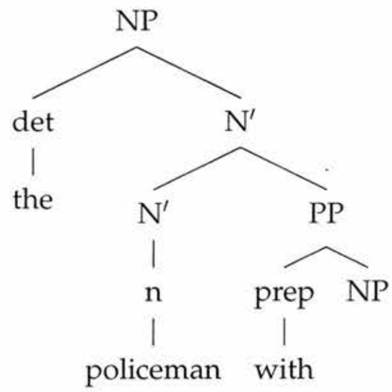
(5.6b) $[_{NP}\ [_{DetP}$ the $]\ [_{N'}\ [_{N'}$ policeman $]\ [_{PP}$ with glasses $]]]$

(5.12)



On encountering the next word *'with'*, a PP is constructed, headed by that word, which is then 'adjoined' to the NP in (5.12) in the following way: A head-adjunct structure is constructed with its head daughter set to the head daughter of the NP (i.e. the N′ node in (5.12)), and its adjunct daughter set to the newly constructed PP. The head daughter of the NP is then set (modified) to the newly constructed head-adjunct structure. The resulting structure is given in (5.13).

[5]I must note, however, that Konieczny doesn't in fact acknowledge left recursion as a problem. He proposes adjoining for processing examples of post-modification without being explicit about the reasons behind it (although he subsequently mentions in a footnote (page 215) that "since adjoining can be accomplished lazily, i.e. initiated on demand by a subsequent item that wants to be adjoined, there is no need to allow recursive head-projection in an adjunct scheme to *predict* an adjunct in advance"; see below).

(5.13) 

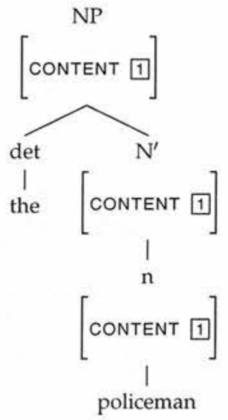
HPSG is a strictly declarative theory, meaning that it only embodies grammars whose constructs are all order-independent constraints suitable for being processed in a monotonic (information combining) fashion. A processing mechanism that makes use of nonmonotonic operations has the potential of overriding certain constraints imposed by the grammar on structures during processing, hence providing resulting structures that wouldn't in fact be accounted for by the underlying grammar.

Konieczny points out that the nonmonotonicity of adjoining is limited to the constituent structure portion in the feature structure of the sign node where that operation takes place (i.e. the mother NP node in (5.12)). However, that clearly doesn't guarantee the 'soundness' of the processing mechanism[6] in any way, since certain fields within the SYNSEM value of such a sign node may be related (e.g. via structure-sharing) to certain fields within any of its daughter values, due to particular principles in the grammar. To make this point clear, let us now have a closer look at the tree structures in (5.12) and (5.13) above, considering this time, for example, the CONTENT values of certain nodes in those structures.

The structure before adjoining takes place, (5.12), is repeated below in (5.14), with the structure-sharings imposed on the CONTENT values explicitly shown. As seen in (5.14), the mother NP node (a head-specifier structure), where the adjoining is to take place, structure-shares its CONTENT value with that of its head daughter N′ node, which is further structure-shared with the CONTENT value of the n node, due to the Semantics Principle in the grammar.

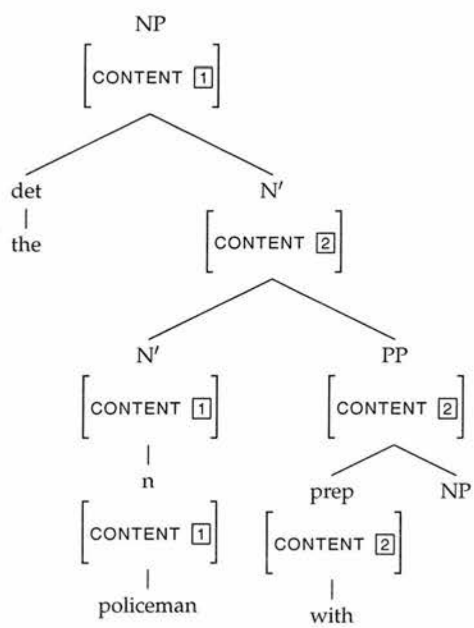
[6] Any analysis provided by the parser for a given input string should actually be licensed by the underlying grammar.

(5.14)



Let us now go back to the structure after adjoining, (5.13), which is repeated below in (5.15), again with the structure-sharings imposed on the CONTENT values explicitly shown. Notice that the structure-sharing (tag 1) between the CONTENT values of the mother NP and the N′ headed by the n node, imposed on the structure before the adjoining, is still in effect. The newly constructed (and adjoined) N′, however, being a head-adjunct structure, structure-shares (tag 2) its CONTENT value with that of its adjunct daughter (i.e. the PP node in (5.15)).

(5.15)



The point to note here is that in the resulting structure, (5.15), the CONTENT value of the mother NP node (i.e. the head-specifier structure) is *not* structure-shared with that of its head daughter (and in fact with *either* of its daughters), thereby violating the Semantics Principle in the grammar. Consequently, the structure in (5.15) is essentially *not* licensed by the underlying HPSG grammar, meaning that the adjoining operation outlined above violates the soundness of the SOUL system of Konieczny (1996).

One must bear in mind that the above illustration constitutes only one aspect of the problem here. There may well be certain other principles of the underlying HPSG grammar that are violated in the resulting structure in a similar way, because of the nonmonotonicity involved in the adjoining operation. Nevertheless, illustrating the violation of only one principle is sufficient to prove that this operation violates the soundness of the SOUL system.

From the foregoing discussion, it seems essential for an incremental parsing approach for HPSG to make use of some form of nonmonotonicity in order to guarantee both 'completeness'[7] and 'termination'.[8] In other words, the parser should be permitted to commit itself to decisions, concerning the attachment of newly encountered constituents, which it may nonmonotonically revise at subsequent stages of processing, if need arises. The point then is to ensure that the nonmonotonicity embodied in the parser doesn't have any undesirable consequences on the soundness of the overall approach.

5.4 Nonmonotonic Extensions of Unification

Constraint-based theories of grammar traditionally use feature structures to represent linguistic knowledge. A characteristic feature of feature structures is that they are suitable for being manipulated in a monotonic (information combining) manner by the unification operation. Yet, there have been several constructs in the literature proposed in certain theories whose behaviour cannot be captured in a totally monotonic fashion. Examples are default principles such as the HFC in GPSG (Gazdar *et al.* (1985)), and the priority union operation proposed by Kaplan (1987) for the analysis of ellipsis constructions such as gapping constructions in LFG (Kaplan and Bresnan (1982)). In addition, approaches to hierarchical lexicon design in unification-based formalisms have also exploited default (nonmonotonic) inheritance (which cannot be implemented by ordinary unification) (e.g. Flickinger (1987), Flickinger and Nerbonne (1992), Krieger and Nerbonne (1993)). (See Dörre *et al.* (1990), Bouma (1992), and Lascarides *et al.* (1996) for overviews and additional references in both domains.)

[7]For any given input string, the parser actually provides all possible analyses, if any, licensed by the underlying grammar.

[8]The processing eventually comes to an end.

In order to provide a well-defined formal framework for such motivations, several researchers have proposed a notion of 'default unification' which extends ordinary unification by letting default information be overriden by non-default information in feature structures, and have provided different definitions of that notion, e.g. Bouma (1992), Carpenter (1993), Copestake (1993), Russell *et al.* (1993), Young and Rounds (1993), and Lascarides *et al.* (1996). One common feature of the first four definitions of 'default unification' mentioned above is that they have been proposed as *off-line* extensions to unification-based formalisms. In Bouma (1992), for example, the applications outlined are limited to nonmonotonic template inheritance, lexical defaults and an implementation of the HFC by compilation of underspecified ID rules into fully specified rules, which can then be used by a parser relying on ordinary unification. Likewise, Carpenter (1993), Copestake (1993) and Russell *et al.* (1993) have all focussed on default inheritance in the hierarchical lexicon. Two limitations of these four definitions of default unification that restrict their application domain to a certain extent are: i) the operation is non-commutative and non-associative, hence making the result order sensitive; and ii) default information surviving a (default) unification becomes non-default, hence ruling out incremental evaluation of related default specifications (cf. Lascarides *et al.* (1996)).

Young and Rounds (1993) introduce the notion of 'nonmonotonic sorts', with default and non-default parts, which can be combined using an associative and commutative unification operation. Although their definition does not deal with prioritized defaults they state that that would be a straightforward extension. Still, there is one limitation of their definition which affects its applicability in the present context, namely that it doesn't handle default re-entrancies. An interesting application of their approach has been proposed by Strömbäck (1995), who extends the definition of nonmonotonic sorts in a way that enables the user to define nonmonotonic operations on feature structures. Strömbäck outlines various applications of that extension such as defining lexical defaults and implementing certain nonmonotonic constructs utilized in LFG such as 'any-values', 'value-constraints', and checking 'completeness' and 'coherence' conditions on f-structures.

Finally, the most recent definition of default unification comes from Lascarides *et al.* (1996), who provide an order independent default unification for typed feature structures. They call this operation 'persistent' default unification, meaning that the default information, if it survives unification, persists as being default in the resulting structure. In addition, the operation also handles default re-entrancies, as well as defining a priority relation over defaults such that default information in a feature structure with a more specific type overrides *conflicting* default information in a feature structure with a more general type (where specificity/generality is defined by the subtyping relation in the type hierarchy). Recall that in the HPSG grammar overviewed in Chapter 2 (Section 2.3), the use of the default constraints on certain types in the hierarchy also follows the framework of Lascarides *et al.* (1996). Moreover, Lascarides *et al.* suggest several motivations for using default unification outside the lexicon. The point that most relates to our case is the proposal of an implementation of the HFC interleaved with the construction of syntactic descriptions. However, they point out the difficulty of ensuring the soundness of such an approach, and leave the question open for the time being.

In short, it seems that the state of the research on nonmonotonic extensions of unification and their applications in parsing is not yet at a stage to help us overcome the problems we have discussed in the previous section concerning the incremental processing of certain recursive structures in a language. In this dissertation, I propose a more conventional solution for dealing with the recursive structures discussed in the previous section which relies on the use of *underspecification* in the structure and *nonmonotonicity* in the processing mechanism.

5.5 Underspecification and Nonmonotonicity

As we have seen in Section 5.3, the problem with employing nonmonotonic operations in the processing mechanism, in the present context, is the potential danger of overriding certain constraints imposed on the structure by the underlying grammar. To avoid that, one can take into account the fact that in HPSG any kind of *selection* is always realized via objects of type *synsem* that are structure-shared with the SYNSEM values of the daughters that are selected for. This implies that any nonmonotonic operation that affects the SYNSEM value of an object in the structure should be avoided, since it may override certain constraints imposed by the grammar on the structure at the previous stages of processing.[9] However, the fact that a nonmonotonic operation only applies to a non-SYNSEM field of a sign object does not on its own rule out the possibility of violating the soundness of the overall approach. One has to further make sure that there are no constraints imposed by the grammar on the type of the sign object in question that

[9] The selection of filler daughters is an exception to the above generalization, in that it is realized via objects of type *local* that are structure-shared with the LOCAL values of the selected filler daughters. That, however, doesn't invalidate the present argument, since the LOCAL value of a sign is essentially a part (substructure) of its SYNSEM value.

relates anything within its SYNSEM value to the field to which the nonmonotonic operation is to be applied. (See, for example, the discussion in Section 5.3 on the consequences of the 'adjoining' operation proposed by Konieczny (1996).)

Following the above discussion, only for processing purposes I modify the type hierarchy in the grammar introducing a new type called *headed-phrase-prime (hd-ph-pr)*, inserted between the types *phrase* and *hd-ph* (that is, a subtype of *phrase* and a supertype of *hd-ph*). This new type has the two appropriate features HD-DTR and NON-HD-DTRS,[10] which take values of type *sign* and *list(sign)*, as before. It is essential that there are no constraints in the grammar imposed on the type *hd-ph-pr* that relate any non-SYNSEM feature value to the SYNSEM value itself, or any other field within that value. The nonmonotonicity in processing is then limited to only the non-SYNSEM fields of the underspecified objects of type *hd-ph-pr* in the structure, to avoid overriding any constraints imposed on the structure by the grammar. In the next two sections, I present in detail the processing of the recursive structures mentioned above in Section 5.3, in the present approach.

5.5.1 Processing Recursive N's

Let us first illustrate the parse of (5.6b), repeated below, considering only the most essential steps, i.e. the processing of the N' *'policeman'* and the preposition *'with'*.

(5.6b) $[_{NP}$ $[_{DetP}$ the $]$ $[_{N'}$ $[_{N'}$ policeman $]$ $[_{PP}$ with glasses $]]]$

When the word *'policeman'* is encountered, it is attached as the head daughter of a newly constructed head-complement phrase, as before, which is then attached as the head daughter of the head daughter of the head-specifier phrase (instead of being attached as the head daughter of that head-specifier phrase; the parser is assumed to make this choice non-deterministically, of course). The structure at that point is given in (5.16). Notice that the head daughter of the head-specifier phrase is only constrained as of type *hd-ph-pr*, hence being left underspecified.

[10] That is, of course, in addition to the appropriate features it inherits from its supertypes, e.g. SYNSEM, PHON.

(5.16)

$$\left[\begin{array}{ll} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SPEC} & \boxed{1}\,\text{N}' \\ \text{HD-DTR} & \textit{word} \end{array}\right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{1} \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \text{N}' \\ \text{HD-DTR} & \textit{word} \end{array}\right] \end{array}\right] \end{array}\right]$$

When the next word *'with'* is encountered, it is again attached as the head daughter of a new head-complement phrase (with one non-head daughter), which is simply attached as a non-head daughter to the (underspecified) *hd-ph-pr* object in (5.16), giving rise to the structure in (5.17). Notice that that *hd-ph-pr* object is now further constrained as of type *hd-adjunct-ph*.

(5.17)

$$\left[\begin{array}{ll} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SPEC} & \boxed{1}\,\text{N}' \\ \text{HD-DTR} & \textit{word} \end{array}\right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \boxed{1} \\ \text{NON-HD-DTRS} & \left\langle \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{NON-HD-DTRS} & \left\langle \left[\ \text{SYNSEM} \quad \boxed{2}\,\text{NP}[\textit{acc}]\ \right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{word} & \\ \text{PHON} & \langle \textit{with} \rangle \\ \text{MOD} & \boxed{3}\,\text{N}' \\ \text{COMPS} & \langle \boxed{2} \rangle \end{array}\right] \end{array}\right] \right\rangle \\ \text{HD-DTR} & \left[\begin{array}{ll} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \boxed{3} \\ \text{HD-DTR} & \textit{word} \end{array}\right] \end{array}\right] \end{array}\right]$$

The parse of (5.6b) seems to have progressed in a totally monotonic fashion. By way of comparison, let us now illustrate the parse of (5.6c), repeated below, again considering only the most essential steps, namely the processing of the N$'$ *'policeman'*, and the two prepositions *'with'* and *'in'*.

(5.6c) $[_{NP}\,[_{DetP}$ the $]\,[_{N'}\,[_{N'}\,[_{N'}$ policeman $]\,[_{PP}$ with glasses $]]\,[_{PP}$ in uniform $]]]$

The word *'policeman'* is processed in the same way as before, resulting in the same structure, as shown in (5.18) (cf. (5.16) above).

(5.18)

$$\begin{bmatrix} \textit{hd-spr-ph} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{the} \rangle \\ \text{SPEC} & \boxed{1}\,\mathrm{N}' \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{1} \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{policeman} \rangle \\ \text{SYNSEM} & \mathrm{N}' \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \end{bmatrix} \end{bmatrix}$$

When the first preposition *'with'* is encountered, a new head-complement phrase is constructed, as before, which is then attached as the non-head daughter of a newly constructed head-adjunct phrase, as shown in (5.19). Notice that the head-complement phrase headed by *'policeman'* is now re-attached as the head daughter of that head-adjunct phrase, which is further attached as the head daughter of the *hd-ph-pr* object, *overriding* the previous value. (Again, note the non-determinism that makes the parser to end up with either (5.17) or (5.19), by attaching exactly the same word *'with'* to exactly the same structure (5.16)/(5.18).)

(5.19)

```
[ hd-spr-ph
  NON-HD-DTRS  ⟨ [ hd-comp-ph
                   PHON    ⟨the⟩
                   SPEC    [1] N'
                   HD-DTR  word ] ⟩
  HD-DTR  [ hd-ph-pr
            SYNSEM       [1]
            NON-HD-DTRS  list(sign)
            HD-DTR  [ hd-adjunct-ph
                      SYNSEM  N'
                      NON-HD-DTRS  ⟨ [ hd-comp-ph
                                       NON-HD-DTRS  ⟨ [ SYNSEM  [2] NP[acc] ] ⟩
                                       HD-DTR  [ word
                                                 PHON   ⟨with⟩
                                                 MOD    [3] N'
                                                 COMPS  ⟨[2]⟩ ] ] ⟩
                      HD-DTR  [ hd-comp-ph
                                PHON    ⟨policeman⟩
                                SYNSEM  [3]
                                HD-DTR  word ] ] ] ]
```

Finally, when the second preposition *'in'* is encountered, it again triggers the construction of a head-complement phrase, as usual, which is then attached as a non-head daughter of the (underspecified) *hd-ph-pr* object in (5.19), which is now further constrained to be of type *hd-adjunct-ph*, as shown in (5.20).

(5.20)

```
[ hd-spr-ph
  NON-HD-DTRS  ⟨ [ hd-comp-ph
                   PHON    ⟨the⟩
                   SPEC    [1]N'
                   HD-DTR  word ] ⟩
  HD-DTR  [ hd-adjunct-ph
            SYNSEM  [1]
            NON-HD-DTRS  ⟨ [ hd-comp-ph
                             NON-HD-DTRS  ⟨ [ SYNSEM [4]NP[acc] ] ⟩
                             HD-DTR  [ word
                                       PHON   ⟨in⟩
                                       MOD    [5]N'
                                       COMPS  ⟨[4]⟩ ] ] ⟩
            HD-DTR  [ hd-adjunct-ph
                      SYNSEM  [5]
                      NON-HD-DTRS  ⟨ [ hd-comp-ph
                                       NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                        PHON    ⟨glasses⟩
                                                        SYNSEM  [2]NP[acc]
                                                        HD-DTR  word ] ⟩
                                       HD-DTR  [ word
                                                 PHON   ⟨with⟩
                                                 MOD    [3]N'
                                                 COMPS  ⟨[2]⟩ ] ] ⟩
                      HD-DTR  [ hd-comp-ph
                                PHON    ⟨policeman⟩
                                SYNSEM  [3]
                                HD-DTR  word ] ] ] ]
```

Note, as an example, that in the case of a third PP modifying the N$'$ *'policeman'*, the parser would be assumed to have (non-deterministically) decided to process the second preposition *'in'* in the following way (i.e. rather than the way outlined above which results in the structure (5.20)): i) construct a head-complement phrase headed by *'in'*, as before, and attach that phrase as the head daughter of a newly constructed head-adjunct phrase, ii) re-attach the head daughter of the (underspecified) *hd-ph-pr* object in (5.19) as the head daughter of the new head-adjunct phrase; and iii) attach the new head-adjunct phrase as the head daughter of the *hd-ph-pr* object in (5.19), *overriding* the previous value. (Notice the similarity between these steps and the ones taken above while processing *'with'* that give rise to the structure (5.19).) That would then enable the parser to attach the new head-complement phrase headed by the *third* preposition as a non-head daughter of the still underspecified (*hd-ph-pr*) head daughter of the head-specifier phrase, constraining that head daughter further to be of type *hd-adjunct-ph* (cf. the processing of *'in'* above resulting in the structure (5.20)).

It is clear that the combined effect of underspecification in structure and nonmonotonicity in processing serves as a means of encoding an infinite number of feature structures. In other words, it is a way of leaving the dominance (i.e. 'daughter-of') relations underspecified in a feature structure, since a phrase object attached to the structure as a daughter of an underspecified phrase can later be re-attached as a daughter of one of its daughters. This way, a phrase attached to the structure at one point may later be lowered several times in the structure. This is essential to guarantee the completeness of the algorithm.

It should be noted that the following points need to be further clarified:

i) What determines the construction of a new phrase object of a certain type, or further specification of an already existing underspecified phrase object in the structure as an instance of a certain type?

ii) What determines where a newly constructed phrase object is to be attached?

iii) The nature of the *non-determinism* in processing: when does it arise and at what degree?

iv) What ensures that the resulting structure is fully specified, i.e. free of any underspecified objects? (This point is again essential to guarantee the soundness of the approach.)

5.5.2 Processing Recursive Ss

Let us now turn to the processing of the complex Turkish sentences mentioned in Section 5.3, which again relies on the use of underspecification in structure and nonmonotonicity in processing. The point is illustrated below with the processing of (5.10a), repeated here for convenience.

(5.10a)	Berfu'-ya	[[Güneş'-in	uyu-duğ-u-nu]	ben-im	gör-düğ-üm-ü]
	Berfu-DAT	Güneş-GEN	sleep-FACT-3sPoss-ACC	I-GEN	see-FACT-1sPoss-ACC
	Yasemin söyle-di.				
	Yasemin tell-PAST				
	'YASEMIN has told Berfu that I have seen that Güneş was asleep.'				

When the first word, *'Berfu'ya'*, is encountered, it is attached as the head daughter of a newly constructed head-complement phrase, as usual, which is further attached to STRUCT (the global structure) as a non-head daughter, as shown in (5.21). Notice that STRUCT, at this stage, is only constrained to be of type *hd-ph-pr*.

(5.21)
$$\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \textit{synsem} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu'ya} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \textit{sign} \end{bmatrix}$$

Then the second NP, *'Güneş'in'*, is again attached to STRUCT as a non-head daughter in a similar way, resulting in the structure below:

(5.22)
$$\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \textit{synsem} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Güneş'in} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu'ya} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \textit{sign} \end{bmatrix}$$

Once the first *fact* verb *'uyuduğunu'* is encountered, which subcategorizes for a genitive NP that can be realized by the most recently attached non-head daughter in STRUCT, the parser re-attaches (lowers) that NP as a non-head daughter of a newly constructed head-complement phrase (*fact* clause) headed by the new word *'uyuduğunu'*, giving rise to the structure in (5.23). Notice that at this stage the *fact* clause is attached as a sister of the first (dative) NP in the sentence.

(5.23)

$$
\begin{bmatrix}
\textit{hd-ph-pr} & \\
\text{SYNSEM} & \textit{synsem} \\
\text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \text{S}[\textit{fact, acc}] \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Güneş'in} \rangle \\ \text{SYNSEM} & \boxed{1}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{uyuduğunu} \rangle \\ \text{HEAD} & \textit{verb}[\textit{fact}] \\ \text{COMPS} & \langle \boxed{1} \rangle \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu'ya} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\
\text{HD-DTR} & \textit{sign}
\end{bmatrix}
$$

The next NP *'benim'* is again attached as a non-head daughter of STRUCT, as shown in (5.24). Then the second *fact* verb *'gördüğümü'* leads the parser to re-attach the two most-recently attached non-head daughters in STRUCT as non-head daughters of a new clause, headed by that verb, which is itself attached as a non-head daughter of STRUCT, as shown in (5.25).

After that, the parse proceeds by attaching the next *nominative* NP *'Yasemin'* as a non-head daughter of STRUCT, and finally attaching the *finite* verb *'söyledi'* as the head daughter of STRUCT, as shown in (5.26). Note that *'söyledi'* actually subcategorizes for all the non-head daughters already attached to STRUCT at that point – although not in the same order – and also that STRUCT is further constrained to be of type *hd-comp-ph* in the resulting structure.

(5.24)

$$\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \textit{synsem} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{benim} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \text{S}[\textit{fact, acc}] \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Güneş'in} \rangle \\ \text{SYNSEM} & \boxed{1}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{uyuduğunu} \rangle \\ \text{HEAD} & \textit{verb}[\textit{fact}] \\ \text{COMPS} & \langle \boxed{1} \rangle \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu'ya} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \textit{sign} \end{bmatrix}$$

(5.25)

$$\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \textit{synsem} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \text{S}[\textit{fact, acc}] \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{benim} \rangle \\ \text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \boxed{3}\,\text{S}[\textit{fact, acc}] \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Güneş'in} \rangle \\ \text{SYNSEM} & \boxed{1}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{uyuduğunu} \rangle \\ \text{HEAD} & \textit{verb}[\textit{fact}] \\ \text{COMPS} & \langle \boxed{1} \rangle \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{gördügümü} \rangle \\ \text{HEAD} & \textit{verb}[\textit{fact}] \\ \text{COMPS} & \langle \boxed{2}, \boxed{3} \rangle \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu'ya} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \textit{sign} \end{bmatrix}$$

(5.26)

$$
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{SYNSEM} & \text{S}[\textit{fin}] \\
\text{NON-HD-DTRS} & \left\langle
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{PHON} & \langle \textit{Yasemin} \rangle \\
\text{SYNSEM} & \boxed{4}\,\text{NP}[\textit{nom}] \\
\text{HD-DTR} & \textit{word}
\end{array}\right],
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{SYNSEM} & \boxed{6}\,\text{S}[\textit{fact, acc}] \\
\text{NON-HD-DTRS} & \left\langle
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{PHON} & \langle \textit{benim} \rangle \\
\text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{gen}] \\
\text{HD-DTR} & \textit{word}
\end{array}\right],
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{SYNSEM} & \boxed{3}\,\text{S}[\textit{fact, acc}] \\
\text{NON-HD-DTRS} & \left\langle
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{PHON} & \langle \textit{Güneş'in} \rangle \\
\text{SYNSEM} & \boxed{1}\,\text{NP}[\textit{gen}] \\
\text{HD-DTR} & \textit{word}
\end{array}\right]
\right\rangle \\
\text{HD-DTR} &
\left[\begin{array}{ll}
\textit{word} & \\
\text{PHON} & \langle \textit{uyuduğunu} \rangle \\
\text{HEAD} & \textit{verb}[\textit{fact}] \\
\text{COMPS} & \langle \boxed{1} \rangle
\end{array}\right]
\end{array}\right]
\right\rangle \\
\text{HD-DTR} &
\left[\begin{array}{ll}
\textit{word} & \\
\text{PHON} & \langle \textit{gördügümü} \rangle \\
\text{HEAD} & \textit{verb}[\textit{fact}] \\
\text{COMPS} & \langle \boxed{2}, \ ,\boxed{3} \rangle
\end{array}\right]
\end{array}\right],
\left[\begin{array}{ll}
\textit{hd-comp-ph} & \\
\text{PHON} & \langle \textit{Berfu'ya} \rangle \\
\text{SYNSEM} & \boxed{5}\,\text{NP}[\textit{dat}] \\
\text{HD-DTR} & \textit{word}
\end{array}\right]
\right\rangle \\
\text{HD-DTR} &
\left[\begin{array}{ll}
\textit{word} & \\
\text{PHON} & \langle \textit{söyledi} \rangle \\
\text{HEAD} & \textit{verb}[\textit{fin}] \\
\text{COMPS} & \langle \boxed{4}, \boxed{5}, \boxed{6} \rangle
\end{array}\right]
\end{array}\right]
$$

To sum up, consecutively encountered constituents are first attached as non-head daughters of the same clause, and are later re-attached (lowered) as non-head daughters of the embedded clauses once the respective heads of those clauses are encountered. Sturt and Crocker (1995) cite the following example from Inoue (1991), which suggests that native speakers of Japanese (another "free" word order, head-final language) might actually be adopting the same strategy while processing such cases.

(5.27) Bob ga Mary ni [$t_{nom/i}$ ringo wo tabeta] inu$_i$ wo ageta.
Bob NOM Mary DAT apple ACC eat-PAST dog ACC give-PAST
'Bob gave Mary the dog which ate the apple.'

> "Comprehenders report a "surprise" effect on reaching the first verb, *tabeta* ("ate"). This is explained on the assumption that the nominative, dative and accusative arguments ("Bob", "Mary" and "the apple"), are initially postulated as coarguments of the same clause, in advance of reaching the verb. On reaching the transitive verb "ate", this analysis is falsified, since this verb cannot take a dative argument." Sturt and Crocker (1995)

Nevertheless, one may argue that attaching consecutively encountered constituents to the same clause while processing such structures may not be justified from a psycholinguistic viewpoint in cases where certain clues in the language being parsed reveal that those constituents could actually not belong to the same clause. In languages such as Turkish and Japanese, for example, one can exploit the CASE values of constituents to improve the incrementality of the parser by foreseeing clause boundaries while processing embedded constructions even before encountering their verbal heads. In the next section, I elaborate further on this point for the case of Turkish.

5.6 Exploiting the CASE Values

This section investigates how one can improve the incrementality of structuring in the present parsing approach for a head-final language such as Turkish, where CASE values of constituents play an important role in signalling their grammatical role. Consider, for example, the complex Turkish sentence in (5.28), where a number of consecutive NPs that belong to different clauses one embedded in another, are followed by the verbs that select for them at the end of each clause.

(5.28) Ben $[_{S[fact]}$ adam-ın $[_{S[fact]}$ kadın-ın yürü-düğ-ü-nü]
I man-GEN woman-GEN walk-FACT-3sPoss-ACC
gör-düğ-ü-nü] bil-iyor-um.
see-FACT-3sPoss-ACC know-PROG-1SG
'I know that the man sees that the woman walks.'

In Section 5.5.2, we saw that in the present approach the first three NPs (the first one nominative, the other two genitive) would first be attached as non-head daughters of the same clause, and the second and third NPs would later be re-attached (lowered) as non-head daughters of the *fact* clauses to which they actually belong, only after encountering the respective verbal head of each clause. It is, however, possible in certain cases, such as (5.28), to foresee the existence of an embedded clause before encountering its verbal head, by taking into account certain restrictions on the co-occurrence of NPs with particular case-markers in a clause. For example, knowing that a specific, nominative NP and a genitive NP, or two genitive NPs cannot be immediate constituents of the same clause (see below), one can construct the embedded clauses in (5.28) at the time of processing the genitive NP subject of each clause, attaching that NP as the non-head daughter of that clause. In the rest of this section, I concern myself with certain clues the parser might benefit from in this way while processing Turkish.

For convenience, I repeat here that Turkish has the following seven morphological cases: nominative (morphologically unmarked), genitive (marked with the suffix '*-(n)In*'), accusative (marked with '*-(y)I*'), dative (marked with '*-(y)E*'), ablative (marked with '*-(n)dEn*'), locative (marked with '*-(n)dE*'), and comitative/instrumental (marked with '*-(y)lE*'). Below I summarize the functions each of these cases signals for a phrase carrying the respective case-marker (no marker in the case of nominative case).

5.6.1 Genitive and Nominative Phrases

A genitive phrase can be either of the following: i) the subject of a non-finite sentence; or ii) the possessor of a possessive NP.

A nominative NP can have either of the following functions: i) the subject of a (finite or non-finite) sentence ii) the unmarked object of a sentence. Note that in the latter case the NP cannot be one of the 'implicitly specific' NPs, such as pronouns or proper nouns (see Chapter 3, Section 3.2, for a discussion of this restriction on the case-marking of direct objects). Note also that again in the latter case the NP must be in the immediately preverbal position of the sentence (see Section 3.4.1 for a discussion of this restriction on word order). Finally, a nominative S or VP can only function as the subject of a sentence.

Keeping these functions in mind, one can then impose the following restrictions on the parser with respect to the attachment of phrases with CASE values of type *nominative* or *genitive*, as immediate constituents of the same clause.

On encountering a new phrase with *genitive* CASE value, if there is already a *nominative* or *genitive* phrase attached to the clause where the new phrase is expected to be attached to, the parser can infer that that new phrase is either the possessor of an embedded possessive NP or the subject of an embedded clause, hence in neither case can it be an immediate sister of the (subject) *nominative/genitive* phrase that has previously been attached to the current clause. So, for example, while processing any of the sentences in (5.29), the parser can construct the embedded phrase (the possessive NP in (5.29a,c), and the embedded clause in (5.29b,d) – the inner one in (5.29d)) at the time of processing the genitive NP *'benim'*, and attach that NP as a non-head daughter of that embedded phrase. Note that in the case of any non-head daughters intervening between those two non-head daughters in question (such as *'kitabı'* in (5.29)) the parser should *non-deterministically* decide where to attach those intervening daughters since they might turn out to belong to either of the outer phrase (cf. (5.29a,c)) and the embedded one (cf. (5.29b,d)).

(5.29) a. *Kadın* kitab-ı [*ben-im* oda-m-a] götür-dü.
woman book-ACC I-GEN room-1sPoss-DAT take-PAST
'The woman took the book to my room.'

b. *Kadın* [kitab-ı *ben-im* oku-duğ-um-u] san-dı.
woman book-ACC I-GEN read-FACT-1sPoss-ACC think-PAST
'The woman thought that *I* was reading the book.'

c. [*Kadın-ın* kitab-ı [*ben-im* oda-m-a] götür-me-si] ben-i
woman-GEN book-ACC I-GEN room-1sPoss-DAT take-ACT-3sPoss I-ACC
şaşırt-tı.
surprise-PAST
'I was surprised that the woman took the book to my room.'

d. [*Kadın-ın* [kitab-ı *ben-im* oku-duğ-um-u] san-ma-sı]
woman-GEN book-ACC I-GEN read-FACT-1sPoss-ACC think-ACT-3sPoss
ben-i şaşırt-tı.
I-ACC surprise-PAST
'I was surprised that the woman thought that *I* was reading the book.'

Similarly, if there is already a *nominative* or *genitive* phrase attached as a non-head daughter of the current clause, such as *'kadın'* in (5.30a) or *'kadının'* in (5.30b), then, on encountering a *specific, nominative* NP, such as *'ben'* in both (5.30a,b), the parser can infer that the second NP can only be the subject of a 'new' embedded clause, and hence can construct that embedded clause (the inner one in the case of (5.30b)) at that point, attaching the second NP as a non-head daughter of that clause.

(5.30) a. *Kadın* [kitab-ı *ben* oku-yor-um] san-dı.
woman book-ACC I read-PROG-1SG think-PAST
'The woman thought that *I* was reading the book.'

b. [*Kadın-ın* [kitab-ı *ben* oku-yor-um] san-ma-sı] ben-i
woman-GEN book-ACC I read-PROG-1SG think-ACT-3sPoss I-ACC
şaşırt-tı.
surprise-PAST
'I was surprised that the woman thought that *I* was reading the book.'

If, however, the second (*nominative*) NP is *not specific*, then the parser should not be allowed to make any such predictions at the time of processing that NP (since it may turn out to be a *nominative* object in the same clause as the previously attached (subject) NP, as in (5.31a,b)) even though it may still end up being attached to an embedded clause as in (5.31c,d).

(5.31) a. *Mehmet kitap* yaz-dı.
Mehmet book write-PAST
'Mehmet wrote a book/books.'

b. [*Mehmet'-in kitap* yaz-ma-sı] ben-i şaşırt-tı.
Mehmet-GEN book write-ACT-3sPoss I-ACC surprise-PAST
'I was surprised that Mehmet wrote a book/books.'

c. *Mehmet* [*kitap* bit-ti] san-dı.
Mehmet book finish-PAST think-PAST
'Mehmet thought that the book was finished.

d. [*Mehmet'-in* [*kitap* bit-ti] san-ma-sı] ben-i şaşırt-tı.
Mehmet-GEN book finish-PAST think-ACT-3sPoss I-ACC surprise-PAST
'I was surprised that Mehmet thought that the book was finished.

Finally, a *nominative* S or VP following a *nominative/genitive* phrase can again enable the parser to construct the embedded clause to which the second phrase is to be attached, at the time of processing that phrase. Consider, for example, the processing of (5.32), with two embedded clauses, the outer one being *finite*, and the inner one an *act* clause whose genitive subject *benim* has been long-distance backgrounded in the main sentence. On encountering the *act* verb *'okumam'* (with no case-marking at all, hence *nominative*), since there is already a nominative NP, *'kadın'*, attached to the current clause, the parser, in addition to constructing the inner clause headed by *'okumam'*, can also construct the outer embedded clause (whose head, *'duyuldu'*, is yet-to-come), and attach the (inner) *act* clause as a non-head daughter of that outer clause.

(5.32) [*Kadın* [[$\_{i}$ *kitab-ı* *oku-ma-m*] duy-ul-du] san-dı] ben-im$_i$.
woman book-ACC read-ACT-1sPoss hear-PASS-PAST think-PAST I-GEN
'The woman thought that it was heard that I was reading the book.'

5.6.2 Accusative and Dative Phrases

Accusative and dative phrases function as complements of verbal (and occasionally nominal) heads. Given that lexical heads do not normally subcategorize for more than one complement with the same CASE value, it is perhaps not surprising that one does not come across examples, in Turkish, where more than one accusative marked phrase or more than one dative marked phrase occur as immediate constituents of the same clause. However, there are two rather interesting facts which suggest that Turkish clauses are indeed restricted not to have more than one accusative marked phrase or more than one dative marked phrase as immediate constituents.

The first fact is related to causativization in Turkish, which is achieved by the suffixation of either of the suffixes *'-dIr'* and *'-t'* to an intransitive or transitive verb. The point is that in the case of a transitive verb already subcategorizing for an *accusative* complement, causativization adds an additional *dative* complement to the verb's argument list, as in (5.33), whereas in the case of a transitive verb already subcategorizing for a *dative* complement, it adds an additional *accusative* complement to the arguments, as in (5.34) (cf. Knecht (1986)).

(5.33) a. Berfu çocuğ-a inan-dı.
Berfu child-DAT believe-PAST
'Berfu believed the child.

b. Mehmet Berfu-yu çocuğ-a inan-dır-dı.
Mehmet Berfu-ACC child-DAT believe-CAUS-PAST
'Mehmet made Berfu believe the child.'

c. * Mehmet Berfu-ya çocuğ-a inan-dır-dı.
Mehmet Berfu-DAT child-DAT believe-CAUS-PAST

(5.34) a. Berfu kitab-ı al-dı.
Berfu book-ACC buy-PAST
'Berfu bought the book.'

b. Mehmet Berfu-ya kitab-ı al-dır-dı.
Mehmet Berfu-DAT book-ACC buy-CAUS-PAST
'Mehmet made Berfu buy the book.'

c. * Mehmet Berfu-yu kitab-ı al-dır-dı.
Mehmet Berfu-ACC book-ACC buy-CAUS-PAST

The second point is related to the use of benefactives in Turkish. Although Turkish benefactatives are in general expressed by PPs headed by the postposition *için 'for'*, it is also possible to use a dative NP for that purpose, unless there is already a dative constituent in the clause. So, for instance, the PP *'adam için'* in (5.35a) can readily be replaced by the *dative* NP *'adama'*, as seen in (5.35b), whereas this is not possible in (5.36a), as seen in (5.36b), since there is already a *dative* NP, *'hastaneye'*, in the clause.

(5.35) a. Kadın [adam için] bir hediye al-dı.
woman man for a present buy-PAST

b. Kadın adam-a bir hediye al-dı.
woman man-DAT a present buy-PAST
'The woman bought a present for the man.'

(5.36) a. Kadın [adam için] hastane-ye git-ti.
woman man for hospital-DAT go-PAST
'The woman went to the hospital for the man.'

b. * Kadın adam-a hastane-ye git-ti.
woman man-DAT hospital-DAT go-PAST

One can then safely assume that Turkish clauses do exhibit a restriction which prevents them from having more than one accusative phrase or more than one dative phrase as immediate constituents.

An incremental parser can therefore exploit this restriction by constructing an embedded clause whenever it encounters an *accusative/dative* phrase, if there is already a non-head daughter with the same CASE value attached to the current clause. Thus, for instance, while processing (5.37a) the parser can construct the embedded clause right after encountering the second *accusative* NP *'çocuğu'*. Likewise, in (5.37b), with two embedded *fact* clauses, the parser can construct both clauses upon encountering the first *fact* verb, dative marked *'uyuduğuna'* (since there is already a *dative* NP, *'Güneş'e'*, attached to the current clause), and attach the inner S[*fact*] as a non-head daughter of the outer one, which is further attached as a sister of the *dative* NP *'Güneş'e'*.

(5.37) a. *Berfu-yu* [Mehmet-in *çocuğ-u* gör-düğ-ü-ne] ben
Berfu-ACC Mehmet-GEN child-ACC see-FACT-3sPoss-DAT I
inan-dır-dı-m.
believe-CAUS-PAST-1SG
'I made Berfu believe that Mehmet has seen the child.'

b. *Güneş'-e* [[*çocuğ-un* *uyu-duğ-u-na*] Berfu-yu ben-im
Güneş-DAT child-GEN sleep-FACT-3sPoss-DAT Berfu-ACC I-GEN
inan-dır-dığ-ım-ı] Mehmet söyle-di.
believe-CAUS-FACT-1sPoss-ACC Mehmet tell-PAST
'Mehmet has told Güneş that I made Berfu believe that the child was asleep.'

5.6.3 Ablative, Locative and Instrumental Phrases

Ablative, locative and instrumental phrases in Turkish may function as complements or adjuncts of sentences. As seen in (5.38), one can have examples with more than one constituent with the same CASE value in the same clause, for all three cases. Therefore, an incremental parser cannot exploit these three cases in the same way as it exploits the other cases as discussed above.

(5.38) a. *Okul-dan* tek temsilci *son sınıf-lar-dan* çık-tı.
school-ABL only representative final year-PLU-ABL come out-PAST
'From the school the only representative came out from the final years.'

b. *Türkiye'de* en sıcak gün-ler *Güney-de* yaşa-n-ır.
Turkey-LOC most hot day-PLU south-LOC experience-PASS-PRES
'In Turkey, the hottest days are experienced in the South.'

c. Mehmet *büyük bir dikkat-le* kağıd-ı *makas-la* kes-ti.
Mehmet great a care-INS paper-ACC scissors-INS cut-PAST
'Mehmet, with great care, cut the (piece of) paper with scissors.'

5.7 Long-distance Topicalization: An Incremental View

In Chapter 3 (Section 3.4.2), we saw that constituents of embedded clauses in Turkish can be readily extracted out of their clauses, occurring in the sentence-initial or post-predicate positions of other (surrounding) clauses. In (5.39a,b), for example, the *genitive* NP subject of the embedded S[*fact*] occurs in the sentence-initial and post-predicate positions of the matrix clause, respectively. Recall that I have dubbed the former case long-distance topicalization and the latter long-distance backgrounding.

(5.39) a. Adam-ın$_i$ [ben san-a [$\_i$ kadın-ı gör-düğ-ü-nü]
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC
söyle-di-m.]
tell-PAST-1SG
'As for the man, I have told you that he has seen the woman.'

b. [Ben san-a [$\_i$ kadın-ı gör-düğ-ü-nü] söyle-di-m]
I you-DAT woman-ACC see-FACT-3sPoss-ACC tell-PAST-1SG
adam-ın$_i$.
man-GEN
'I have told you that he, the man, has seen the woman.'

Morphological case-marking enables Turkish speakers to recover the actual filler-gap relations in long-distance dependencies rather easily in most cases, and in the case of long-distance topicalization, at the time of processing the head whose argument has been topicalized. While processing (5.39a), for example, as soon as encountering the embedded verb *'gördüğünü'*, Turkish speakers have a very strong tendency to link the sentence-initial genitive NP *'adamın'* to that verb, and hence interpret that NP as the "missing" subject of the embedded clause. Long-distance topicalization therefore constitutes an interesting case from the viewpoint of incremental processing, in that an incremental parser for Turkish can exploit this strong tendency by linking the filler NP to the embedded verb in the same way – as soon as that verb is encountered – and constructing the corresponding head-filler phrase right at that point, before the main verb is encountered (i.e. before the head daughter of that head-filler phrase is fully specified).

Recall from Chapter 2 (Section 2.2, pages 22–24) that in this dissertation, I assume the traceless account of unbounded dependency constructions proposed in Chapter 9 of Pollard and Sag (1994), where a nonlocal dependency is first introduced in the structure by the non-empty INHER|SLASH value of the lexical head that licenses the missing element. Consequently, in the present approach, which incorporates the strategy mentioned above, a lexical head with a non-empty INHER|SLASH value (e.g. *'gördüğünü'* in (5.39)) triggers a search in the structure to see if the missing element can be realized by a phrase that has already been attached to the structure (e.g. the genitive NP *'adamın'* in (5.39a)). (Details of this search process are discussed in Chapter 6, Section 6.6.)

It should be noted, however, that recovering the head-filler dependencies in this way, i.e. as soon as a lexical head with a non-empty INHER|SLASH value is encountered, is in no way guaranteed to provide all available readings in certain cases of ambiguity. Consider, for instance, (5.40), with the first five words the same as in (5.39a).

(5.40) a. [Adam-ın$_i$ [ben san-a [[$\_i$ kadın-ı gör-düğ-ü-nü] çocuğ-un
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC child-GEN
bil-diğ-i-ni] söyle-di-m.]]
know-FACT-3sPoss-ACC tell-PAST-1SG
'As for the man, I have told you that THE CHILD knew that he has seen the woman.'

b. [Adam-ın$_i$ [ben san-a [$\_i$ [[$\_j$ kadın-ı gör-düğ-ü-nü]
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC
çocuğ-un$_j$] bil-diğ-i-ni] söyle-di-m.]]
child-GEN know-FACT-3sPoss-ACC tell-PAST-1SG
'As for the man, I have told you that he knew that she, the child, has seen the woman.'

(5.40a) is a reading where an attempt of linking the sentence-initial genitive NP *'adamın'* to the first embedded verb *'gördüğünü'* (as soon as that verb is encountered) would give rise to the actual analysis. (5.40b), on the other hand, is an alternative reading where the missing subject of the embedded clause headed by *'gördüğünü'* is in fact in the post-verbal position of that clause (i.e. backgrounded), and the sentence-initial *'adamın'* is in fact the long-distance topicalized subject of another embedded clause headed by the second *fact* verb *'bildiğini'*. Intuitive evidence suggests that native speakers disambiguate between these two readings by the help of prosodic clues, and in the absence of prosodic information, (5.40a) is easier by far to recover. Thus, a serial parser, for example, should provide that analysis first, and the less preferred one only on backtracking.

For expository reasons, I have so far indicated filler-gap relations in long-distance dependencies by using identically numbered subscripts for the fillers and their respective gaps, assuming the gap position in the typical word order. Note, however, that this is not entirely faithful to the assumption (mentioned above) that a nonlocal dependency is first introduced in the structure by the non-empty INHER|SLASH value of the lexical head licensing the missing element. In fact, Pollard and Sag (1994) motivate the traceless account of UDCs based on the results of Pickering and Barry (1991), which suggest a *gap-free* analysis of UDCs where a filler is directly associated with its subcategoriser, rather than a respective gap position. That point is essential to determining the dependency patterns (i.e. nested vs. intersecting) in certain ambiguous cases, such as (5.41), in Turkish. Notice that in (5.41a) the two fillers are associated with two corresponding gap positions (in the typical order), as before, whereas in (5.41b) each filler is associated with its respective subcategorizer.

(5.41) a. [Adam-ın$_{i/j}$ [kadın-ın$_{j/i}$ [çocuğ-a [$\_i$ [$\_j$ kitab-ı
man-GEN woman-GEN child-DAT book-ACC
gör-düğ-ü-nü] bil-diğ-i-ni] ben söyle-di-m.]]]
see-FACT-3sPoss-ACC know-FACT-3sPoss-ACC I tell-PAST-1SG

b. [Adam-ın$_{i/j}$ [kadın-ın$_{j/i}$ [çocuğ-a [[kitab-ı gör-düğ-ü-nü$_j$]
man-GEN woman-GEN child-DAT book-ACC see-FACT-3sPoss-ACC
bil-diğ-i-ni$_i$] ben söyle-di-m.]]]
know-FACT-3sPoss-ACC I tell-PAST-1SG
'I have told the child that the man/woman knew that the woman/man has seen the book.'

The point to note is that in (5.41a) the 'man-see/woman-know' dependency pattern is nested, and the 'woman-see/man-know' pattern is intersecting, whereas in (5.41b) it is the other way around. Whatever dependency pattern (nested/intersecting) one assumes for such potentially ambiguous examples, the fact is that Turkish speakers have a very strong tendency for the 'woman-see/man-know' dependency, as opposed to the 'man-see/woman-know' dependency.[11] An incremental parser can capture this preference in the process of filling a missing element introduced by a newly encountered lexical head, by giving preference to a potential filler that has been attached to the structure more recently than the others. Thus, for example, in the case of (5.41), a serial parser should first try to fill the gap introduced by the first *fact* verb *'gördüğünü'* with

[11]It is rather tempting to argue, on the basis of the Nested Dependency Constraint (NDC) of Fodor (1978) (cf. (i)), that the association in (5.41b) is more plausible, since the nested dependency pattern according to that association is indeed the preferred reading.

(i) Nested Dependency Constraint: If there are two or more filler-gap dependencies in the same sentence, their scopes may not intersect if either disjoint or nested dependencies are compatible with the well-formedness conditions of the language.

Note, however, that the NDC itself relies on *filler-gap* associations, and in fact (5.41a) seems to constitute a counterexample for that constraint, if one assumes the gap position in the typical word order. One may, of course, argue, relying on the "free" nature of the word order in Turkish, for a marked gap position, as in (ii), which would then associate the preferred reading with a nested dependency. However, considering the pragmatic function of word order variation in Turkish, such an argument would seem to be somewhat weak.

(ii) [Adam-ın$_{i/j}$ [kadın-ın$_{j/i}$ [çocuğ-a [[$\_j$ kitab-ı gör-düğ-ü-nü] $\_i$
man-GEN woman-GEN child-DAT book-ACC see-FACT-3sPoss-ACC
bil-diğ-i-ni] ben söyle-di-m.]]]
know-FACT-3sPoss-ACC I tell-PAST-1SG
'I have told the child that the man/woman knew that the woman/man has seen the book.'

the more recently attached genitive NP *'kadının'*, hence favouring the preferred reading, and should provide the other parse only on backtracking.

5.8 Summary

In this chapter, I have presented an informal introduction to an incremental parsing approach for HPSG grammars. I argued, in Section 5.3, that the possibility of left recursive structures in a language would force any such parsing approach to employ some sort of nonmonotonicity in order guarantee both completeness and termination. In Section 5.4, I presented an overview of several nonmonotonic extensions of unification, and argued that the state of the research on such extensions is not yet at a stage to help us with the present case.

In Section 5.5, I proposed an incremental parsing approach for HPSG grammars, which retains unification as the underlying constraint satisfaction operation, and makes use of underspecification in structure and nonmonotonicity in processing. Crucially, the nonmonotonicity of processing is limited to only the non-SYNSEM fields of underspecified objects in the structure, to avoid overriding any constraints imposed on the structure by the underlying HPSG grammar at the previous stages of processing, hence retaining the soundness of the overall approach.

In the final two sections of the chapter, I focussed on certain issues that arise in incremental processing of a "free" word order, head-final language like Turkish. First, in Section 5.6, I investigated how the parser can benefit from the CASE values in Turkish in foreseeing the existence of an embedded phrase/clause before encountering its head. Then in Section 5.7, I presented an incremental view of long-distance topicalization in Turkish, which enables the parser to incrementally recover the long-distance dependencies while processing such structures, as soon as encountering a lexical head with a "missing" argument. I pointed out that such incremental recovery of long-distance dependencies further enables one to capture certain (strong) preferences that humans exhibit in processing certain cases with potentially ambiguous long-distance dependency relations.

Chapter 6

Parsing Approach

In this chapter, I discuss the incremental parsing approach for HPSG grammars introduced in Chapter 5, in more detail. First, I outline the general characteristics of the approach. Then in Section 6.1, I present the main body of the parsing algorithm. In Section 6.2, I focus on the process of attaching a newly encountered word in the input to the global structure, and provide a general description of that process.

In Section 6.3, I present the attachment procedures I have developed for Turkish, which also take into account the linear precedence constraints proposed for Turkish in Chapter 3 (Section 3.4.5). In Section 6.4, I present two example parses, which illustrate the use of some of the routines presented in Section 6.3, by the parser. Then in Section 6.5, I discuss the details of the strategy of exploiting the CASE values in incrementally structuring embedded constructions (cf. Chapter 5, Section 5.6), together with an example parse. Finally, in Section 6.6, I present the details of incremental processing of long-distance topicalization, first introduced in Chapter 5 (Section 5.7), again complementing the discussion with an example parse.

As mentioned in Chapter 5 (Section 5.2), parsing a string in the present approach always starts with an underspecified global structure, and proceeds by attaching every word in the input string one by one to that global structure, thereby constraining the structure further and further at each step (i.e. with the processing of each word). Once all the words in the input are consumed, the parser returns the global structure as its output, which – to be an acceptable one – is required to be *fully* specified.

I present a totally 'transparent' approach for parsing HPSG grammars, one that works directly on the representations provided by the HPSG formalism.[1] To that end, it makes systematic use of the selection features governed by the HPSG formalism such as the valence features SPR, SUBJ and COMPS, and the head selection features MOD and SPEC, in determining the type of phrase that a particular word (that is, the one just encountered) may be a part of.[2] In addition, in the process of attaching a word to the structure, the parser also takes into account the linear precedence constraints employed by the language being parsed.

6.1 Parsing Algorithm

At the beginning of the parse, the (global) structure of the as yet unencountered input string, STRUCT, is constrained as an object of type *headed-phrase-prime (hd-ph-pr)*. The feature structure for STRUCT at that stage of the parse is given in (6.1).

(6.1)

$$\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{PHON} & \textit{list(phonstring)} \\ \text{SYNSEM} & \begin{bmatrix} \textit{synsem} & \\ \text{LOCAL} & \begin{bmatrix} \textit{local} & \\ \text{CAT} & \begin{bmatrix} \textit{cat} & \\ \text{HEAD} & \textit{head} \\ \text{MARKING} & \textit{marking} \\ \text{SPR} & \textit{list(synsem)} \\ \text{SUBJ} & \textit{list(synsem)} \\ \text{COMPS} & \textit{list(synsem)} \end{bmatrix} \\ \text{CONTENT} & \textit{content} \\ \text{CONTEXT} & \textit{context} \end{bmatrix} \\ \text{NONLOCAL} & \textit{nonlocal} \end{bmatrix} \\ \text{NON-HD-DTRS} & \textit{list(sign)} \\ \text{HD-DTR} & \textit{sign} \end{bmatrix}$$

During the processing of a single word, every *special* selection feature-value pair (see Section 6.2) in the lexical entry of that word non-deterministically triggers either of the following steps:

[1]Berwick and Weinberg (1984) introduce the notion of 'type transparancy', suggesting that cognitively plausible parsers work directly on the representations provided by grammars. Here I use the word 'transparent' without making any strong claims about the cognitive plausibility of the present parsing approach.

[2]In that respect, the present approach bears a certain resemblance to the work on compilation of HPSG grammars into feature-based TAGs by Kasper *et al.* (1995), which also exploits the selection features in the HPSG formalism in projecting the lexical types of HPSG grammars to the elementary trees of the TAG formalism.

i) **Phrase specification:** The further specification of a *hd-ph-pr* object in STRUCT, as an instance of the phrase type signified by the selection feature in question.

ii) **Phrase construction:** The construction of a new phrase object of the type signified by the selection feature in question, which is then attached to STRUCT as a daughter of one of the underspecified objects.

Given that there may be an arbitrary number of embeddings in the structure (cf. the examples in Chapter 5), one needs a way of keeping track of the underspecified phrase objects in the structure for the purposes of both steps mentioned above. To that end, I make use of a stack of phrases, which I call predicted phrases, PRED(ICTED)-PHRASES. Elements on PRED-PHRASES are simply pointers to the underspecified sub-structures of STRUCT that are predicted by some already encountered daughter, e.g. complement daughters predicted by lexical heads, and head daughters predicted by specifiers, adjuncts or markers. Only one of those predicted daughters – the one on the top of the stack – is considered to be 'active' at any given point in the course of the parse, meaning that the parser can only further specify that underspecified phrase, or attach a newly constructed phrase as a daughter of only that phrase.

A phrase specification step always leads the parser to pop the active phrase, which is now fully specified as an instance of one of the most specific phrase types (e.g. *hd-comp-ph, head-spr-ph*, etc.), off the stack (to avoid overriding any constraints imposed on the structure by the grammar). Moreover, a phrase specification/construction step may further lead the parser to push new phrases onto the stack that are predicted by the word just being processed.

PRED-PHRASES, at the beginning of the parse, is initialized to contain a pointer to the global structure, STRUCT, the only attachment site available at that stage. At the end of the parse, PRED-PHRASES must be empty for the string to be grammatical, which guarantees that any predicted phrases in the course of the parse will indeed have been encountered by the end. It should be noted that any underspecified phrase during processing is pushed onto the stack, and that a phrase is popped off the stack only after (and as soon as) it is fully specified (i.e. constrained as an instance of one of the most specific phrase types). Consequently, requiring the stack to be empty guarantees that any underspecified phrase in the course of the parse will have been fully specified by the end. Note also that although STRUCT itself may be popped off the stack at some point before the end of the parse, any change caused by the attachment process to the elements on the stack even after that point will have an immediate effect on STRUCT, since those elements are essentially pointers to certain sub-structures of STRUCT.

The main body of the parsing algorithm and the grammaticality principle are presented in (6.2) and (6.3), respectively. 'ND' stands for 'non-deterministically', and 'STACK (X)' is a function that returns a stack whose only element is a pointer to object X. Note that the non-determinism at step (6.2ci) is due to the possibility of lexical ambiguity in the language, and the one at step (6.2cii) is due to the choice the parser is supposed to make between a phrase specification step and a corresponding phrase construction step, as mentioned above. The algorithm doesn't commit itself to any particular strategy to deal with the non-determinism introduced by these two steps.

(6.2) **Main Body:**

a. Constrain STRUCT as an object of type *hd-ph-pr*.

b. Initialize PRED-PHRASES to **STACK** (STRUCT).

c. For each WORD encountered do

  i. (ND) Fetch lexical entry NewWord for WORD from LEXICON.

  ii. (ND) Attach NewWord (to STRUCT).

d. Return STRUCT.

(6.3) **Grammaticality Principle:**

At the end of the parse (i.e. once all the words in the input string are processed), PRED-PHRASES must be *empty*.

Although the main body itself is general, the attachment process at step (6.2cii) varies for languages with different word order properties as a result of the fact that those languages employ different linear precedence constraints to account for the word order restrictions they exhibit. Accordingly, one would expect the attachment process to be similar for languages with similar word order properties. Although this is true in general, it is further possible to tune that process to a particular language, considering certain other features of that language, a point that I discuss in Sections 6.5 and 6.6 for the case of Turkish.

Following the notion of 'dynamics in algorithm development' introduced by Milward (1994), one can view the parse of a given input string using this algorithm in a dynamic way, as a sequence of states, where a pair of consecutive states represents a transition from the former state in the pair to the one following, by the complete processing of a single word in the input string (cf. step (6.2c) in the main body above). Each state is then composed of the values of the global structure, STRUCT, and the stack of predicted phrases, PRED-PHRASES, at a certain stage of the parse. In the initial state of the parse, STRUCT is only constrained as an object of type *hd-ph-pr*, and PRED-PHRASES only contains a pointer to STRUCT. In the final state, PRED-PHRASES must be empty for the string to be grammatical (and STRUCT will have been constrained as an instance of one of the most specific phrase types, since that is the only way it may have been popped off the stack).

6.2 Attachment Process

As mentioned above, the parser benefits from the selection features in the formalism to determine the type of the phrase that a particular word with a particular selection feature-value pair may be a part of. So, for example, a non-empty SUBJ value in the new word's lexical entry indicates that a phrase headed by that word (i.e. a projection of the word) is to function as the head daughter in a head-subject phrase. Similarly, a non-empty SPR value signals (for a projection of the new word) a head daughter role in a head-specifier phrase. In addition to these valence features, via which heads select for their arguments, the HPSG formalism also equips certain non-head daughters such as adjuncts, specifiers and markers with special features that enable them to select for the heads they are to modify or specify. Adjuncts, for instance, select for their heads via the MOD feature, for which they always have a value of type *synsem* that is to be structure-shared with the SYNSEM value of the head daughter in a head-adjunct phrase. Likewise, specifiers and markers select for their heads via a *synsem*-valued SPEC feature. Consequently, a *synsem*-valued MOD feature may be considered as a sign for a non-head daughter role in a head-adjunct phrase, and a *synsem*-valued SPEC feature as a sign for a non-head daughter role in a head-specifier or a head-marker phrase. The point to note is that, depending on the word order restrictions a certain language exhibits, a particular special selection feature-value pair X (such as the ones mentioned above) should lead the parser to either of the following two alternatives:

i) If the daughter bearing X is expected to follow the daughter that it selects for via X (according to the word order restrictions in the language), then, on encountering a word with X, the parser should infer that the daughter that is being selected for must have already been encountered, and hence no further prediction is necessary.

ii) If the daughter bearing X is expected to precede the daughter it selects for via X, then processing a word with X should lead the parser to predict the daughter that is selected for, by pushing it onto the stack of predicted phrases.

So, for example, in a head-final language where arguments and adjuncts always precede their heads, a *synsem*-valued MOD feature enables the parser to predict the head daughter at the time of processing the adjunct daughter. In the case of a non-empty SUBJ value, on the other hand, the parser infers that a phrase that has already been attached to STRUCT is to function as the subject of a projection of the new word.

One thing to point out is that I assume that there are head-complement phrases with only a lexical (i.e. of type *word*) head daughter and an empty list of non-head daughters (cf. Chapter 2, Section 2.3, fn. 8). A newly encountered word is therefore always attached as the head daughter of a head-complement phrase,[3] with either an empty or a non-empty NON-HD-DTRS value, depending on whether the word has an empty or a non-empty COMPS value, respectively. Any additional special selection feature-value pair in the lexical entry of the same word further triggers an additional phrase specification/construction step. Note that nonbranching phrases (with empty NON-HD-DTRS values) are eliminated in recent work in HPSG, e.g. Sag (to appear). Yet, assuming such structures in the present algorithm enables one to define the attachment procedures in a much more uniform manner, as will become clear in Section 6.3, where I introduce the details of the attachment procedures for Turkish.

6.2.1 An Example Parse

Keeping in mind the discussion so far, let us now consider the parse of a sentence such as (6.4), to gain a better insight into the attachment process. (Recall that the processing of this sentence was first introduced informally in Chapter 5, Section 5.2.)

(6.4) Scully loves cats.

At the beginning of the parse, the global structure, STRUCT, is constrained as an object of type *hd-ph-pr*, and it is the only phrase in the stack, PRED-PHRASES (cf. the first two steps in the algorithm, i.e. (6.2a,b)). The state of the parse at this stage is given in (6.5). (Concerning this parse state and the ones to come, the following points should be noted: i) the stack PRED-PHRASES is represented in the list notation with the top element on

[3]The only exception is markers; see Section 6.3.

the stack on the left end of the list; and ii) re-entrancies – identically numbered tags – can be viewed as structure-sharing, in the usual way, within a single feature structure, but otherwise should be viewed as pointers to the same linguistic object from an implementational point of view.)

(6.5)

STRUCT	PRED-PHRASES
[1] [*hd-ph-pr*]	< [1] >

On encountering *'Scully'* with no special selection feature-value pair selecting for a sister, the parser is simply to attach it as the head daughter of a head-complement phrase (with an empty NON-HD-DTRS value). As the active phrase at this stage is STRUCT, *'Scully'* is attached as the head daughter of that phrase, which is further constrained as of type *hd-comp-ph*. Since that phrase is now fully specified, it is also popped off the stack, which is then left empty. The state of the parse at that stage is illustrated in (6.6).

(6.6)

STRUCT	PRED-PHRASES
[*hd-comp-ph*; PHON ⟨*Scully*⟩; SYNSEM NP; HD-DTR *word*]	<>

When the next word *'loves'* is encountered, the parser fails to attach it to STRUCT, since the stack is now empty (hence no attachment site is available). Here, the non-determinism of the attachment process mentioned at step (6.2cii) of the algorithm comes into the picture. Note that the step outlined above is actually a phrase specification step. Let us now assume that while processing *'Scully'*, the parser non-deterministically decides to take the corresponding phrase construction step in the following way: it constructs a new phrase of type *hd-comp-ph* (headed by *'Scully'*), and attaches that new phrase as a non-head daughter of the active phrase, STRUCT, giving rise to the parse state in (6.7).[4]

(6.7)

STRUCT	PRED-PHRASES
[1] [*hd-ph-pr*; NON-HD-DTRS ⟨ [*hd-comp-ph*; PHON ⟨*Scully*⟩; SYNSEM NP; HD-DTR *word*] ⟩; HD-DTR *sign*]	< [1] >

[4]It should be noted that these phrase specification and construction steps roughly correspond to the 'state-application' and 'state-prediction' rules, respectively, of Milward (1995), who provides an incremental interpretation of Categorial Grammar.

Following up with the parse state in (6.7), (6.8) illustrates the state of the parse once the next word *'loves'* has been processed. As seen in (6.8), while processing *'loves'*, the parser constructs a new head-complement phrase headed by that word, and since *'loves'* subcategorizes for an NP[*nom*] subject that can be realized by the non-head daughter already attached to the active phrase, the new phrase is attached as the head daughter of the active phrase, STRUCT, which is further constrained as *hd-subj-ph* and popped off the stack (since it is now fully specified). In addition, *'loves'* also subcategorizes for an NP[*acc*] complement (via its non-empty COMPS value) that hasn't yet been encountered (according to the linear precedence constraints governing English). That complement has nevertheless been attached as a non-head daughter of the newly constructed head-complement phrase headed by *'loves'* (note the structure-sharing, tag [4], between the element in the COMPS list of *'loves'* and the SYNSEM value of that non-head daughter), as well as being pushed onto the stack of predicted phrases (tag [5]), hence becoming the active phrase.

(6.8)

```
STRUCT
[ hd-subj-ph
  SYNSEM        [ HEAD   [6]
                  SUBJ   ⟨⟩
                  COMPS  ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON    ⟨Scully⟩
                    SYNSEM  [3] NP[nom]
                    HD-DTR  word ] ⟩
  HD-DTR        [ hd-comp-ph
                  SYNSEM       [ HEAD   [6]
                                 SUBJ   ⟨[3]⟩
                                 COMPS  ⟨⟩ ]
                  NON-HD-DTRS  ⟨ [5] [ hd-ph-pr
                                       SYNSEM  [4] NP[acc] ] ⟩
                  HD-DTR       [ word
                                 PHON   ⟨loves⟩
                                 HEAD   [6] verb[fin]
                                 SUBJ   < [3] >
                                 COMPS  < [4] > ] ] ]

PRED-PHRASES
< [5] >
```

Finally, when the last word *'cats'* is encountered, it is simply attached as the head daughter of the active phrase, [5], which is further constrained as of type *hd-comp-ph* and popped off the stack. The final state of the parse is given in (6.9). Notice that the stack is now empty, satisfying the Grammaticality Principle stated in (6.3).

(6.9) STRUCT

```
[ hd-subj-ph
  SYNSEM        [ HEAD   [6]
                  SUBJ   ⟨⟩
                  COMPS  ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON    ⟨Scully⟩
                    SYNSEM  [3] NP[nom]
                    HD-DTR  word ] ⟩
  HD-DTR        [ hd-comp-ph
                  SYNSEM        [ HEAD   [6]
                                  SUBJ   ⟨[3]⟩
                                  COMPS  ⟨⟩ ]
                  NON-HD-DTRS   ⟨ [ hd-comp-ph
                                    PHON    ⟨cats⟩
                                    SYNSEM  [4] NP[acc]
                                    HD-DTR  word ] ⟩
                  HD-DTR        [ word
                                  PHON   ⟨loves⟩
                                  HEAD   [6] verb[fin]
                                  SUBJ   < [3] >
                                  COMPS  < [4] > ] ] ]
```

PRED-PHRASES

<>

To sum up, parsing a string always starts in a state expecting an object of type *hd-ph-pr* (the global structure STRUCT), and proceeds by attaching (the lexical entries for) the words in the input string to STRUCT as soon as each word is encountered. The attachment of a single word involves the specification/construction of at least one, and possibly more, phrase objects as sub-structures of STRUCT, whose types are determined by the particular selection feature-value pair(s) in the lexical entry of the word in question. The specification/construction of a phrase object of a particular type may result in the prediction of a (number of) yet-unencountered daughter(s), which is (are) then pushed onto the stack. A successful parse is achieved if all the predicted phrases (including STRUCT, predicted at the beginning of the parse) are fully specified and popped off the stack by the end of the parse (that is, once all the words in the input are processed).

6.2.2 Success of Phrase Specification/Construction Steps

Although a phrase specification step might fail in certain cases due to particular constraints imposed on the predicted phrases by the grammar, a phrase construction step always succeeds so long as there is an attachment site available for the newly constructed phrase (i.e. the stack is non-empty). Consider, for example, the parse of (6.10), with a 'that'-less complement clause.

(6.10) Mulder thinks Scully loves cats.

(6.11) illustrates the parse state after the word *'thinks'* is processed. On encountering the next word *'Scully'*, the specification of the active phrase, $\boxed{4}$, as an NP (headed by *'Scully'*) fails, since the SYNSEM value of $\boxed{4}$ is constrained as S[*fin*]. However, the corresponding construction step (which involves the construction of a new head-complement phrase headed by *'Scully'*, which is further attached as a non-head daughter of the active phrase, $\boxed{4}$) succeeds, and the rest of the parse proceeds in a way similar to the parse of (6.4) above.

(6.11) <u>STRUCT</u>

```
[ hd-subj-ph
  SYNSEM        [ HEAD   [3]
                  SUBJ   ⟨⟩
                  COMPS  ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON     ⟨Mulder⟩
                    SYNSEM   [1] NP[nom]
                    HD-DTR   word ] ⟩
  HD-DTR        [ hd-comp-ph
                  SYNSEM       [ HEAD   [3]
                                 SUBJ   ⟨[1]⟩
                                 COMPS  ⟨⟩ ]
                  NON-HD-DTRS  ⟨ [4] [ hd-ph-pr
                                       SYNSEM  [2] S[fin] ] ⟩
                  HD-DTR       [ word
                                 PHON   ⟨thinks⟩
                                 HEAD   [3] verb[fin]
                                 SUBJ   < [1] >
                                 COMPS  < [2] > ] ] ]
```

<u>PRED-PHRASES</u>

< $\boxed{4}$ >

6.2.3 More on the Non-determinism of the Attachment Process

It should be noted that in the case of certain languages (such as English), a phrase construction step may itself be non-deterministic in the following way:

i) It may involve the attachment of a newly constructed phrase as a 'non-head' daughter of an underspecified phrase, as in the attachment of the NP *'Scully'* as the non-head daughter of the global structure in the parse of (6.4) (cf. the parse state in (6.7)).

ii) It may involve the attachment of a newly constructed phrase as the 'head' daughter of an underspecified phrase, as in the attachment of the N′ *'policeman'* to the structure in the parse of (6.12), as shown in (6.13). (See also Chapter 5, Section 5.5, pp. 144–147 where this parse is first introduced.)

(6.12) $[_{NP}\ [_{DetP}\ \text{the}\ ]\ [_{N'}\ [_{N'}\ [_{N'}\ \text{policeman}\ ]\ [_{PP}\ \text{with glasses}\ ]]\ [_{PP}\ \text{in uniform}\ ]]]$

(6.13)

STRUCT

```
[ hd-spr-ph
  SYNSEM        [ HEAD  [3] noun
                  SPR   ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON    ⟨the⟩
                    SYNSEM  [4] [ SPEC [1] N′ ]
                    HD-DTR  word ] ⟩
  HD-DTR        [2] [ hd-ph-pr
                      SYNSEM       [1] [ HEAD [3]
                                         SPR  ⟨[4]⟩ ]
                      NON-HD-DTRS  list(sign)
                      HD-DTR       [ hd-comp-ph
                                     PHON    ⟨policeman⟩
                                     SYNSEM  N′
                                     HD-DTR  word ] ] ]
```

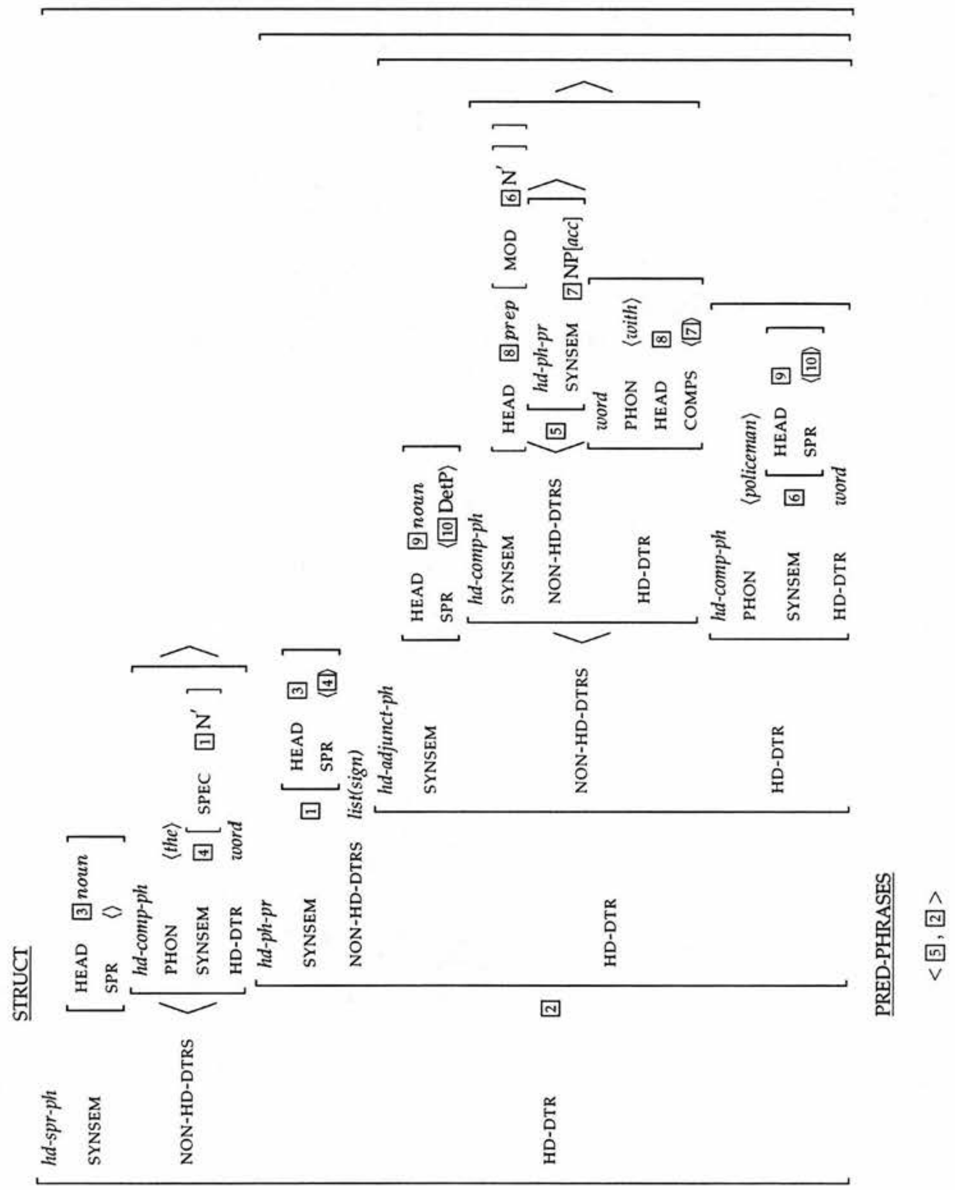
PRED-PHRASES

< [2] >

This kind of non-determinism of a construction step is due to the fact that in languages like English, one may have certain phrase types such as head-subject phrases, where the daughter being selected for is a 'non-head' daughter preceding the selecting (head) daughter (which actually bears the special feature signifying the phrase type), as well as certain other phrase types such as head-adjunct phrases with post-modifiers, where the daughter being selected for is a 'head' daughter preceding the selecting (non-head) daughter. However, one can get rid of the above-mentioned non-determinism of a construction step in a given language, if for any phrase type with the daughter being selected for preceding the selecting daughter, the (phrase-initial) selected daughter is always either a non-head daughter or the head daughter. In a head-final language, for example, since in any phrase type the head daughter follows its sister(s), for a construction step one only needs the alternative (i) mentioned above, that is, attaching the newly constructed phrase as a 'non-head' daughter of an underspecified phrase. At first sight, it may seem as if head-background phrases would constitute a problem for such a simplification in the case of Turkish, since the head daughter in such a phrase precedes its sister. Note, however, that in a head-background phrase the (phrase-initial) head daughter is actually the selecting daughter of the phrase (which selects the non-head daughter via a *local* object in its non-empty INHER|SLASH set). One can therefore readily adopt the above simplification for head-final languages, in parsing Turkish (and in fact any typically head-final language that allows post-predicate scrambling).

Another point is that, as mentioned before, the non-determinism introduced by the specification and construction steps for head-complement phrases exists for other types of phrase objects as well. As an example, consider again the parse of (6.12), this time the processing of the first preposition *'with'*. A new head-complement phrase is constructed, as usual, which is then attached as the non-head daughter of a newly constructed head-adjunct phrase (because of the *synsem*-valued MOD feature of *'with'*), as shown in (6.14). Notice that the head-complement phrase headed by *'policeman'* in (6.13) is now re-attached as the head daughter of the new head-adjunct phrase, which is further attached as the head daughter of the *hd-ph-pr* phrase, overriding the previous value. (By way of comparison, see (5.17) in Chapter 5, Section 5.5, page 144, for the corresponding 'specification' step in the parse of (5.6b), *'the policeman with glasses'*.)

(6.14)



The construction step in (6.14) then lets the parser attach the PP headed by the second preposition *'in'* as the non-head daughter of 4, and further specify that phrase as *hd-adjunct-ph,* as shown in (6.15).

(6.15)

STRUCT	PRED-PHRASES
(feature structure below)	< [12] >

```
[ hd-spr-ph
  SYNSEM        [ HEAD  [3] noun
                  SPR   ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON    ⟨the⟩
                    SYNSEM  [4] [ SPEC [1] N' ]
                    HD-DTR  word ] ⟩
  HD-DTR        [ hd-adjunct-ph
                  SYNSEM       [1] [ HEAD [3]
                                     SPR  ⟨[4]⟩ ]
                  NON-HD-DTRS  ⟨ [ hd-comp-ph
                                   SYNSEM       [ HEAD [14] prep [ MOD [11] N' ] ]
                                   NON-HD-DTRS  ⟨ [12] [ hd-ph-pr
                                                         SYNSEM [13] NP[acc] ] ⟩
                                   HD-DTR       [ word
                                                  PHON   ⟨in⟩
                                                  HEAD   [14]
                                                  COMPS  ⟨[13]⟩ ] ] ⟩
                  HD-DTR       [ hd-adjunct-ph
                                 SYNSEM       [11] [ HEAD [3]
                                                     SPR  ⟨[10]⟩ ]
                                 NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                  SYNSEM       [ HEAD [8] prep [ MOD [6] N' ] ]
                                                  NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                                   PHON    ⟨glasses⟩
                                                                   SYNSEM  [7] NP[acc]
                                                                   HD-DTR  word ] ⟩
                                                  HD-DTR       [ word
                                                                 PHON   ⟨with⟩
                                                                 HEAD   [8]
                                                                 COMPS  ⟨[7]⟩ ] ] ⟩
                                 HD-DTR       [ hd-comp-ph
                                                PHON    ⟨policeman⟩
                                                SYNSEM  [6] [ HEAD [3]
                                                              SPR  ⟨[10]⟩ ]
                                                HD-DTR  word ] ] ] ]
```

6.2.4 The Notion of Incrementality

The notion of incrementality I use here is that each word of a given input string is attached to a global structure (assigned to the input) from left to right as soon as it is encountered. Disjunctive representations are not allowed within the structure at any point in the course of the parse. Schabes (1990) introduces the notion of 'Valid Prefix Property', which he suggests a procedure should obey in order to be regarded as incremental:

(6.16) **The Valid Prefix Property**
If the input tokens $a_1 \ldots a_k$ have been read then it is guaranteed that there is a string of tokens $b_1 \ldots b_m$ (b_i may not be part of the input) such that the string $a_1 \ldots a_k b_1 \ldots b_m$ is a valid string of the language.

In other words, the left-to-right processing of an input string must be terminated as soon as it is certain that there is no valid continuation of the part of the input already processed. Note that the present algorithm does not satisfy this criterion, since in certain cases a wrong analysis may not be rejected at the earliest possible point, due to underspecification in the structure. For instance, while processing (6.17) the analysis with *'Kim'* being a male would only be rejected when the preposition *'from'* was processed, rather than the previous word *'husband'* (which would be the earliest possible point to reject that analysis).

(6.17) Kim's husband from America

6.3 Attachment Procedures for Turkish

This section presents the attachment routines for Turkish in a pseudo-code notation. The main body of the 'attach' routine is given in (6.18). (Note that 'attach' is called by the main body of the algorithm at step (6.2cii); cf. page 167.) For any feature name X and *sign* object Y, the function 'get-X(Y)' returns the value of X in Y, and '=' denotes 'unification'. In addition, the individual phrase routines (e.g. 'head-marker-ph-routine', 'head-comp-ph-routine') in this pseudo-code notation are assumed to be called by unification, an input/output passing mechanism (as is the case with predicates in Prolog and LIFE).

Each one of these phrase routines (except for check-CASE at step 2.5) involves i) a sub-routine for phrase specification, which constrains the top element on the stack as an instance of the phrase type in question, popping it off the stack; and ii) a sub-routine for phrase construction, which constructs a new phrase object that is an instance of the phrase type in question, and attaches that new phrase as a non-head daughter of the top element on the stack (see sections to come for details). The parser is assumed to non-deterministically call one or the other sub-routine for any phrase type call. Note, once again, that the processing of a given word involves the specification/construction of at least one, and possibly more, phrase objects, depending on the existence of the corresponding special selection feature-value pair in the lexical entry for that word.

(6.18) attach(NewWord)

1. if get-HEAD(NewWord) = *marker* then
   head-marker-ph-routine(NewWord,PredHeadMark)
2. else
   2.1. head-comp-ph-routine(NewWord,PredCompDtrs)

   2.2. if get-SUBJ(NewWord) = *nelist-synsem* then
        head-subj-ph-routine

   2.3. if get-MOD(NewWord) = *synsem* then
        head-adjunct-ph-routine(PredHeadAdj)

   2.4. else if get-SPEC(NewWord) = *synsem* then
        head-spr-ph-routine(PredHeadSpr)

   2.5. if has-CASE(NewWord) then
        check-CASE

   2.6. if get-INHER-SLASH(NewWord) = *ne-set-local* then
        check-fronting(get-INHER-SLASH(NewWord))

   2.7. head-backg-ph-routine
3. Push any predicted daughters onto the stack in
   the reverse order as they are predicted.

As mentioned in Section 6.2, a newly encountered word, with the exception of markers, always triggers the specification/construction of a head-complement phrase, with either an empty or a non-empty NON-HD-DTRS value, depending on whether the word has an empty or a non-empty COMPS value, respectively. Any additional special selection feature-value pair in the lexical entry of the same (non-marker) word further triggers an additional phrase specification/construction step. On the other hand, if the new word is a marker (i.e. has a HEAD value of type *marker*), then it is directly attached as the non-head daughter of a head-marker phrase, since such a daughter is constrained to be of type *word* in the grammar. Thus, 'attach' first checks whether the HEAD value of the new word is of type *marker*, and if so, calls the head-marker phrase routine (which specifies/constructs a head-marker phrase, attaching the new word as its non-head daughter; see Section 6.3.1).

If the new word is not a marker, then 'attach' proceeds to call the head-complement phrase routine, and the remaining phrase routines, depending on the existence of the corresponding special selection feature-value pair in the lexical entry for the new word. It is important to note that the order in which these routines are called is significant. The routine for head-complement phrases is called first, since all other phrase types require the daughter to which a projection of the new word might be attached, to have an empty COMPS value in the grammar (cf. the Empty COMPS Constraint in the grammar). For example, the head daughter of a head-subject phrase, or the non-head daughter of a head-adjunct or a head-specifier phrase, to which a projection of the new word might end up having been attached, are all constrained to have an empty COMPS value. In other words, those daughters are all required to have combined with all their complements, if any. Similar reasoning applies to the relative ordering of the calls for the remaining routines.

In Chapter 3 (Section 3.4.1), I propose a flat structure for Turkish sentences, where the verbal head selects its subject via the COMPS feature just like the other arguments. Thus, Turkish sentences are considered instances of *hd-comp-ph*. In addition, again in Chapter 3 (Section 3.3), I propose a subject analysis for possessors in Turkish, meaning that possessive NPs are assumed to be instances of *hd-subj-ph*. Hence, the head-subject phrase routine (cf. step 2.2) is included in (6.18) only to cover the possessive NPs.

Recall from Chapter 2 that in HPSG, specifiers and heads reciprocally select for each other via the SPEC and SPR features, respectively. Consequently, in terms of incremental processing, in the case of a language with specifiers preceding their heads, a head-specifier phrase would always be signified by the *synsem*-valued SPEC feature of the specifier (non-head) daughter. On the other hand, in the case of a language with specifiers following their heads, such a phrase would always be signified by the non-empty SPR value of the head daughter. Since Turkish belongs to the former class, the head-specifier phrase routine is called in the case of a word with a *synsem*-valued SPEC feature (cf. step 2.4).

Note that the new word can only have either a *substantive* type HEAD value, with an appropriate MOD feature, or a *functional* type HEAD value, with an appropriate SPEC feature, as specified by the 'if-else' statement at steps 2.3 and 2.4.

In Chapter 5 (Section 5.6), I discuss how an incremental parser can exploit the CASE values of constituents while processing complex Turkish sentences with embedded complement clauses. This strategy is incorporated in the 'attach' routine at step 2.5, which checks if the new word has an appropriate CASE feature, and if so, calls the routine 'check-CASE' (see Section 6.5). The routine 'has-CASE' returns *true* if the new word has a HEAD value of type *noun*, or one of type *verb* with the VFORM value constrained as *nominalization* or *participle*; it returns *false* otherwise. (Recall from Chapter 3, Section 3.1, that CASE is defined as an appropriate feature for only types *noun*, *nominalization*, and *participle* in the grammar; note that, unlike German, for example, in Turkish specifiers and adjuncts do not inflect according to the CASE value of the head noun.)

Again, in Chapter 5 (Section 5.7), I propose an incremental recovery of filler-gap relations in long-distance topicalization in Turkish, which is triggered by a lexical head with a non-empty INHER|SLASH value. Step 2.6 in (6.18) is responsible for the incorporation of that strategy in the algorithm, by a call for the 'check-fronting' routine (see Section 6.6).

For reasons that I discuss in Section 6.3.4, the present approach doesn't deal with backgrounding in Turkish in an incremental fashion, as soon as a lexical head with a non-empty INHER|SLASH value is encountered, but rather it specifies/constructs such a phrase only after encountering the filler daughter.

Recall from the discussion in Section 6.2 that every phrase type is signified by a certain daughter selecting for the other daughter(s) in the phrase via a certain special selection feature value. In the case of a phrase type where the selecting daughter follows the daughter(s) selected for, at the time of processing the selecting daughter, that (those) daughter(s) must have already been encountered, hence no further prediction is necessary. On the other hand, if the selecting daughter in a particular type of phrase precedes the daughter(s) being selected, then that (those) daughter(s) need to be predicted by being pushed onto the stack. Since in Turkish markers, adjuncts and specifiers (i.e. the selecting daughters in the respective phrase types) precede their heads, the corresponding phrase specification/construction routines are assumed to return a predicted daughter (PredHeadMark, PredHeadAdj, and PredHeadSpr at steps 1, 2.3 and 2.4, respectively). Although head-complement phrases are typically head-final in Turkish, we have seen in Chapter 3 (Section 3.4.2) that *marked* complement clauses always immediately follow their heads. Consequently, the list of predicted daughters at step 2.1, PredCompDtrs, is either empty or contains only one element constrained to have an S[*marked*] SYNSEM value (see Section 6.3.2). In a head-subject phrase (i.e. possessive NP) the head daughter, which selects the non-head daughter via the SUBJ feature, follows that daughter, hence no predicted daughter is returned by the head-subject phrase routine (cf. step 2.2).

Finally, at step 3 any daughters predicted at steps 1-2 are pushed onto the stack in the reverse order as they are predicted, with the assumption that all phrase types specify continuous structures.

Let us now consider the individual phrase specification/construction routines called by 'attach' in detail. As mentioned before, for each phrase type, there exist a sub-routine for phrase specification, and one for phrase construction.

The routines for head-marker and head-complement phrases differ from the rest in one important respect. That is, the daughter that signifies such a phrase (i.e. the non-head daughter in a head-marker phrase and the head daughter in a head-complement phrase) is constrained to be of type *word*, rather than *phrase*, in the grammar. Thus, a newly encountered word signifying such a phrase is always directly attached as the corresponding daughter of the phrase, whereas in the case of all other phrase types, it is always a phrase containing the new word that is attached as a daughter.

Below, I first present the specification/construction routines for head-marker phrases and head-complement phrases, and then the remaining phrase types. (Note that in the presentation of the algorithm here the stack PRED-PHRASES is assumed to be a global variable that is accessible by any of the phrase routines without being passed to the routine by the calling mechanism.)

6.3.1 Head-Marker Phrases

(6.19) and (6.20) deal with specification and construction of head-marker phrases, respectively. 'Pop(X)' is a function which takes a stack X as input, returns its top element and, as a side-effect, updates the stack by removing that top element. 'Top(X)', on the other hand, is a function that only returns the top element of the input stack X (without any side-effect on X). In both cases, the function call 'pop/top(X)' fails, if the stack is empty. '=', as before, denotes the unification operation, and '←' denotes the assignment operation.

Notice that in both sub-routines, the new word, which signifies a head-marker phrase (with its *marker*-type HEAD value), is attached as the non-head daughter of the head-marker phrase, and the head daughter of that phrase (which hasn't yet been encountered) is returned back as being predicted. In the case of the specification routine, (6.19), the active phrase, which is always underspecified as of type *hd-ph-pr*, is further constrained (specified) as of type *hd-mark-ph*, and is also popped off the stack of predicted phrases. In the construction routine, (6.20), however a new phrase is constructed, which is constrained as of type *hd-mark-ph*, and is attached as a non-head daughter of the active phrase (in addition to the non-head daughters already attached to that phrase), overriding the previous NON-HD-DTRS value of the phrase.

(6.19) <u>head-marker-ph-routine(NewWord,PredDtr)</u>

1. ActPhrase = pop(PRED-PHRASES)

2. $\text{ActPhrase} = \begin{bmatrix} \textit{hd-mark-ph} & \\ \text{NON-HD-DTRS} & \langle \text{NewWord} \rangle \\ \text{HD-DTR} & \text{PredDtr} \end{bmatrix}$

(6.20) <u>head-marker-ph-routine(NewWord,PredDtr)</u>

1. ActPhrase = top(PRED-PHRASES)

2. $\text{FirstNonHeadDtr} \leftarrow \begin{bmatrix} \textit{hd-mark-ph} & \\ \text{NON-HD-DTRS} & \langle \text{NewWord} \rangle \\ \text{HD-DTR} & \text{PredDtr} \end{bmatrix}$

3. get-NON-HD-DTRS(ActPhrase) ←

$$\begin{bmatrix} \textit{nelist-ph} & \\ \text{FIRST} & \text{FirstNonHeadDtr} \\ \text{REST} & \text{get-NON-HD-DTRS(ActPhrase)} \end{bmatrix}$$

6.3.2 Head-Complement Phrases

The two sub-routines for head-complement phrases that are responsible for specification and construction steps are given in (6.21) and (6.22), respectively. 'Extract-synsems-of' and 'construct-signs-of' are two functions that are in a way "inverse" of each other. The former takes a list of *sign* objects as input, and returns a list of *synsem* objects whose elements are structure-shared with the SYNSEM values of the corresponding elements in the input list. The latter, on the other hand, takes a list of *synsem* objects as input, and returns a list of *sign* objects whose elements structure-share their SYNSEM values with the corresponding elements in the input list.[5] '•' denotes list concatenation.

The specification routine for head-complement phrases, (6.21), sets out by popping the active phrase off the stack. It then compares the SYNSEM values of any non-head daughters already encountered and attached to the active phrase, with the *synsem* objects in the new words COMPS list, using the 'comps-difference' function, to predict any complements of the new word that are still to come (cf. step 4). Note that 'comps-difference' performs a comparison of these two lists of *synsem* objects, taking 'unification' as the basis for the "sameness" of any two *synsem* objects, and without considering any restrictions on the respective order of the objects in the two input lists, since complements (in the preverbal position) can scramble rather freely in Turkish.[6]

We have seen in Chapter 3 that all complements, with the exception of *marked* complement clauses, are restricted to precede the head in the typical order (see the corresponding LP constraints in (3.65) in that chapter, page 71). This restriction is handled by 'check-head-final-LP', which basically requires the list of the SYNSEM values of the predicted complements either to be empty or to contain only one element which is an S[*marked*]. In addition, we have also seen in Chapter 3 that non-case-marked object complements (i.e. NP[*nom*] objects, and VP[*inf*] and S[*fin,unmarked*] complement clauses)

[5]Note that the term 'inverse' above is used in a rather loose sense. For a list L of *synsem*-objects, the consecutive applications of the two functions, i.e. extract-synsems-of(construct-signs-of(L)), indeed returns the original input list L. However, for a list L of *sign*-objects, construct-signs-of(extract-synsems-of(L)) may not return the original list L (although it is guaranteed to return a list L' that is 'unifiable' with L).

[6]Note that in the present parsing approach for Turkish, I abandon the constraint on the type *hd-comp-ph* in the grammar presented in Chapter 2 (cf. Table 2.1) which requires the elements in the NON-HD-DTRS list of any instance of *hd-comp-ph* to structure-share their SYNSEM values with the corresponding elements in the COMPS list of the head daughter of the phrase. That constraint is in a way *procedurally* imposed on head-complement phrases during processing by steps 4, 7 and 8 in (6.21) (and the corresponding steps 4, 10 and 11 in the head-complement phrase construction routine given in (6.22) below).

always *immediately* precede their head in Turkish (again see the corresponding LP constraint in (3.66) in that chapter, page 72). This restriction is handled by 'check-non-case-obj-LP', whose details are discussed below. Then at step 8 any predicted non-head daughter is added to the ones already encountered, overriding the NON-HD-DTRS value of the active phrase. Finally, at step 9 the new word is attached as the head daughter of the active phrase, which is further constrained as of type *hd-comp-ph*. Notice also that any predicted daughter is returned back to the main 'attach' routine, to be pushed onto the stack.

(6.21) head-comp-ph-routine(NewWord,PredDtrs)

1. ActPhrase = pop(PRED-PHRASES)
2. OldNonHeadDtrs = get-NON-HD-DTRS(ActPhrase)
3. OldNonHeadDtrSynsems = extract-synsems-of(OldNonHeadDtrs)
4. PredDtrSynsems = comps-difference(get-COMPS(NewWord), OldNonHeadDtrSynsems)
5. check-head-final-LP(PredDtrSynsems)
6. check-non-case-obj-LP(OldNonHeadDtrSynsems, get-ARG-ST(NewWord))
7. PredDtrs = construct-signs-of(PredDtrSynsems)
8. get-NON-HD-DTRS(ActPhrase) $\leftarrow$ PredDtrs $\bullet$ OldNonHeadDtrs
9. ActPhrase = $\begin{bmatrix} \textit{hd-comp-ph} & \\ \text{HD-DTR} & \text{NewWord} \end{bmatrix}$

(6.22), which is responsible for the construction step for head-complement phrases, is similar to (6.21), except that it constructs a new head-complement phrase (cf. step 11), rather than constraining the active phrase as *hd-comp-ph*. The new phrase is then attached as a non-head daughter of the active phrase, again overriding the previous NON-HD-DTRS value of that phrase. Note that some of the most recently attached non-head daughters in the active phrase, TempNonHeadDtrs, are re-attached as non-head daughters of the new head-complement phrase, whereas the others, RestNonHeadDtrs, are left as non-head daughters of the active phrase. Clearly, either or both of these lists may be empty.

(6.22) head-comp-ph-routine(NewWord,PredDtrs)

1. ActPhrase = top(PRED-PHRASES)

2. OldNonHeadDtrs = get-NON-HD-DTRS(ActPhrase)

3. append(TempNonHeadDtrSynsems, RestNonHeadDtrSynsems,
   extract-synsems-of(OldNonHeadDtrs))

4. PredDtrSynsems = comps-difference(get-COMPS(NewWord),
   TempNonHeadDtrSynsems)

5. check-head-final-LP(PredDtrSynsems)

6. check-non-case-obj-LP(TempNonHeadDtrSynsems, get-ARG-ST(NewWord))

7. TempNonHeadDtrs = construct-signs-of(TempNonHeadDtrSynsems)

8. RestNonHeadDtrs = construct-signs-of(RestNonHeadDtrSynsems)

9. append(TempNonHeadDtrs, RestNonHeadDtrs,
   OldNonHeadDtrs)

10. PredDtrs = construct-signs-of(PredDtrSynsems)

11. FirstNonHeadDtr ← $\begin{bmatrix} \textit{hd-comp-ph} & \\ \text{NON-HD-DTRS} & \text{PredDtrs} \bullet \text{TempNonHeadDtrs} \\ \text{HD-DTR} & \text{NewWord} \end{bmatrix}$

12. get-NON-HD-DTRS(ActPhrase) ← $\begin{bmatrix} \textit{nelist-ph} & \\ \text{FIRST} & \text{FirstNonHeadDtr} \\ \text{REST} & \text{RestNonHeadDtrs} \end{bmatrix}$

Let us now consider the details of 'check-non-case-obj-LP', (6.23), mentioned above in (6.21) and (6.22). (6.23) checks if the second element in the ARG-ST list of the new word is a non-case-marked complement that should immediately precede the head daughter (i.e. if it has a HEAD value of type *noun*, and if so, if its CASE value is *nominative*, or if it has a HEAD value of type *verb*, and if so, if its VFORM value is *inf* or *fin* – in which case the MARKING value should be *unmarked*), and if it is, then constrains it to be structure-shared with the SYNSEM value of the most recently attached non-head daughter in the head-complement phrase to be headed by the new word (that is, the first element in that list).

(6.23) check-non-case-obj-LP(NonHeadDtrSynsems, ArgsNewWord)

1. SecondArg = second(ArgsNewWord)

2. if (head(SecondArg) = *noun* and
   case(SecondArg) = *nominative*) or
   (head(SecondArg) = *verb* and
   (vform(SecondArg) = *inf* or
   vform(SecondArg) = *fin* and
   marking(SecondArg) = *unmarked*)) then
   2.1. first(NonHeadDtrSynsems) = SecondArg

It might prove helpful to make a comparison with the application rule of Categorial Grammar, to help the reader grasp the main idea behind these routines. Those complements of a lexical head which should precede it, i.e. the categories that are required by a function on the left in CG terms, should already have been encountered when the lexical head is processed. Although the complements that should follow the head, i.e. the categories that are required on the right, have not yet been encountered at that point, they are still included among the complement daughters of the phrase headed by that lexical head, as well as being pushed onto the stack, hence being predicted.

6.3.3 Head-Adjunct/Specifier/Subject Phrases

Let us now discuss the routines for head-adjunct, head-specifier and head-subject phrases. Here I make a distinction between the kinds of phrases signified by a head selection feature of the non-head daughter (i.e. head-adjunct/head-specifier phrases signified by the *synsem*-valued MOD/SPEC feature of the non-head daughter), and those signified by a non-head selection feature of the head daughter (i.e. head-subject phrases signified by the non-empty SUBJ value of the head daughter).

As mentioned above, these routines differ from the previous ones in that although it is a special selection feature-value pair of the new word that triggers the specification/construction of a phrase by any of these routines, it is always a phrase containing the new word that is attached as one of the daughters of that newly specified/constructed phrase (rather than the new word itself). Recall from Section 6.3 that while processing a single word the parser may call more than one phrase routine depending on the existence of the corresponding special selection feature-value pair in that word's lexical entry. Every routine works on the output of the one called just before, leading the parser to pass through a number of *intermediary* states during the processing of a single word. Since a phrase specification routine always pops the active phrase off the stack, during the processing of a single word the active phrase may change several times (that is, whenever a phrase specification routine succeeds).[7] Consequently, in the discussion below, whenever a non-head daughter of the active phrase is said to be *re-attached*, (one should bear in mind that) that daughter may first have been attached there either i) at a previous stage during the processing of the same word; or ii) during the processing of a previously encountered word.

Head Selection Routines

Let us first discuss the case with the head-selection features MOD and SPEC. The subroutines dealing with specification and construction steps for these phrase types are given in (6.24) and (6.25), respectively, with X standing for either specifier or adjunct. (6.24) simply pops the active phrase off the stack and constrains it to be of type *hd-X-ph*. Note that the head daughter of the active phrase, which hasn't yet been encountered, is returned back as a predicted daughter.

---

[7]Note, however, that any as yet unencountered daughter predicted by any of the phrase specification/construction routines during the processing of a particular word is always pushed onto the stack only at the final stage of the processing of that word (cf. step 3 in the 'attach' routine, (6.18)). Hence, any prediction of a daughter yet to come at one stage of the processing of a particular word does not have any affect on the subsequent stages of the processing of the same word.

(6.24) <u>head-X-ph-routine(PredDtr)</u>

1. ActPhrase = pop(PRED-PHRASES)
2. ActPhrase = $\begin{bmatrix} \textit{hd-X-ph} & \\ \textsc{hd-dtr} & \text{PredDtr} \end{bmatrix}$

(6.25), on the other hand, constructs a new phrase of type *hd-X-ph*, re-attaching the most recently attached non-head daughter in the active phrase as the non-head daughter of the new phrase, and further attaches that phrase as a non-head daughter of the active phrase. Note that the head daughter of the head-X phrase is again returned back as being predicted.

(6.25) <u>head-X-ph-routine(PredDtr)</u>

1. ActPhrase = top(PRED-PHRASES)
2. get-$\textsc{non-hd-dtr}$(ActPhrase) = $\begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{NonHeadDtr} \\ \textsc{rest} & \text{RestNonHeadDtrs} \end{bmatrix}$
3. FirstNonHeadDtr $\leftarrow$ $\begin{bmatrix} \textit{hd-X-ph} & \\ \textsc{non-hd-dtrs} & \langle \text{NonHeadDtr} \rangle \\ \textsc{hd-dtr} & \text{PredDtr} \end{bmatrix}$
4. get-$\textsc{non-hd-dtrs}$(ActPhrase) $\leftarrow$ $\begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FirstNonHeadDtr} \\ \textsc{rest} & \text{RestNonHeadDtrs} \end{bmatrix}$

Non-head Selection Routines

Turning now to the case with the non-head selection feature SUBJ, there again exist two sub-routines, dealing with specification and construction steps for head-subject phrases. The specification routine in (6.26) pops the active phrase off the stack, and removes the most recently attached non-head daughter from the list of non-head daughters, re-attaching it as the head daughter of the active phrase, which is also constrained as *hd-subj-ph*. Note once again that there is no predicted daughter in this case, since the non-head daughter, which is being selected for, must have already been encountered.

(6.26) <u>head-subj-ph-routine</u>

1. ActPhrase = pop(PRED-PHRASES)

2. get-NON-HD-DTRS(ActPhrase) = $\begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FirstDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix}$

3. get-NON-HD-DTRS(ActPhrase) ← RestDtrs

4. ActPhrase = $\begin{bmatrix} \textit{hd-subj-ph} & \\ \textsc{hd-dtr} & \text{FirstDtr} \end{bmatrix}$

The construction routine in (6.27) re-attaches the most recently attached non-head daughter in the active phrase and the one preceding it, as the head daughter and the non-head daughter of a new head-subject phrase, respectively. The new phrase is then attached as a non-head daughter of the active phrase (in addition to any previously attached ones).

(6.27) <u>head-subj-ph-routine</u>

1. ActPhrase = top(PRED-PHRASES)

2. get-NON-HD-DTRS(ActPhrase) = $\begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FirstDtr} \\ \textsc{rest} & \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{SecondDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix} \end{bmatrix}$

3. FirstNonHeadDtr ← $\begin{bmatrix} \textit{hd-subj-ph} & \\ \textsc{non-hd-dtrs} & \langle \text{SecondDtr} \rangle \\ \textsc{hd-dtr} & \text{FirstDtr} \end{bmatrix}$

$$\text{4. get-NON-HD-DTRS(ActPhrase)} \leftarrow \begin{bmatrix} \textit{nelist-ph} & \\ \text{FIRST} & \text{FirstNonHeadDtr} \\ \text{REST} & \text{RestDtrs} \end{bmatrix}$$

6.3.4 Head-Background Phrases

In Chapter 5 (Section 5.3), I discussed certain recursive structures whose processing raises problems for an incremental parsing approach for HPSG grammars. A similar case arises in the incremental processing of examples of backgrounding in Turkish. Consider, for example, (6.28a,b,c), where the extracted *genitive* NP subject of the inner *fact* clause appears in the post-verbal position of the inner and the outer *fact* clauses, and the matrix clause, respectively.

(6.28) a. Berfu'-ya $[_{S[fact]}$ $[_{S[fact]}$ $[_{S[fact]}$ $-_i$ uyu-duğ-u-nu] Güneş'-in$_i$]
Berfu-DAT sleep-FACT-3sPoss-ACC Güneş-GEN
ben-im gör-düğ-üm-ü] Yasemin söyle-di.
I-GEN see-FACT-1sPoss-ACC Yasemin tell-PAST

b. Berfu'-ya $[_{S[fact]}$ $[_{S[fact]}$ $[_{S[fact]}$ $-_i$ uyu-duğ-u-nu] ben-im
Berfu-DAT sleep-FACT-3sPoss-ACC I-GEN
gör-düğ-üm-ü] Güneş'-in$_i$] Yasemin söyle-di.
see-FACT-1sPoss-ACC Güneş-GEN Yasemin tell-PAST

c. $[_{S[fin]}$ Berfu'-ya $[_{S[fact]}$ $[_{S[fact]}$ $-_i$ uyu-duğ-u-nu] ben-im
Berfu-DAT sleep-FACT-3sPoss-ACC I-GEN
gör-düğ-üm-ü] Yasemin söyle-di] Güneş'-in$_i$
see-FACT-1sPoss-ACC Yasemin tell-PAST Güneş-GEN
'YASEMIN has told Berfu that I have seen that she, Güneş, was asleep.'

Since there may be an arbitrary number of embeddings in such cases, and since an extracted complement may appear in the post-verbal position of any of the surrounding clauses, the incremental specification/construction of a head-background phrase, as soon as encountering a head with a non-empty INHER|SLASH value, seems to be problematic again. The solution I adopt here is actually based on a departure from strict incrementality. In other words, the parser postpones the specification/construction of a head-background phrase in such cases until encountering the filler daughter itself. The specification and construction routines for head-background phrases are presented in (6.29) and (6.30), respectively.

(6.29) pops the active phrase off the stack, as usual. It then singles out the second but most recently attached non-head daughter in the active phrase, and re-attaches that phrase as the head daughter of the active phrase, and at the same time constrains the active phrase to be of type *hd-backg-ph*.

(6.29) head-backg-ph-routine

1. ActPhrase = pop(PRED-PHRASES)

2. $\text{get-}\textsc{non-hd-dtrs}(\text{ActPhrase}) = \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FillerDtr} \\ \textsc{rest} & \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{HeadDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix} \end{bmatrix}$

3. $\text{get-}\textsc{non-hd-dtrs}(\text{ActPhrase}) \leftarrow \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FillerDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix}$

4. $\text{ActPhrase} = \begin{bmatrix} \textit{hd-backg-ph} & \\ \textsc{hd-dtr} & \text{HeadDtr} \end{bmatrix}$

(6.30) is similar to (6.29) except that (instead of further specifying the active phrase) it constructs a new phrase of type *hd-backg-ph*, re-attaching the most recently attached non-head daughter in the active phrase as the non-head daughter of the new phrase, and the second but most recently attached daughter as the head daughter of the new phrase. The head-background phrase is then attached as a non-head daughter of the active phrase.

(6.30) head-backg-ph-routine

1. ActPhrase = top(PRED-PHRASES)

2. $\text{get-}\textsc{non-hd-dtrs}(\text{ActPhrase}) = \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FillerDtr} \\ \textsc{rest} & \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{HeadDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix} \end{bmatrix}$

$$3.\ \text{FirstNonHeadDtr} \leftarrow \begin{bmatrix} \textit{hd-backg-ph} & \\ \textsc{non-hd-dtrs} & \langle \text{FillerDtr} \rangle \\ \textsc{hd-dtr} & \text{HeadDtr} \end{bmatrix}$$

$$4.\ \text{get-}\textsc{non-hd-dtrs}\text{(ActPhrase)} \leftarrow \begin{bmatrix} \textit{nelist-ph} & \\ \textsc{first} & \text{FirstNonHeadDtr} \\ \textsc{rest} & \text{RestDtrs} \end{bmatrix}$$

6.4 Example Parses

In this section, I present two parses, which exemplify the use of some of the phrase specification/construction routines in Section 6.3.

As mentioned before, in all phrase types other than *hd-mark-ph* and *hd-comp-ph*, the selecting daughter is of type *phrase* rather than *word*. A word with a special selection feature-value pair signifying such a phrase type may actually be arbitrarily embedded within the selecting daughter of the phrase in question, in which case the respective selection feature value is passed up to the actual selecting daughter via principles such as the HFP and the VALP in the grammar. (Note once again that these principles are all formulated as constraints imposed on particular phrase types in the type hierarchy.) In (6.31), for example, the NP MOD value of the adjective *'yaşlı'* is passed up to the head-adjunct phrase *'çok çok yaşlı'* headed by that adjective, which is the actual selecting (non-head) daughter of the head-adjunct phrase (6.31), via the HFP (as part of the HEAD value).

(6.31) [$_{NP}$ [$_{AdjP}$ çok [$_{AdjP}$ çok yaşlı]] [$_{NP}$ bir adam]]
very very old a man
'a very very old man'

(6.32) illustrates the parse of (6.31) step by step, where every pair of consecutive steps represents the states of the parse (i.e. the values of the global structure, STRUCT, and the stack of predicted phrases, PRED-PHRASES) right before and after the processing of the single word whose lexical entry is given in between, together with the particular phrase specification/construction routine(s) called during the processing of that word. Only the sequence of algorithm steps that non-deterministically produce the correct analysis is provided. Note that every object of a particular type must satisfy all the constraints imposed on that type in the grammar, and that only the most essential constraints are explicitly shown here in order to improve the readibility. Recall also the following points from Section 6.2.1, page 169: i) the stack PRED-PHRASES is represented in the list notation with the top element on the stack on the left end of the list; and ii) re-entrancies – identically numbered tags – can be viewed as structure-sharing, in the usual way, within a single feature structure, but otherwise should be viewed as pointers to the same linguistic object from an implementational point of view. Below I explain each step in (6.32).

1. At the beginning of the parse, STRUCT is constrained as an object of type *hd-ph-pr*, and it is the only element on the stack.

2. The processing of the first word *'çok'* involves calls for two phrase routines: i) the head-complement phrase construction routine (6.22), which constructs a head-complement phrase headed by *'çok'*, and attaches it as a non-head daughter in the active phrase [1]; and ii) the head-adjunct phrase construction routine (6.25) (because of the *synsem*-valued MOD feature of *'çok'*), which constructs a new head-adjunct phrase, re-attaching the head-complement phrase headed by *'çok'* as its non-head daughter, and attaches the new phrase as a non-head daughter of the active phrase [1]. Note that the yet-unencountered head daughter, [4], of the head-adjunct phrase is pushed onto the stack, hence being predicted. Notice also that the SYNSEM value of that predicted daughter is structure-shared with the MOD value of the non-head daughter of the head-adjunct phrase, hence constrained as AdjP, due to the relevant constraint on the type *hd-adjunct-ph*.

3. The second word *'çok'* again triggers calls for two phrase routines: i) (6.22), as before; and ii) the specification routine for head-adjunct phrases (6.24), which constrains the active phrase [4] as *hd-adjunct-ph*, popping it off the stack. Again, the as yet unencountered head daughter [6] is pushed onto the stack.

4. The third word *'yaşlı'* also triggers calls for two phrase routines: i) the specification routine for head-complement phrases (6.21), which constrains the active phrase [6] as of type *hd-comp-ph*, popping it off the stack; and ii) the specification routine for head-adjunct phrases (6.24), which constrains the new active phrase [1] as of type *hd-adjunct-ph*, popping it off the stack. The as yet unencountered head daughter, [8], of the outmost head-adjunct phrase is pushed onto the stack as usual.

5. The fourth word *'bir'* is again projected as a head-complement phrase by a call for (6.22), and its *synsem*-valued SPEC feature triggers an additional call for the specification

routine for head-specifier phrases (6.24), which constrains the active phrase [8] as *hd-spr-ph*, popping it off the stack. The as yet unencountered head daughter, [10], of that phrase is pushed onto the stack. Notice that the SYNSEM value of that predicted daughter is structure-shared with the SPEC value of the non-head daughter, hence constrained as N′, due to the SPEC Principle imposed on the type *hd-spr-ph*.

6. Finally, the fifth word *'adam'* is simply attached as the head daughter of the active phrase [10] by a call for (6.21), which further constrains that phrase as *hd-comp-ph*, as usual, popping it off the stack. Notice that the stack of predicted phrases is now empty, satisfying the requirement by the Grammaticality Principle, (6.3), for the sentence to be grammatical.

(6.32)

1. [1] [*hd-ph-pr*] PRED-PHRASES : < [1] >

↓ **çok** **very** [*word* MOD AdjP] (6.22) and (6.25)

2. [1] [*hd-ph-pr*
NON-HD-DTRS ⟨ [*hd-adjunct-ph*
SYNSEM [HEAD [3]]
NON-HD-DTRS ⟨ [*hd-comp-ph* PHON ⟨*çok*⟩ MOD [2] AdjP HD-DTR *word*] ⟩
HD-DTR [4] [*hd-ph-pr* SYNSEM [2] [HEAD [3]]]] ⟩]

PRED-PHRASES : < [4], [1] >

↓ **çok** **very** [*word* MOD AdjP] (6.22) and (6.24)

3. [1] [*hd-ph-pr*
NON-HD-DTRS ⟨ [*hd-adjunct-ph*
SYNSEM [HEAD [3]]
NON-HD-DTRS ⟨ [*hd-comp-ph* PHON ⟨*çok*⟩ MOD [2] AdjP HD-DTR *word*] ⟩
HD-DTR [*hd-adjunct-ph*
SYNSEM [2] [HEAD [3]]
NON-HD-DTRS ⟨ [*hd-comp-ph* PHON ⟨*çok*⟩ MOD [5] AdjP HD-DTR *word*] ⟩
HD-DTR [6] [*hd-ph-pr* SYNSEM [5] [HEAD [3]]]]] ⟩]

PRED-PHRASES : < $\boxed{6}$,$\boxed{1}$ >

↓ **yaşlı** $\begin{bmatrix} \textit{word} \\ \text{MOD} \quad \text{NP} \end{bmatrix}$ (6.21) and (6.24)

old

4. $$\begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \begin{bmatrix}\text{HEAD} & \boxed{7}\,\textit{noun}\\ \text{SPR} & \langle\rangle\end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \left[\text{HEAD } \boxed{3}\left[\text{MOD } \boxed{9}\,\text{NP}\right]\right] \\ \text{NON-HD-DTRS} & \left\langle\begin{bmatrix}\textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{çok}\rangle \\ \text{MOD} & \boxed{2}\,\text{AdjP}\\ \text{HD-DTR} & \textit{word}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \boxed{2}\left[\text{HEAD }\boxed{3}\right] \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix}\textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{çok}\rangle\\ \text{MOD} & \boxed{5}\,\text{AdjP}\\ \text{HD-DTR} & \textit{word}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \begin{bmatrix}\textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{yaşlı}\rangle\\ \text{SYNSEM} & \boxed{5}\left[\text{HEAD }\boxed{3}\right]\\ \text{HD-DTR} & \textit{word}\end{bmatrix}\end{bmatrix}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \boxed{8}\begin{bmatrix}\textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{9}\begin{bmatrix}\text{HEAD} & \boxed{7}\\ \text{SPR} & \langle\rangle\end{bmatrix}\end{bmatrix}\end{bmatrix}$$

PRED-PHRASES : < $\boxed{8}$ >

↓ **bir** $\begin{bmatrix} \textit{word} \\ \text{HEAD} \quad \textit{det}\left[\text{SPEC N}'\right] \end{bmatrix}$ (6.22) and (6.24)

a

5.

```
[ hd-adjunct-ph
  SYNSEM         [ HEAD [7] noun
                   SPR  ⟨⟩ ]
  NON-HD-DTRS  ⟨ [ hd-adjunct-ph
                   SYNSEM       [ HEAD [3] [ MOD [9] NP ] ]
                   NON-HD-DTRS  ⟨ [ hd-comp-ph
                                    PHON    ⟨çok⟩
                                    MOD     [2] AdjP
                                    HD-DTR  word ] ⟩
                   HD-DTR       [ hd-adjunct-ph
                                  SYNSEM       [2] [HEAD [3]]
                                  NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                   PHON    ⟨çok⟩
                                                   MOD     [5] AdjP
                                                   HD-DTR  word ] ⟩
                                  HD-DTR       [ hd-comp-ph
                                                 PHON    ⟨yaşlı⟩
                                                 SYNSEM  [5] [HEAD [3]]
                                                 HD-DTR  word ] ] ] ⟩
  HD-DTR         [ hd-spr-ph
                   SYNSEM       [9] [ HEAD [7]
                                      SPR  ⟨⟩ ]
                   NON-HD-DTRS  ⟨ [ hd-comp-ph
                                    PHON    ⟨bir⟩
                                    SYNSEM  [12] [ SPEC [11] N' ]
                                    HD-DTR  word ] ⟩
                   HD-DTR       [10] [ hd-ph-pr
                                       SYNSEM [11] [ HEAD [7]
                                                     SPR  ⟨[12]⟩ ] ] ] ]
```

PRED-PHRASES : < [10] >

↓ **adam** [*word* SYNSEM N′] (6.21)

man

6.

```
[ hd-adjunct-ph
  SYNSEM        [ HEAD  [7] noun
                  SPR   ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-adjunct-ph
                    SYNSEM       [ HEAD [3] [ MOD [9] NP ] ]
                    NON-HD-DTRS  ⟨ [ hd-comp-ph
                                     PHON    ⟨çok⟩
                                     MOD     [2] AdjP
                                     HD-DTR  word ] ⟩
                    HD-DTR       [ hd-adjunct-ph
                                   SYNSEM       [2] [HEAD [3]]
                                   NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                    PHON    ⟨çok⟩
                                                    MOD     [5] AdjP
                                                    HD-DTR  word ] ⟩
                                   HD-DTR       [ hd-comp-ph
                                                  PHON    ⟨yaşlı⟩
                                                  SYNSEM  [5] [HEAD [3]]
                                                  HD-DTR  word ] ] ] ⟩
  HD-DTR        [ hd-spr-ph
                  SYNSEM       [9] [ HEAD  [7]
                                     SPR   ⟨⟩ ]
                  NON-HD-DTRS  ⟨ [ hd-comp-ph
                                   PHON    ⟨bir⟩
                                   SYNSEM  [12] [ SPEC [11] N' ]
                                   HD-DTR  word ] ⟩
                  HD-DTR       [ hd-comp-ph
                                 PHON    ⟨adam⟩
                                 SYNSEM  [11] [ HEAD  [7]
                                                SPR   ⟨[12]⟩ ]
                                 HD-DTR  word ] ] ]
```

PRED-PHRASES : <>

Next, consider (6.33), a complex sentence with an embedded nominalization clause which also contains an accusative possessive NP. Note once again that I assume a flat structure analysis for Turkish sentences, and a subject analysis for possessors in Turkish.

(6.33) Ban-a $[_{S[fact]}$ $[_{NP}$ Mehmet'-in kitab-ı-nı] bir adam-ın
I-DAT Mehmet-GEN book-3sPoss-ACC a man-GEN
oku-duğ-u-nu] Berfu söyle-di.
read-FACT-3sPoss-ACC Berfu tell-PAST
'BERFU told me that, as for Mehmet's book, a man was reading it.'

The parse of (6.33) is again illustrated step by step in (6.34), whose steps are explained below.

1. At the beginning of the parse, STRUCT is constrained as an object of type *hd-ph-pr*, and it is the only element on the stack.

2. During the processing of the first word *'bana'* the construction routine for head-complement phrases, (6.22), is called, which constructs a new head-complement phrase headed by that word, and attaches it as a non-head daughter of the active phrase, [1].

3. The second word *'Mehmet'in'* again only triggers a call for (6.22), which constructs a new head-complement phrase headed by that word, and attaches it as a non-head daughter of the active phrase.

4. The processing of the third word *'kitabını'* involves a call for the head-complement phrase construction routine (6.22), as usual, followed by a call for the head-subject phrase construction routine (6.27) (due to the non-empty SUBJ value of the new word). (6.22) constructs a new head-complement phrase headed by the new word, which is then attached as the head daughter of the new head-subject phrase constructed by (6.27). Notice that the *genitive* NP, *'Mehmet'in'*, that has previously been attached as a non-head daughter of the active phrase [1] (at step 3) is re-attached as a non-head daughter of the new head-subject phrase. Furthermore, this new phrase is attached as a non-head daughter of the active phrase, in addition to the previously attached dative NP, *'bana'*.

5. The processing of the fourth word *'bir'* again involves calls for two phrase construction routines, (6.22), as usual, and (6.25), the construction routine for head-specifier phrases (because of the *synsem*-valued SPEC feature of *'bir'*). (6.22) constructs a new

head-complement phrase headed by the new word, which is then attached as the non-head daughter of the new head-specifier phrase constructed by (6.25) that is further attached as a non-head daughter of the active phrase $\boxed{1}$. Note that the as yet unencountered head daughter, $\boxed{6}$, of the new head-specifier phrase (whose SYNSEM value is structure-shared with the SPEC value of the non-head daughter – tag $\boxed{5}$ – due to the SPEC Principle on the type *hd-spr-ph*) is pushed onto the stack, hence becoming the active phrase.

6. The genitive N' *'adamın'*, once encountered, is simply attached as the head daughter of the active phrase, $\boxed{6}$, by a call for (6.21) (the specification routine for head-complement phrases), which also constrains the active phrase as of type *hd-comp-ph*, popping it off the stack.

7. When the *fact* verb *'okuduğunu'*, which subcategorizes for a *genitive* NP and an *accusative* NP, is encountered, it triggers a call for (6.22), which constructs a new head-complement phrase headed by that word, and re-attaches the two most-recently attached *genitive* and *accusative* NPs as non-head daughters of the new head-complement phrase that is further attached as a non-head daughter of the active phrase, $\boxed{1}$ (in addition to the previously attached dative NP non-head daughter, *'bana'*).

8. At step 8, with a call for (6.22), the new word *'Berfu'* is attached as the head daughter of a new head-complement phrase, which is further attached as a non-head daughter of the active phrase, $\boxed{1}$.

9. Finally, at step 9, the finite verb *'söyledi'*, which subcategorizes for all three non-head daughters already attached to the active phrase – although not in the same order – is simply attached as the head daughter of that phrase by a call for (6.21) (the specification routine for head-complement phrases), which also constrains the active phrase as of type *hd-comp-ph* and pops it off the stack. Notice that the stack of predicted phrases is now empty, satisfying the requirement by the Grammaticality Principle, (6.3), for the sentence to be grammatical.

(6.34)

1. $\boxed{1}$ [*hd-ph-pr*] PRED-PHRASES : < $\boxed{1}$ >

$\downarrow$ **bana** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \end{bmatrix}$ (6.22)

I-DAT

2. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{bana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : < $\boxed{1}$ >

$\downarrow$ **Mehmet'in** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \end{bmatrix}$ (6.22)

Mehmet-GEN

3. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Mehmet'in} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{bana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : < $\boxed{1}$ >

$\downarrow$ **kitabını** $\begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{noun}[\text{CASE } \textit{acc}] \\ \text{SUBJ} & \langle \text{NP}[\textit{gen}] \rangle \end{bmatrix}$ (6.22) and (6.27)

book-3sPoss-ACC

4.
$$
\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-subj-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13}\,\textit{noun}[\textit{acc}] \\ \text{SUBJ} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Mehmet'in} \rangle \\ \text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kitabını} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13} \\ \text{SUBJ} & \langle\boxed{2}\rangle \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{bana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}
$$

PRED-PHRASES : < $\boxed{1}$ >

↓ **bir** $\begin{bmatrix} \textit{word} & \\ \text{HEAD} & \textit{det}\,[\text{SPEC N}'] \end{bmatrix}$ (6.22) and (6.25)

a

5.
$$
\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-spr-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3}\,\textit{noun} \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{bir} \rangle \\ \text{SYNSEM} & \boxed{4}\,[\text{SPEC}\ \boxed{5}\,\text{N}'] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \boxed{6}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{5}\begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\boxed{4}\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-subj-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13}\,\textit{noun}[\textit{acc}] \\ \text{SUBJ} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Mehmet'in} \rangle \\ \text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kitabını} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13} \\ \text{SUBJ} & \langle\boxed{2}\rangle \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{bana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}
$$

PRED-PHRASES : < $\boxed{6}$,$\boxed{1}$ >

↓ **adamın** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{N}'[\textit{gen}] \end{bmatrix}$ (6.21)

man-GEN

6. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} \\ \text{NON-HD-DTRS} \left\langle \begin{bmatrix} \textit{hd-spr-ph} \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3}\,\textit{noun}[\textit{gen}] \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} \\ \text{PHON} & \langle \textit{bir} \rangle \\ \text{SYNSEM} & \boxed{4}\begin{bmatrix} \text{SPEC} & \boxed{5}\,\text{N}' \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \boxed{6}\begin{bmatrix} \textit{hd-comp-ph} \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \boxed{5}\begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\boxed{4}\rangle \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-subj-ph} \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13}\,\textit{noun}[\textit{acc}] \\ \text{SUBJ} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} \\ \text{PHON} & \langle \textit{Mehmet'in} \rangle \\ \text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} \\ \text{PHON} & \langle \textit{kitabını} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{13} \\ \text{SUBJ} & \langle\boxed{2}\rangle \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} \\ \text{PHON} & \langle \textit{bana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : $< \boxed{1} >$

$$\downarrow \ \textbf{okuduğunu} \ \begin{bmatrix} \textit{word} \\ \text{HEAD} & \textit{verb}\begin{bmatrix} \text{VFORM} & \textit{nomin}[\textit{acc}] \end{bmatrix} \\ \text{COMPS} & \langle \text{NP}[\textit{gen}], \text{NP}[\textit{acc}] \rangle \end{bmatrix} \qquad (6.22)$$

read-FACT-3sPoss-ACC

$$
7.\quad \boxed{1}\begin{bmatrix}
\textit{hd-ph-pr} & \\
\textsc{non-hd-dtrs} & \left\langle \begin{bmatrix}
\textit{hd-comp-ph} & \\
\textsc{synsem} & \begin{bmatrix} \textsc{head} & \boxed{3}\,\mathit{verb}\begin{bmatrix}\textsc{vform} & \mathit{nomin[acc]}\end{bmatrix} \\ \textsc{comps} & \langle\rangle \end{bmatrix} \\
\textsc{non-hd-dtrs} & \left\langle \begin{bmatrix}
\textit{hd-spr-ph} & \\
\textsc{synsem} & \boxed{7}\begin{bmatrix} \textsc{head} & \boxed{3}\,\mathit{noun[gen]} \\ \textsc{spr} & \langle\rangle \end{bmatrix} \\
\textsc{non-hd-dtrs} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \textsc{phon} & \langle \mathit{bir} \rangle \\ \textsc{synsem} & \boxed{4}\begin{bmatrix}\textsc{spec} & \boxed{5}\,\mathrm{N}'\end{bmatrix} \\ \textsc{hd-dtr} & \mathit{word} \end{bmatrix} \right\rangle \\
\textsc{hd-dtr} & \boxed{6}\begin{bmatrix} \textit{hd-comp-ph} & \\ \textsc{phon} & \langle \mathit{adamın} \rangle \\ \textsc{synsem} & \boxed{5}\begin{bmatrix} \textsc{head} & \boxed{3} \\ \textsc{spr} & \langle\boxed{4}\rangle \end{bmatrix} \\ \textsc{hd-dtr} & \mathit{word} \end{bmatrix}
\end{bmatrix},
\begin{bmatrix}
\textit{hd-subj-ph} & \\
\textsc{synsem} & \boxed{8}\begin{bmatrix} \textsc{head} & \boxed{13}\,\mathit{noun[acc]} \\ \textsc{subj} & \langle\rangle \end{bmatrix} \\
\textsc{non-hd-dtrs} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \textsc{phon} & \langle \mathit{Mehmet'in} \rangle \\ \textsc{synsem} & \boxed{2}\,\mathrm{NP}[\mathit{gen}] \\ \textsc{hd-dtr} & \mathit{word} \end{bmatrix} \right\rangle \\
\textsc{hd-dtr} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \textsc{phon} & \langle \mathit{kitabını} \rangle \\ \textsc{synsem} & \begin{bmatrix} \textsc{head} & \boxed{13} \\ \textsc{subj} & \langle\boxed{2}\rangle \end{bmatrix} \\ \textsc{hd-dtr} & \mathit{word} \end{bmatrix}
\end{bmatrix} \right\rangle \\
\textsc{hd-dtr} & \begin{bmatrix} \mathit{word} & \\ \textsc{phon} & \langle \mathit{okuduğunu} \rangle \\ \textsc{synsem} & \begin{bmatrix} \textsc{head} & \boxed{3} \\ \textsc{comps} & \langle\boxed{7},\boxed{8}\rangle \end{bmatrix} \end{bmatrix}
\end{bmatrix},
\begin{bmatrix} \textit{hd-comp-ph} & \\ \textsc{phon} & \langle \mathit{bana} \rangle \\ \textsc{synsem} & \mathrm{NP}[\mathit{dat}] \\ \textsc{hd-dtr} & \mathit{word} \end{bmatrix} \right\rangle
\end{bmatrix}
$$

PRED-PHRASES : < $\boxed{1}$ >

$$
\downarrow \quad \textbf{Berfu} \quad \begin{bmatrix} \mathit{word} & \\ \textsc{synsem} & \mathrm{NP}[\mathit{nom}] \end{bmatrix} \qquad (6.22)
$$

Berfu

```
8. [1] [ hd-ph-pr
         NON-HD-DTRS ⟨ [ hd-comp-ph
                         PHON    ⟨Berfu⟩
                         SYNSEM  NP[nom]
                         HD-DTR  word ] ,
           [ hd-comp-ph
             SYNSEM [ HEAD  [3] verb [ VFORM nomin[acc] ]
                      COMPS ⟨⟩ ]
             NON-HD-DTRS ⟨ [ hd-spr-ph
                             SYNSEM [7] [ HEAD [3] noun[gen]
                                          SPR  ⟨⟩ ]
                             NON-HD-DTRS ⟨ [ hd-comp-ph
                                             PHON   ⟨bir⟩
                                             SYNSEM [4] [ SPEC [5] N' ]
                                             HD-DTR word ] ⟩
                             HD-DTR [6] [ hd-comp-ph
                                          PHON   ⟨adamın⟩
                                          SYNSEM [5] [ HEAD [3]
                                                       SPR  ⟨[4]⟩ ]
                                          HD-DTR word ] ] ,
                           [ hd-subj-ph
                             SYNSEM [8] [ HEAD [13] noun[acc]
                                          SUBJ ⟨⟩ ]
                             NON-HD-DTRS ⟨ [ hd-comp-ph
                                             PHON   ⟨Mehmet'in⟩
                                             SYNSEM [2] NP[gen]
                                             HD-DTR word ] ⟩
                             HD-DTR [ hd-comp-ph
                                      PHON   ⟨kitabını⟩
                                      SYNSEM [ HEAD [13]
                                               SUBJ ⟨[2]⟩ ]
                                      HD-DTR word ] ] ⟩
             HD-DTR [ word
                      PHON   ⟨okuduğunu⟩
                      SYNSEM [ HEAD  [3]
                               COMPS ⟨[7],[8]⟩ ] ] ] ,
           [ hd-comp-ph
             PHON   ⟨bana⟩
             SYNSEM NP[dat]
             HD-DTR word ] ⟩ ]
```

PRED-PHRASES : < [1] >

```
↓ söyledi  [ word
             HEAD   verb [VFORM fin]
             COMPS  ⟨NP[nom], NP[dat], S[fact, acc]⟩ ]      (6.21)

  tell-PAST
```

9.

```
[ hd-comp-ph
  SYNSEM        [ HEAD   [9] verb[fin]
                  COMPS  ⟨⟩ ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON    ⟨Berfu⟩
                    SYNSEM  [10] NP[nom]
                    HD-DTR  word ] ,
                  [ hd-comp-ph
                    SYNSEM       [11] [ HEAD   [3] verb [ VFORM  nomin[acc] ]
                                        COMPS  ⟨⟩ ]
                    NON-HD-DTRS  ⟨ [ hd-spr-ph
                                     SYNSEM       [7] [ HEAD  [3] noun[gen]
                                                        SPR   ⟨⟩ ]
                                     NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                      PHON    ⟨bir⟩
                                                      SYNSEM  [4] [ SPEC  [5] N' ]
                                                      HD-DTR  word ] ⟩
                                     HD-DTR       [6] [ hd-comp-ph
                                                        PHON    ⟨adamın⟩
                                                        SYNSEM  [5] [ HEAD  [3]
                                                                      SPR   ⟨[4]⟩ ]
                                                        HD-DTR  word ] ] ,
                                   [ hd-subj-ph
                                     SYNSEM       [8] [ HEAD  [13] noun[acc]
                                                        SUBJ  ⟨⟩ ]
                                     NON-HD-DTRS  ⟨ [ hd-comp-ph
                                                      PHON    ⟨Mehmet'in⟩
                                                      SYNSEM  [2] NP[gen]
                                                      HD-DTR  word ] ⟩
                                     HD-DTR       [ hd-comp-ph
                                                    PHON    ⟨kitabını⟩
                                                    SYNSEM  [ HEAD  [13]
                                                              SUBJ  ⟨[2]⟩ ]
                                                    HD-DTR  word ] ] ⟩
                    HD-DTR       [ word
                                   PHON    ⟨okuduğunu⟩
                                   SYNSEM  [ HEAD   [3]
                                             COMPS  ⟨[7],[8]⟩ ] ] ] ,
                  [ hd-comp-ph
                    PHON    ⟨bana⟩
                    SYNSEM  [12] NP[dat]
                    HD-DTR  word ] ⟩
  HD-DTR        [ word
                  PHON    ⟨söyledi⟩
                  SYNSEM  [ HEAD   [9]
                            COMPS  ⟨[10],[12],[11]⟩ ] ] ]
```

PRED-PHRASES : <>

6.5 Exploiting the CASE Values

Let us now consider the routine 'check-CASE', presented in (6.18) in pseudo-code notation, which is responsible for the incorporation in the algorithm of the exploitation of the CASE values in the way discussed in Chapter 5 (Section 5.6).

(6.35) check-CASE

1. ActPhrase = top(PRED-PHRASES)
2. NonHeadDtrs = get-NON-HD-DTRS(ActPhrase)
3. if check-NON-HD-DTRS-cases(NonHeadDtrs) then
   3.1. split-NON-HD-DTRS(FirstNonHeadDtrs,RestNonHeadDtrs, NonHeadDtrs)

   3.2. NewActPhrase ← $\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \text{FirstNonHeadDtrs} \end{bmatrix}$

   3.3. get-NON-HD-DTRS(ActPhrase) ← $\begin{bmatrix} \textit{nelist-ph} & \\ \text{FIRST} & \text{NewActPhrase} \\ \text{REST} & \text{RestNonHeadDtrs} \end{bmatrix}$

   3.4. PRED-PHRASES ← push(NewActPhrase)

(6.35) checks (at step 3) whether the CASE value of the non-head daughter that has just been attached to the active phrase requires the construction of a new phrase, considering the CASE values of all the non-head daughters in the active phrase. If so, it constructs a new phrase of type *hd-ph-pr*, and lowers a number of the most recently attached non-head daughters of the active phrase (FirstNonHeadDtrs) as the non-head daughters of the newly constructed phrase. It further attaches the new phrase as a non-head daughter of the current active phrase, in addition to the previously attached (and not lowered) non-head daughters (RestNonHeadDtrs). Finally, the newly constructed headed phrase is pushed onto the stack, becoming the new active phrase.

The essential step in (6.35) is step 3.1, where the parser 'non-deterministically' splits the list of the non-head daughters in the active phrase into the two lists FirstNonHeadDtrs (i.e. the daughters to be lowered as the non-head daughters of the new phrase) and

RestNonHeadDtrs (i.e. the daughters to be left as non-head daughters of the current active phrase). Following the discussion in Chapter 5 (Section 5.6), these two lists must satisfy either of the following two constraints:

i) There exists a phrase in RestNonHeadDtrs with a CASE value of type *nominative/genitive*, and the first element of FirstNonHeadDtrs (i.e. the non-head daughter just being attached) is also a phrase with a *nominative/genitive* CASE value (but not a *nominative, nonspecific* NP, in which case no restructuring is necessary).

ii) There exists a phrase in RestNonHeadDtrs with a CASE value of type *accusative/dative*, and the first element of FirstNonHeadDtrs is also a phrase with the same CASE value.

In other words, the two non-head daughters that trigger the construction of the new phrase (because of their particular CASE values) are to end up as non-head daughters of different phrases (the original active phrase and the newly constructed phrase).

Consider now the parse of (6.36), illustrated step by step in (6.37), as an example where the CASE values would be of use to the parser in the way discussed above. Since the first two NPs in (6.36), *'kadın'* and *'adamın'*, have CASE values of type *nominative* and *genitive*, respectively, the second NP is attached as a non-head daughter of an embedded phrase, [2], sister to the first NP (see Chapter 5 – Section 5.6.1, page 155 – for the relevant restriction), and that phrase is pushed onto the stack, as shown at step 3 in (6.37). Then, the embedded verb *'yürüdüğünü'*, once encountered, is simply attached as the head daughter of that embedded phrase, as shown at step 4.

(6.36) Kadın [adam-ın yürü-düğ-ü-nü] gör-dü.
woman man-GEN walk-FACT-3sPoss-ACC see-PAST
'The woman saw that the man was walking.'

(6.37)

1. [1] [*hd-ph-pr*] PRED-PHRASES : < [1] >

↓ **kadın** $\begin{bmatrix} \textit{word} \\ \text{SYNSEM} \quad \text{NP}[\textit{nom}] \end{bmatrix}$ (6.22)

woman

2. $\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kadın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$

PRED-PHRASES : < $\boxed{1}$ >

↓ **adamın** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \end{bmatrix}$ (6.22) and (6.35)

man-GEN

3.

$$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \boxed{2}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kadın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : < $\boxed{2}$, $\boxed{1}$ >

↓ **yürüdüğünü** $\begin{bmatrix} \textit{word} & \\ \text{COMPS} & < \text{NP}[\textit{gen}] > \end{bmatrix}$ (6.21)

walk-FACT-3sPoss-ACC

4. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \textit{verb}\begin{bmatrix}\text{VFORM} & \textit{nomin}[\textit{acc}]\end{bmatrix} \\ \text{COMPS} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \boxed{4}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{yürüdüğünü} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{COMPS} & \langle \boxed{4} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kadın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix} \quad \text{PRED-PHRASES} : <\boxed{1}>$$

$$\downarrow \quad \textbf{gördü} \quad \begin{bmatrix} \textit{word} & \\ \text{COMPS} & <\text{NP}[\textit{nom}], \text{S}[\textit{fact}, \textit{acc}]> \end{bmatrix} \qquad (6.21)$$

see-PAST

5. $$\begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{5}\,\textit{verb}[\textit{fin}] \\ \text{COMPS} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \boxed{6}\begin{bmatrix} \text{HEAD} & \textit{verb}\begin{bmatrix}\text{VFORM} & \textit{nomin}[\textit{acc}]\end{bmatrix} \\ \text{COMPS} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \boxed{4}\,\text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{yürüdüğünü} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{COMPS} & \langle \boxed{4} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kadın} \rangle \\ \text{SYNSEM} & \boxed{7}\,\text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{gördü} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{5} \\ \text{COMPS} & \langle \boxed{7}, \boxed{6} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \quad \text{PRED-PHRASES} : <>$$

6.6 Incremental Processing of Long-distance Topicalization

In this section, I discuss the details of the routine 'check-fronting', which enables the parser to recover the filler-gap relations in long-distance topicalization in an incremental fashion.

Consider first the parse of (6.38) to get a better insight into the behaviour of the parser, adopting such a strategy. Note that in (6.38) the *genitive* NP subject, *'adamın'*, of the embedded *fact* clause has been long-distance topicalized, appearing in the sentence-initial position of the matrix clause.

(6.38) [Adam-ın$_i$ [ben san-a [_$_i$ kadın-ı gör-düğ-ü-nü] söyle-di-m.]]
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC tell-PAST-1SG
'As for the man, I have told you that he has seen the woman.'

The first two NPs, *'adamın'* and *'ben'*, are *genitive* and *nominative, specific*, respectively, hence *'ben'* is attached as a non-head daughter of an embedded clause sister to *'adamın'* (see Section 6.5 on exploiting the CASE values in structuring phrases in this way). After that the parse proceeds rather smoothly, and the NPs *'sana'* and *'kadını'* are simply attached as sisters of *'ben'*. The processing of the *fact* verb *'gördüğünü'*, however, triggers two rather independent kinds of restructuring of the current structure. Firstly, the *accusative* NP *'kadını'* is lowered as a non-head daughter of the embedded *fact* clause headed by the new word, in the usual way. Secondly, a search is triggered (by the non-empty INHER|SLASH value of the new word) for an already encountered *genitive* NP that could be interpreted as the "missing" subject complement of the embedded *fact* clause. This search process only considers the non-head daughters of the phrases on the stack as a potential filler. Once such a filler is found (in this case *'adamın'*), that filler is interpreted as the missing subject of the embedded clause by structure-sharing its LOCAL value with the *local* object in the INHER|SLASH set of *'gördüğünü'* (which corresponds to the LOCAL value of its subject complement that has been extracted by the subject extraction lexical rule; see Chapter 3, page72). Moreover, the phrase where the search succeeds (in this case the global structure, STRUCT) is constrained as of type *hd-topic-ph*, and *removed* from the stack. (Note that this removal cannot be called popping, since STRUCT is not on the top of the stack at that point.)

The routine 'check-fronting', which performs the search and restructuring processes mentioned above, is summarized informally in (6.39). Step 1 of (6.39) performs a 'non-deterministic' search for each *local* object in the INHER|SLASH set of the new word (that is, for each element that corresponds to the LOCAL value of a complement that has been extracted by one of the extraction lexical rules). The search process considers only the phrases on the stack. In each of those phrases, any non-head daughter, except for the leftmost one, is checked to see if it can fill in the gap (that is, if the LOCAL value of that particular non-head daughter *unifies* with the respective *local* object in the INHER|SLASH set of the new word). Note that in none of the phrases is the leftmost non-head daughter considered as a potential filler. For the current active phrase, the reason is that if the leftmost daughter could actually fill in the gap, then it would have initially been lowered as a non-head daughter of the head-complement phrase headed by the new word, rather than being interpreted as long-distance topicalized. For the rest of the phrases, the reason relates to the fact that in each of those phrases the leftmost daughter is a phrase of type *hd-ph-pr* that has been constructed and pushed onto the stack by the routine 'check-CASE' in (6.35), and hasn't yet been fully specified (encountered).

Once an appropriate filler is found, the parser constructs a new phrase of type *hd-topic-ph* (step 2.1). If the filler is the second but leftmost non-head daughter in the phrase where the search has succeeded, then the leftmost daughter of that phrase (which is always a *hd-ph-pr* object constructed by 'check-CASE' as mentioned above) is removed from the NON-HD-DTRS list of that phrase (step 2.2.1), and is attached as the head daughter of the newly constructed head-topic phrase (step 2.2.2). (Note that that head daughter has already been pushed onto the stack by 'check-CASE' when it was first constructed.) On the other hand, if there are any fully encountered non-head daughters that precede the filler in the same phrase, the parser constructs a new phrase of type *hd-ph-pr* (step 2.3.1), removes all those daughters preceding the filler from the NON-HD-DTRS list of the phrase that they were previously attached (step 2.3.2), and re-attaches them as non-head daughters of the newly constructed *hd-ph-pr* phrase (step 2.3.3). And finally, it attaches that new phrase as the head daughter of the head-topic phrase (step 2.3.4), and also *inserts* it in the stack such that it is just above the phrase among whose non-head daughters the filler has originally been found (step 2.3.5).

Finally, step 2.4 non-deterministically performs either a specification or a construction step:

- A specification step is achieved simply by unifying the head-topic phrase constructed at step 2.1 with the phrase where the filler has originally been found (step 2.4.1.1), and removing that phrase from the stack, since it has now been fully specified (step 2.4.1.2).

- A construction step involves the removal of the filler from the NON-HD-DTRS list of the phrase where the search has originally succeeded (step 2.4.2.1), and its re-attachment as the non-head daughter of the head-topic phrase constructed at step 2.1 (step 2.4.2.2), and finally, the attachment of that head-topic phrase as the leftmost non-head daughter of the phrase from which the filler has just been removed (step 2.4.2.3).

(6.39) **check-fronting(NewWordInherSlash)**

for each element Gap ∈ NewWordInherSlash do

1. (ND) search for a Phrase in PRED-PHRASES

   1.1. (ND) looking for FillerDtr among the non-head daughters in Phrase
   (not the leftmost non-head daughter)
   such that Gap *unifies* with the LOCAL value of FillerDtr
   (i.e. the respective gap can be filled with this non-head daughter)

2. if such a FillerDtr is found among the non-head daughters of Phrase then

   2.1. construct HeadTopicPhrase of type *hd-topic-ph*

   2.2. if FillerDtr is the second non-head daughter in Phrase then

      2.2.1. remove the first non-head daughter from
      the NON-HEAD-DTS list of Phrase

      2.2.2. re-attach that daughter as the HD-DTR of HeadTopicPhrase

   2.3. else

      2.3.1. construct HeadDtr of type *hd-ph-pr*

      2.3.2. remove the non-head daughters that precede FillerDtr in Phrase
      from the NON-HEAD-DTS list of Phrase

      2.3.3. re-attach those daughters as non-head daughters of HeadDtr

      2.3.4. attach HeadDtr as the HD-DTR of HeadTopicPhrase

      2.3.5. *insert* HeadDtr in PRED-PHRASES just above Phrase

2.4. (ND)

2.4.1. specification step

2.4.1.1. unify HeadTopicPhrase with Phrase

2.4.1.2. *remove* Phrase from PRED-PHRASES

2.4.2. construction step

2.4.2.1. remove FillerDtr from the NON-HEAD-DTS list of Phrase

2.4.2.2. re-attach FillerDtr as the non-head daughter of HeadTopicPhrase

2.4.2.3. attach HeadTopicPhrase as the first (leftmost) non-head daughter of Phrase

The parse of (6.38), repeated below, is illustrated step by step in (6.40). Note the restructuring at step 6.2, once the processing of the embedded verb *'gördüğünü'* is completed. The *genitive* NP *'adamın'* already attached to STRUCT, [1], is interpreted as the missing subject complement of the embedded *fact* clause by structure-sharing its LOCAL value with the (only) element in the INHER|SLASH set of *'gördüğünü'*. The *hd-ph-pr* object [2] (the non-head daughter that precedes the filler *'adamın'* in [1]) is re-attached as the head daughter of the newly constructed head-topic phrase (cf. step 2.2 in (6.39)), and furthermore [1] (the phrase where the filler is found) is unified with that head-topic phrase and also removed from the stack (cf. the specification step 2.4.1 in (6.39)). When the main verb *'söyledim'* is finally encountered, it is simply attached as the head daughter of the active phrase, [2], as shown at step 7 in (6.40).

(6.38)	Adam-ın$_i$	[ben	san-a	[$\_{i}$ kadın-ı	gör-düğ-ü-nü]	söyle-di-m.]
	man-GEN	I	you-DAT	woman-ACC	see-FACT-3sPoss-ACC	tell-PAST-1SG

'As for the man, I have told you that he has seen the woman.'

(6.40)

1. $\boxed{1}$ [*hd-ph-pr*] PRED-PHRASES : < $\boxed{1}$ >

$\downarrow$ **adamın** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \end{bmatrix}$ (6.22)

man-GEN

2. $\boxed{1}$ $\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$

PRED-PHRASES : < $\boxed{1}$ >

$\downarrow$ **ben** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \end{bmatrix}$ (6.22) and (6.35)

I

3. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \boxed{2}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{ben} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : < 2 , 1 >

↓ **sana** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \end{bmatrix}$ (6.22)

you-DAT

4. $$\boxed{1}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \boxed{2}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{sana} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{dat}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{ben} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{adamın} \rangle \\ \text{SYNSEM} & \text{NP}[\textit{gen}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \end{bmatrix}$$

PRED-PHRASES : < 2 , 1 >

↓ **kadını** $\begin{bmatrix} \textit{word} & \\ \text{SYNSEM} & \text{NP}[\textit{acc}] \end{bmatrix}$ (6.22)

woman-ACC

5. [1] [*hd-ph-pr*
NON-HD-DTRS ⟨ [2] [*hd-ph-pr*
NON-HD-DTRS ⟨ [*hd-comp-ph* PHON ⟨*kadını*⟩ SYNSEM NP[*acc*] HD-DTR *word*], [*hd-comp-ph* PHON ⟨*sana*⟩ SYNSEM NP[*dat*] HD-DTR *word*], [*hd-comp-ph* PHON ⟨*ben*⟩ SYNSEM NP[*nom*] HD-DTR *word*] ⟩], [*hd-comp-ph* PHON ⟨*adamın*⟩ SYNSEM NP[*gen*] HD-DTR *word*] ⟩]

PRED-PHRASES : < [2], [1] >

↓ **gördüğünü** [*word* COMPS < NP[*acc*] > INHER|SLASH {np[*gen*]}] (6.22)

see-FACT-3sPoss-ACC

6.1. (before 'check-fronting')

[1] [*hd-ph-pr*
NON-HD-DTRS ⟨ [2] [*hd-ph-pr*
NON-HD-DTRS ⟨ [*hd-comp-ph*
SYNSEM [HEAD [3] *verb* [VFORM *nomin*[*acc*]] COMPS ⟨⟩ INHER|SLASH {[5] np[*gen*]}]
NON-HD-DTRS ⟨ [*hd-comp-ph* PHON ⟨*kadını*⟩ SYNSEM [4] NP[*acc*] HD-DTR *word*] ⟩
HD-DTR [*word* PHON ⟨*gördüğünü*⟩ SYNSEM [HEAD [3] COMPS ⟨[4]⟩ INHER|SLASH {[5]}]]], [*hd-comp-ph* PHON ⟨*sana*⟩ SYNSEM NP[*dat*] HD-DTR *word*], [*hd-comp-ph* PHON ⟨*ben*⟩ SYNSEM NP[*nom*] HD-DTR *word*] ⟩], [*hd-comp-ph* PHON ⟨*adamın*⟩ SYNSEM NP[*gen*] HD-DTR *word*] ⟩]

PRED-PHRASES : < [2], [1] >

6.2. (after 'check-fronting')

```
hd-topic-ph
SYNSEM          HEAD            [6] verb[non-base]
                INHER|SLASH     {}
NON-HD-DTRS   < hd-comp-ph
                PHON            ⟨adamın⟩
                SYNSEM|LOCAL    [5] np[gen]
                HD-DTR          word >
HD-DTR    [2]   hd-ph-pr
                SYNSEM          HEAD              [6]
                                INHER|SLASH       {…,[5],…}
                                TO-BIND|SLASH     {[5]}
                NON-HD-DTRS   < hd-comp-ph
                                SYNSEM        HEAD         [3] verb [VFORM nomin[acc]]
                                              COMPS        ⟨⟩
                                              INHER|SLASH  {[5] np[gen]}
                                NON-HD-DTRS < hd-comp-ph
                                              PHON         ⟨kadını⟩
                                              SYNSEM       [4] NP[acc]
                                              HD-DTR       word >
                                HD-DTR        word
                                              PHON         ⟨gördüğünü⟩
                                              SYNSEM       HEAD         [3]
                                                           COMPS        ⟨[4]⟩
                                                           INHER|SLASH  {[5]}
                              , hd-comp-ph
                                PHON          ⟨sana⟩
                                SYNSEM        NP[dat]
                                HD-DTR        word
                              , hd-comp-ph
                                PHON          ⟨ben⟩
                                SYNSEM        NP[nom]
                                HD-DTR        word >
```

PRED-PHRASES : < [2] >

↓ **söyledim** [word, COMPS < NP[*nom*], NP[*dat*], S[*fact, acc*] >] (6.21)

tell-PAST-1SG

7.

```
[ hd-topic-ph
  SYNSEM        [ HEAD          [6] verb[fin]
                  INHER|SLASH   {} ]
  NON-HD-DTRS   ⟨ [ hd-comp-ph
                    PHON            ⟨adamın⟩
                    SYNSEM|LOCAL    [5] np[gen]
                    HD-DTR          word ] ⟩
  HD-DTR        [ hd-comp-ph
                  SYNSEM        [ HEAD              [6]
                                  INHER|SLASH       {[5]}
                                  TO-BIND|SLASH     {[5]} ]
                  NON-HD-DTRS   ⟨ [ hd-comp-ph
                                    SYNSEM        [7] [ HEAD          [3] verb [ VFORM  nomin[acc] ]
                                                        COMPS         ⟨⟩
                                                        INHER|SLASH   {[5] np[gen]} ]
                                    NON-HD-DTRS   ⟨ [ hd-comp-ph
                                                      PHON      ⟨kadını⟩
                                                      SYNSEM    [4] NP[acc]
                                                      HD-DTR    word ] ⟩
                                    HD-DTR        [ word
                                                    PHON      ⟨gördüğünü⟩
                                                    SYNSEM    [ HEAD          [3]
                                                                COMPS         ⟨[4]⟩
                                                                INHER|SLASH   {[5]} ] ] ] ,
                                  [ hd-comp-ph
                                    PHON      ⟨sana⟩
                                    SYNSEM    [8] NP[dat]
                                    HD-DTR    word ] ,
                                  [ hd-comp-ph
                                    PHON      ⟨ben⟩
                                    SYNSEM    [9] NP[nom]
                                    HD-DTR    word ] ⟩
                  HD-DTR        [ word
                                  PHON      ⟨söyledim⟩
                                  SYNSEM    [ HEAD    [6]
                                              COMPS   ⟨[9],[8],[7]⟩ ] ] ] ]
```

PRED-PHRASES : <>

It is important to note once again that the search and restructuring processes in (6.39) only apply to the phrases in the stack. Consider, for instance, (6.41), where the *genitive* subject of the embedded *fact* clause has been long-distance topicalized in the mother *finite* S (HD-DTR of the *that*-clause) rather than the main sentence.

(6.41) Kadın san-dı [ki [çocuğ-un$_i$ [adam [_$_i$ yürü-düğ-ü-nü] gör-dü.]]]
woman think-PAST that child-GEN man walk-FACT-3sPoss-ACC see-PAST
'The woman thought that, as for the child, the man saw him walking.'

Note that during the processing of (6.41), at the time of encountering the embedded verb *'yürüdüğünü'*, the very initial active phrase that was on the stack at the beginning of the parse, i.e. STRUCT, would already have been popped (when the main verb *'sandı'* was processed), and hence *'kadın'*, which would have been attached to that phrase, would not be considered as a potential filler for the missing subject.

Also, recall from Chapter 5 (Section 5.7) that recovering the head-filler dependencies in this way is in no way guaranteed to provide all available readings in certain cases of ambiguity (cf. example (5.40) on page 161). Thus, one must also have a dummy routine with the same name ('check-fronting') that always simply returns *true* without any further action, in order to provide the other possible readings. Since readings provided by the incremental recovery of those dependencies are strongly preferred by humans, a serial parser, for example, should give priority to the 'check-fronting' routine summarized in (6.39) over any such dummy routine.

Finally, as discussed before in Chapter 5 (Section 5.7), in the case of potentially ambiguous examples such as (6.42), Turkish speakers have a strong preference for the 'woman-see/man-know' association, as opposed to the alternative 'man-see/woman-know' association.

(6.42) [Adam-ın$_{i/j}$ [kadın-ın$_{j/i}$ [çocuğ-a [[kitab-ı gör-düğ-ü-nü$_j$]
man-GEN woman-GEN child-DAT book-ACC see-FACT-3sPoss-ACC
bil-diğ-i-ni$_i$] ben söyle-di-m.]]]
know-FACT-3sPoss-ACC I tell-PAST-1SG
'I have told the child that the man/woman knew that the woman/man has seen the book.'

That fact suggests that in a (serial) implementation of this algorithm the search process mentioned in (6.39) should start in the current active phrase and proceed with the other phrases on the stack in turn, considering in each phrase any non-head daughter as a potential filler, starting from the second but leftmost daughter. (Recall that the leftmost daughter is never considered as a potential filler.) That would guarantee that in the case of ambiguity as in (6.42) above, the (serial) parser would favour the preferred reading, and return the other parse(s) only on backtracking.

6.7 Summary

In this chapter, I have presented the details of the parsing approach proposed in this dissertation for incremental processing of language, relying on an HPSG grammar. In this approach, parsing a string always starts with an underspecified global structure, and proceeds by attaching every word in the input string to that structure, thereby constraining the structure further and further with the processing of each word. In attaching a newly encountered word to the structure, the parser makes systematic use of the selection features of the HPSG formalism (in determining the type(s) of the phrase(s) that the word in question may be a part of), as well as taking into account the linear precedence constraints employed by the language being parsed.

Following the discussion in Chapter 5 on processing left recursive structures in a language, during the processing of an input string certain sub-structures of the global structure are *non-deterministically* left underspecified, whose daughters may later be nonmonotonically lowered in the structure, if need arises.

In Section 6.3, I discussed in detail the attachment procedures I have developed for Turkish, which also take into account the linear precedence constraints proposed for Turkish in Chapter 3 (Section 3.4.5). Then in Section 6.5, I presented the details of the incremental construction of embedded phrases/clauses while parsing Turkish, by exploiting certain restrictions on the co-occurrence of phrases with particular CASE values as sister constituents of the same clause in Turkish. Finally, in Section 6.6, I outlined the incremental processing of long-distance topicalization in Turkish. The processing of a newly encountered word with a non-empty INHER|SLASH value triggers a search in the structure for each of the "missing" arguments of the new word, which results in restructuring of the global structure in case of a *successful* search. I further made a number of suggestions that would enable a serial parser to capture strong preferences humans exhibit in certain cases of potentially ambiguous long-distance dependencies in Turkish.

Chapter 7

Computational, Psycholinguistic and Implementational Issues

This chapter is concerned with some further issues concerning the incremental parsing approach for HPSG grammars presented in Chapters 5 and 6. In Section 7.1, I examine the degree of non-determinism embodied in the algorithm, and suggest ways of improving the efficiency of a parser implemented using this algorithm. In Section 7.2, I focus on the question of psychological plausibility of the approach. Finally, in Section 7.3, I discuss certain implementational aspects of a parsing system implemented for Turkish in the LIFE programming language.

7.1 Computational Complexity

In this section, I concern myself with the degree of non-determinism that arises during the processing of an input string using the parsing algorithm presented in Chapter 6, and suggest ways of improving the efficiency of a parser implemented using that algorithm. Note that there are two sources of non-determinism in the main body of the algorithm, one due to lexical ambiguity, and the other due to (global or local) structural ambiguity.[1] In what follows, I focus on the latter source of non-determinism. Although

[1]'Local structural ambiguity' refers to cases where an initial sub-string of a given input has a number of possible analyses, which are nevertheless disambiguated by the end of the input. Consider, for example, the processing of the English 'that'-less clause (i), where the NP *'Scully'*, once encountered, may be interpreted either as an object of the main verb *'knows'*, or as the subject of its complement S (assuming that the kind of complement *'knows'* takes is left underspecified in its lexical entry). However, once the next verb

most of the discussion below is mainly based on the attachment procedures for Turkish presented in Chapter 6 (Section 6.3), many points directly relate to processing a head-final language.

As can be seen from the attachment routine in Chapter 6 (Section 6.3), the attachment of a single word may involve calls for up to 6 different phrase routines, that is, the ones for the types *hd-comp-ph*, *hd-subj-ph*, *hd-adjunct-ph*, *hd-spr-ph*, *hd-topic-ph*, and *hd-backg-ph*, depending on the existence of the respective special selection feature-value pair in the lexical entry of the word in question.[2] Considering that each phrase routine involves a non-deterministic choice point between a specification and a construction step, in the *worst* case processing a single word has a branching factor of 2^6. Note, however, that this is a rather unlikely situation in practice for two reasons: First, the number of special selection feature-value pairs in a lexical entry is on the average 2, and is, in general, no more than 3. Second, the success of applying a specification step is rather restricted due to the constraints imposed by the grammar on the predicted phrases during processing.

There would possibly be various ways of simulating the non-determinism in processing. One thing to point out is that any implementation of the algorithm will have to copy the state of the parse (that is, the global structure, STRUCT, and the stack of predicted phrases, PRED-PHRASES) at every non-deterministic choice point (unless it exploits some sort of ambiguity packing procedure to share parts of those structures that are common to a number of non-deterministic paths; see below). That can be seen as a real deficit, particularly since the parser will behave in exactly the same way, in different non-deterministic paths, for most parts of an input string, hence resulting in a good deal of similarity in the structures belonging to different paths. Note that the main complexity in this context arises from copying the global structure. In other words, copying the stack shouldn't lead to a big burden in terms of time or space complexity, since it merely consists of pointers to certain sub-structures of STRUCT.

Tomita (1986) introduces a way of efficient ambiguity packing in LR parsing of general context-free grammars by structuring stacks as directed acyclic graphs (DAGs). Milward (1994) adopts this idea in representing the state categories in his dynamic spec-

'loves' is encountered, it becomes clear that only the sentential complement analysis is possible. Note that this kind of ambiguity is a consequence of (left-to-right) incremental processing.

(i) Mulder knows Scully loves cats.

[2] Ignore for the time being the complexity introduced by the routine 'check-CASE', which I discuss later in the section.

ification of Lexicalised Dependency Grammars (Milward (1992)), as DAGs, and provides $O(n^3)$ 'recognition' time complexity for that specification, where 'n' stands for the number of words in a given input sentence. In the present approach, however, since STRUCT itself is a feature structure, i.e. a DAG in mathematical terms, packing the ambiguity at choice points in a kind of compact representation shared by a number of non-deterministic paths doesn't seem to be a trivial process at all. In case of interest, Pereira (1985) presents a method for this kind of sharing of structure for unification-based formalisms. It remains, however, to be shown whether/how that method could be adapted for a typed feature formalism. Note also that although Tomita (1987) presents an efficient method of ambiguity packing for augmented context-free grammars, that method crucially doesn't deal with re-entrancies ("for the sake of efficiency and simplicity"), hence only covers packing of *tree-structured* – rather than *graph-structured* – feature structures (or functional structures in Tomita's terminology).

The discussion in Chapters 5 and 6 does not propose any particular strategy to deal with the non-determinism involved in the present parsing approach (except for a number of suggestions made to enable a serial parser to capture certain preferences that native speakers of Turkish exhibit in processing certain cases of potentially ambiguous long-distance dependencies; cf. Sections 5.7 and 6.6). In the rest of this section, I assume a serial implementation of the algorithm that adopts a Prolog-like depth-first resolution strategy with backtracking. In such an implementation, it seems natural, as a heuristic, to give a specification step priority over a corresponding construction step. In fact, if one constrains STRUCT, at the beginning of the parse, to have a verb[*finite*] HEAD value (i.e. to be a *finite* verb projection), the above strategy would enable the parser to come up with the correct analysis/analyses with the least possible number of transitions in most cases. Note that the significance of constraining STRUCT as a verb[*finite*] projection at the start is to block the success of any calls for any of the phrase *specification* routines attempting to constrain STRUCT anything other than a verb[*finite*] projection (by attaching any of the words in the input string or any phrase containing any such word – projection or otherwise – as an immediate daughter of STRUCT). Consider, for example, (7.1), with a sentence-initial NP complement.

(7.1) Küçük bir çocuk kitab-ı oku-yor-du.
little a child book-ACC read-PROG-PAST
'A little child was reading the book.'

Assuming that STRUCT is *not* constrained to have a verb[*finite*] HEAD value at the start, after processing the first word *'küçük'*, the parser comes up with the parse state in (7.2), in which STRUCT is constrained as an NP.

(7.2) STRUCT PRED-PHRASES

$$\begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \begin{bmatrix}\text{HEAD} & \boxed{3}\,\textit{noun}\\ \text{SPR} & \langle\rangle\end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{küçük}\rangle \\ \text{SYNSEM} & \begin{bmatrix}\text{MOD} & \boxed{1}\end{bmatrix} \\ \text{HD-DTR} & \textit{word}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \boxed{2}\begin{bmatrix}\textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{1}\begin{bmatrix}\text{HEAD} & \boxed{3}\\ \text{SPR} & \langle\rangle\end{bmatrix}\end{bmatrix}\end{bmatrix} \qquad < \boxed{2} >$$

After that the parse proceeds smoothly until (and including) the processing of the third word *'çocuk'*, resulting in the parse state in (7.3), where the stack is left empty.

(7.3) STRUCT PRED-PHRASES

$$\begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \begin{bmatrix}\text{HEAD} & \boxed{3}\,\textit{noun}[\textit{nom}]\\ \text{SPR} & \langle\rangle\end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{küçük}\rangle \\ \text{SYNSEM} & \begin{bmatrix}\text{MOD} & \boxed{1}\end{bmatrix} \\ \text{HD-DTR} & \textit{word}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \begin{bmatrix}\textit{hd-spr-ph} & \\ \text{SYNSEM} & \boxed{1}\begin{bmatrix}\text{HEAD} & \boxed{3}\\ \text{SPR} & \langle\rangle\end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{bir}\rangle \\ \text{SYNSEM} & \boxed{4}\begin{bmatrix}\text{SPEC} & \boxed{5}\end{bmatrix} \\ \text{HD-DTR} & \textit{word}\end{bmatrix}\right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle\textit{çocuk}\rangle \\ \text{SYNSEM} & \boxed{5}\begin{bmatrix}\text{HEAD} & \boxed{3}\\ \text{SPR} & \langle\boxed{4}\rangle\end{bmatrix} \\ \text{HD-DTR} & \textit{word}\end{bmatrix}\end{bmatrix}\end{bmatrix} \qquad <>$$

Then, on encountering the next word *'kitabı'*, with nowhere to attach it (since the stack is now empty), the parser is forced to backtrack. On the other hand, if STRUCT is constrained to have a verb[*finite*] HEAD value at the beginning of the parse, the first word *'küçük'* in (7.1) leads the parser to the parse state in (7.4) (as a first choice, since the head-adjunct phrase specification step that has led the parser to the parse state in (7.2) above fails in this case), and the correct parse is provided without any backtracking (that is, without any backtracking of a *successful* specification/construction step).

(7.4) STRUCT PRED-PHRASES

$$\boxed{3}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \textit{verb}[\textit{fin}] \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-adjunct-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3}\,\textit{noun} \\ \text{SPR} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{küçük} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{MOD} & \boxed{1} \end{bmatrix} \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \boxed{2}\begin{bmatrix} \textit{hd-ph-pr} & \\ \text{SYNSEM} & \boxed{1}\begin{bmatrix} \text{HEAD} & \boxed{3} \\ \text{SPR} & \langle\rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \textit{sign} \end{bmatrix} \qquad < \boxed{2}, \boxed{3} >$$

Thus, it seems that although constraining STRUCT to be a verb[*finite*] projection at the start seriously restricts the generality of the parser (making it appropriate to parse *finite* verb projections only), it may nevertheless prove to be an effective way of improving the processing efficiency (as far as providing the correct analysis/analyses is concerned). Even in the case of a sentence-initial S[*finite*] complement, the parser wouldn't be faced with a real complexity deficit. Consider, for example, (7.5), starting with a S[*finite*] complement.

(7.5) [$\_{i}$ [$_{S[fin]}$ Kitab-ı Berfu oku-du] san-ıyor-du-m] ben$_i$.
book-ACC Berfu read-PAST think-PROG-PAST-1SG I
'I was thinking that BERFU had read the book.'

On encountering the embedded *finite* verb *'okudu'*, the parser first attaches it as the head daughter of STRUCT by a call for the specification routine for head-complement phrases, constraining STRUCT as *hd-comp-ph* and popping it off the stack, as shown in (7.6).

(7.6) STRUCT PRED-PHRASES

$$\begin{bmatrix} \textit{hd-comp-ph} & \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3}\,\textit{verb}[\textit{fin}] \\ \text{COMPS} & \langle\rangle \end{bmatrix} \\ \text{NON-HD-DTRS} & \left\langle \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{Berfu} \rangle \\ \text{SYNSEM} & \boxed{1}\,\text{NP}[\textit{nom}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix}, \begin{bmatrix} \textit{hd-comp-ph} & \\ \text{PHON} & \langle \textit{kitabı} \rangle \\ \text{SYNSEM} & \boxed{2}\,\text{NP}[\textit{acc}] \\ \text{HD-DTR} & \textit{word} \end{bmatrix} \right\rangle \\ \text{HD-DTR} & \begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{okudu} \rangle \\ \text{SYNSEM} & \begin{bmatrix} \text{HEAD} & \boxed{3}\,\textit{verb}[\textit{fin}] \\ \text{COMPS} & \langle \boxed{1}, \boxed{2} \rangle \end{bmatrix} \end{bmatrix} \end{bmatrix} \qquad <>$$

Again, the next word *'sanıyordum'* (with no place to be attached to) forces the parser to backtrack to the point of processing the previous word *'okudu'*. The parser then tries the head-complement phrase construction routine while attaching that word, which gives rise to the parse state in (7.7). Notice that *'okudu'* is now attached as the head daughter of a newly constructed head-complement phrase, and the two previously encountered NPs are also lowered as non-head daughters of that head-complement phrase.

(7.7)

STRUCT

```
[4] [ hd-ph-pr
      SYNSEM        [ HEAD   [3] verb[fin]
                      COMPS  ⟨⟩ ]
      NON-HD-DTRS   [ hd-comp-ph
                      SYNSEM       S[fin]
                      NON-HD-DTRS  ⟨ [ hd-comp-ph                 [ hd-comp-ph
                                       PHON    ⟨Berfu⟩             PHON    ⟨kitabı⟩
                                       SYNSEM  [1] NP[nom]     ,   SYNSEM  [2] NP[acc]
                                       HD-DTR  word ]              HD-DTR  word ] ⟩
                      HD-DTR       [ word
                                     PHON    ⟨okudu⟩
                                     SYNSEM  [ HEAD   [3]
                                               COMPS  ⟨[1], [2]⟩ ] ] ]
      HD-DTR        sign ]
```

PRED-PHRASES

< [4] >

Let us now turn to the complexity of the routine 'check-CASE' discussed in Chapter 6 (Section 6.5). Note that that routine introduces an additional source of non-determinism in processing, since in the case of any intervening daughters (between the ones that trigger restructuring because of their particular CASE values), the parser is to decide (non-deterministically) where they might actually belong, i.e. the outer clause or the newly constructed embedded one. In (7.8), for instance, the *accusative* and *dative* NPs (intervening between the nominative NP *'ben'* and the *genitive* NP *'Mehmet'in'*, which trigger the restructuring; see Chapter 5, Section 5.6.1, for the relevant reasons) are non-deterministically assigned to either clause at the time of processing *'Mehmet'in'*, giving rise to three different paths: i) lowering both NPs as non-head daughters of the inner clause; ii) lowering only the most recently attached NP, i.e. the dative one; or iii) lowering neither NP. Any of these possibilities may turn out to be true as shown in (7.8a-c), respectively.[3]

[3] Although the *dative* NP *'Güneş'e'* in (7.8c) does belong to an embedded clause, that becomes evident only on encountering the second (dative-marked) *fact* verb *'söylediğine'*.

(7.8) a. Ben [Berfu'-yu Ankara'-ya Mehmet'-in götür-düğ-ü-nü]
I Berfu-ACC Ankara-DAT Mehmet-GEN take-FACT-3sPoss-ACC
duy-du-m.
hear-PAST-1SG
'I have heard that Mehmet has taken Berfu to Ankara.'

b. Ben Berfu'-yu [Ankara'-ya Mehmet'-in git-tiğ-i-ne]
I Berfu-ACC Ankara-DAT Mehmet-GEN go-FACT-3sPoss-DAT
inan-dır-dı-m.
believe-CAUS-PAST-1SG
'I have convinced Berfu that Mehmet has left.'

c. Ben Berfu'-yu [Güneş'-e [Mehmet'-in uyu-duğ-u-nu]
I Berfu-ACC Güneş-DAT Mehmet-GEN sleep-FACT-3sPoss-ACC
Yasemin'-in söyle-diğ-i-ne] inan-dır-dı-m.
Yasemin-GEN tell-FACT-3sPoss-DAT believe-CAUS-PAST-1SG
'I have convinced Berfu that Yasemin has told Güneş that Mehmet was asleep.'

Clearly, one must specify an order in which the different possibilities are to be pursued. To that end, one may opt for an order that is psychologically most plausible (which can only be provided by psycholinguistic evidence), or one that is shown to be statistically most likely, depending on the kind of application the parser is to be used for.

For clarification, I must note that incremental recovery of filler-gap relations in long-distance topicalization does *not* introduce any additional source of non-determinism in the parsing process (that is, other than the one that would be there if the strategy were replaced by a conventional head-topic phrase routine that would be called only after the head daughter of a head-topic phrase was fully encountered). However, in the case of ambiguous filler-gap relations, it actually introduces the non-determinism *earlier* in the process, because of the very nature of the incremental recovery of filler-gap dependencies it has been devised for. For example, in (7.9) (first introduced in Chapter 5, Section 5.7) the ambiguity first arises when the inner *fact* verb *'gördüğünü'* is being processed, in an attempt to fill in the "missing" *genitive* subject of this verb with either of the two *genitive* NPs already encountered. In the absence of any incremental recovery of filler-gap relations, however, the ambiguity would be hidden until the finite verb *'söyledim'* was encountered, that is, until the head daughter of the (inner/outer) head-topic phrase was fully encountered.

(7.9) [Adam-ın$_{i/j}$ [kadın-ın$_{j/i}$ [çocuğ-a [[kitab-ı gör-düğ-ü-nü$_j$]
man-GEN woman-GEN child-DAT book-ACC see-FACT-3sPoss-ACC
bil-diğ-i-ni$_i$] ben söyle-di-m.]]]
know-FACT-3sPoss-ACC I tell-PAST-1SG

'I have told the child that the man/woman knew that the woman/man has seen the book.'

Likewise, in (7.10) (which is again first introduced in Chapter 5, Section 5.7) the ambiguity would only arise while processing the *genitive* NP *'çocuğun'*, as a question of where to attach that NP, rather than the first *fact* verb *'gördüğünü'*.

(7.10) a. [Adam-ın$_i$ [ben san-a [[_$_i$ kadın-ı gör-düğ-ü-nü] çocuğ-un
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC child-GEN
bil-diğ-i-ni] söyle-di-m.]]
know-FACT-3sPoss-ACC tell-PAST-1SG
'As for the man, I have told you that THE CHILD knew that he has seen the woman.'

b. [Adam-ın$_i$ [ben san-a [_$_i$ [[_$_j$ kadın-ı gör-düğ-ü-nü]
man-GEN I you-DAT woman-ACC see-FACT-3sPoss-ACC
çocuğ-un$_j$] bil-diğ-i-ni] söyle-di-m.]]
child-GEN know-FACT-3sPoss-ACC tell-PAST-1SG
'As for the man, I have told you that he knew that she, the child, has seen the woman.'

In Chapter 3 (Section 3.4.1, fn. 8), I mention a computational motivation for considering phrases with locally backgrounded constituents in Turkish as instances of *hd-fill-ph*, rather than instances of *hd-comp-ph*. As mentioned then, the motivation relates to a considerable improvement in parsing efficiency in terms of incremental parsing, as a result of the elimination of spurious partial solutions (or local ambiguity, cf. fn. 1) that would otherwise arise. Let me clarify the point using a complex Turkish sentence such as (7.11).

(7.11) Güneş-e [Berfu'nun Mehmet'i gör-düğ-ü-nü] ben söyle-di-m.
Güneş-DAT Berfu-GEN Mehmet-ACC see-FACT-3sPoss-ACC I tell-PAST-1SG
'I have told Güneş that Berfu has seen Mehmet.'

If head-complement phrases are not constrained to be head-final (with the exception of S[*marked*] complements), then, on encountering the *fact* verb *'gördüğünü'* (which subcategorizes for an NP[*gen*] and an NP[*acc*] complement), the parser is free to (non-deterministically) take any one of the three steps illustrated in (7.12a-c), by lowering:

i) both the NP[*gen*], *'Berfu'nun'*, and the NP[*acc*], *'Mehmet'i'*, (without predicting any further complements),

ii) only the NP[*acc*], *'Mehmet'i'*, (predicting a yet-to-come NP[*gen*] complement); or

iii) neither of those two NPs (predicting both complements as being yet-to-come)

as complements of the newly constructed head-complement phrase headed by *'gördüğünü'*. Clearly, it is only the first alternative that leads the parser to the correct parse.

(7.12)	a.	Güneş-e	[Berfu'nun	Mehmet'i	gör-düğ-ü-nü]	...
		Güneş-DAT	Berfu-GEN	Mehmet-ACC	see-FACT-3sPoss-ACC	
	b.	Güneş-e	Berfu'nun	[Mehmet'i	gör-düğ-ü-nü	...
		Güneş-DAT	Berfu-GEN	Mehmet-ACC	see-FACT-3sPoss-ACC	
	c.	Güneş-e	Berfu'nun	Mehmet'i	[gör-düğ-ü-nü	...
		Güneş-DAT	Berfu-GEN	Mehmet-ACC	see-FACT-3sPoss-ACC	

Milward (1995) proposes the use of statistical methods in incremental parsing of lexicalized grammars, for language tuning purposes. He suggests that the parser could be run over corpora to estimate the probabilities of particular transitions in the case of particular words. It may merit further consideration to investigate the use of similar methods in the present parsing approach.[4] Note that one could in that case restrict attention to highly local information such as the constraints on the lexical entry of the new word, and the ones on the active phrase (rather than the entire global structure). In addition, one could be concerned with only the CATEGORY values to further reduce the complexity of such a method, since that is where the syntactic information concerning a sign is encoded. Furthermore, relying on statistical methods, one could also prune the search space by cutting off certain paths after a number of consecutive construction steps, to improve the efficiency of the overall parsing process (although in theory this would be a clear concession from completeness).

7.2 Psychological Plausibility

As stated before in Chapter 5 (again Section 5.3), in order to guarantee both completeness and termination, any incremental processing mechanism for HPSG grammars

[4]See Briscoe and Carroll (1993) for association of unification grammars with stochastic LR parsing of natural language, and Brew (1995) for the association of probabilities with typed feature structures of the form used in HPSG.

should be allowed to commit itself to decisions, concerning the attachment of newly encountered constituents, which it may nonmonotonically revise at subsequent stages of processing, if need arises. As long as the underlying constraint satisfaction operation is unification, the only way of achieving that – without violating the soundness of the approach – seems to be non-deterministically leaving certain sub-structures of the global structure underspecified, whose daughters could later be nonmonotonically lowered in the structure. One obvious drawback of this approach is that although the constituent structure is incrementally constructed, the structure as a whole (i.e. all feature values) may not always be incrementally constructed, since no constraints can be imposed on the underspecified phrases in the structure that relate any feature value within their SYNSEM value to any of the daughter values of the phrase. Another drawback is that the non-determinism involved in processing certain cases – say an English N′ with two post-modifier PPs, such as (7.13) (originally from Chapter 5), which is so easily processible by humans – may be difficult to justify from a psycholinguistic viewpoint.

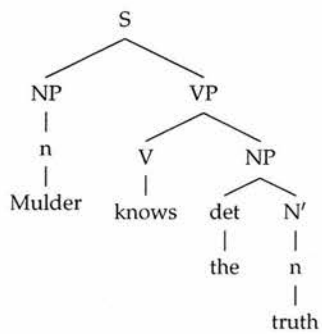
(7.13) $[_{NP}$ $[_{DetP}$ the $]$ $[_{N'}$ $[_{N'}$ $[_{N'}$ policeman $]$ $[_{PP}$ with glasses $]]$ $[_{PP}$ in uniform $]]]$

In Chapter 5 (Section 5.3), I briefly mentioned the SOUL system of Konieczny (1996), which takes HPSG as the competence base. There I outlined a nonmonotonic operation called 'adjoining' proposed by Konieczny to deal with head-adjunct structures with post-modifiers, such as (7.13), which substantially decreases the degree of the non-determinism employed by the processing mechanism of the SOUL system (in comparison to the approach presented here), but nevertheless violates the soundness of the overall approach. Another nonmonotonic operation proposed by Konieczny is 'lowering', used to deal with, for example, the processing of 'that'-less clauses in English, such as (7.14).

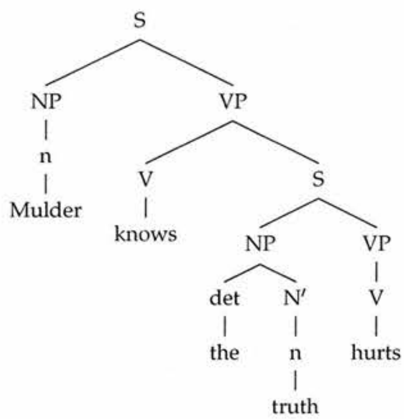
(7.14) Mulder knows the truth hurts.

The operation basically works as follows: when the NP *'the truth'* in (7.14) is first processed, it is attached as a complement of the verb *'knows'*, as shown in (7.15), but is later lowered as the subject daughter of the S complement clause, as shown in (7.16), once the finite verb *'hurts'* is processed. Notice that the NP complement of *'knows'* in (7.15) is *replaced* (overriden) by the S complement clause headed by *'hurts'* in (7.16).

(7.15)

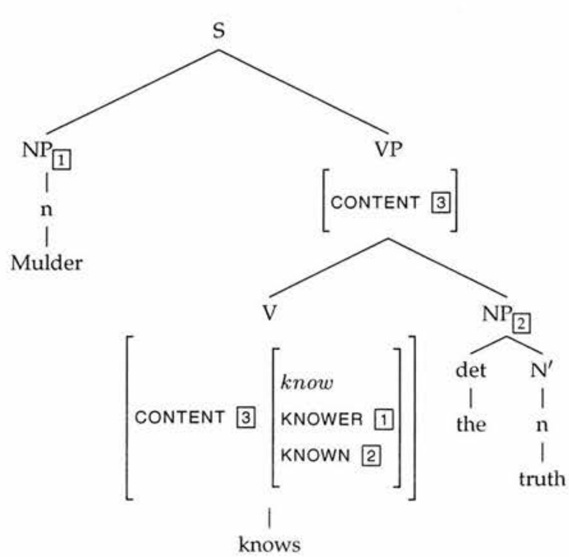


(7.16)



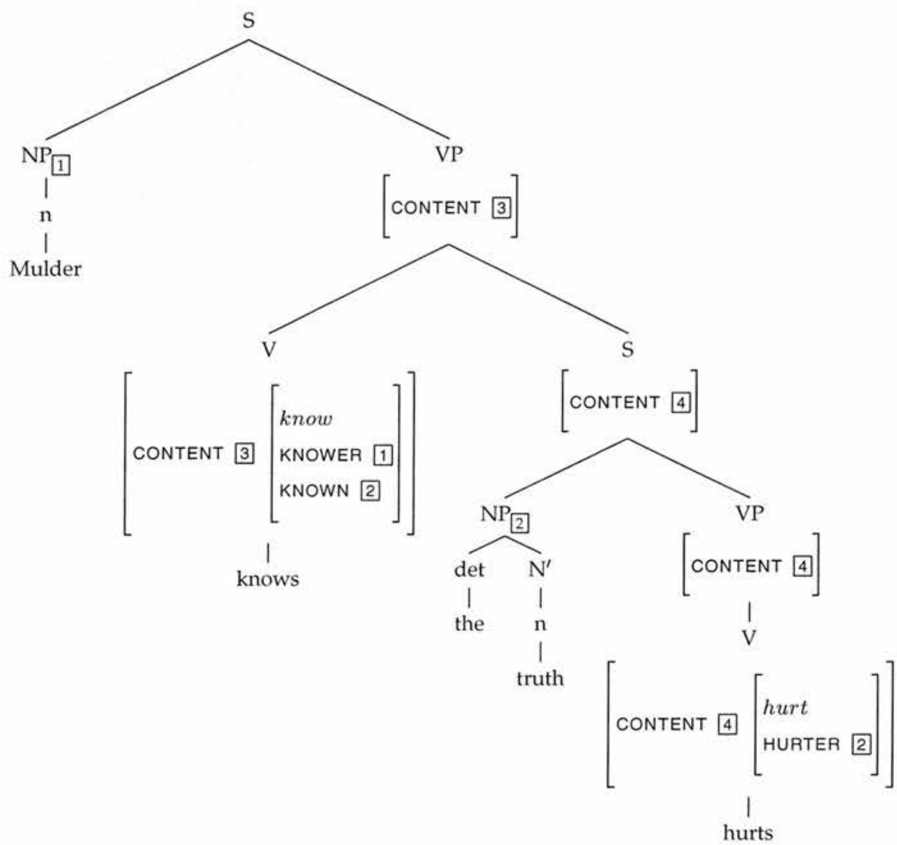
There are two points to note about this operation. First, just like adjoining, it violates the soundness of the approach. To clarify this point, let us consider, for example, the CONTENT values of certain nodes in the structures before and after lowering takes place. The former structure is repeated in (7.17), which illustrates the structure-sharings between the INDEX values of the subject NP *'Mulder'* and the object NP *'the truth'*, and the respective semantic roles within the verb's CONTENT value.

(7.17)



In the structure after lowering, repeated below in (7.18), those constraints are still in effect, which in turn makes *'knows'* assign a semantic role to the subject of its complement S, rather than the complement itself.

(7.18)



The second (and perhaps more important) point concerning lowering is that that operation would in fact never succeed. The reason is simply that when the NP *'the truth'* is first attached to the structure in (7.15), its CASE value is constrained as *accusative* (since the NP complement of *'knows'* is constrained to have *accusative* CASE value). Thus, the same node couldn't later be lowered as the subject of the S complement in (7.16), since such a move would *unsuccessfully* try to unify the *accusative* CASE value of that node with the *nominative* CASE value specified for the subject of the *finite* verb *'hurts'*.

In short, such nonmonotonic operations on fully specified objects cannot be exploited by a processing mechanism relying on unification (at least not, if one is committed to providing sound structures at the end), unless one finds a way of reversing the effects of unification at earlier stages of processing, if need arises. The question of how this could be done (if indeed ever) is, however, beyond the scope of this dissertation.

7.3 Implementation

In this section, I discuss certain aspects of a parsing system for Turkish implemented in the LIFE programming language (Aït-Kaci and Lincoln (1988)) using the parsing algorithm presented in Chapter 6. LIFE provides the programmer with certain features from three different programming paradigms, namely functional programming, logic programming and object-oriented programming; e.g. functions, predicates, a Prolog-like resolution strategy, unification, an inheritance-based sorted feature system, multiple inheritance and constrained sorts (Aït-Kaci *et al.* (1994)).

Part of the parsing system is an implementation of an HPSG grammar for Turkish, that is, the HPSG grammar presented in Chapter 2 (Section 2.3) together with the proposals/modifications introduced for Turkish in Chapters 3 and 4. Note that the grammar merely consists of a type (sort) hierarchy with certain constraints imposed on certain types. The lexicon is implemented as a set of (independent) *word* objects with certain constraints imposed on them (according to the particular lexical entry they stand for). In addition, lexical rules (most of which mainly deal with the agglutinative morphology of Turkish, but some of which actually constitute the basis of the treatments proposed in this dissertation for certain phenomena such as extraction and relativization in Turkish; cf. Chapters 3 and 4, respectively) are implemented as LIFE predicates that relate certain input *word* objects to certain output *word* objects, which are then included in the lexicon. A more advanced implementation of the grammar should clearly pursue a more sophisticated lexicon design, with a multiple-inheritance architecture, to eliminate the redundancy in the lexicon by capturing generalizations, due to properties shared by a class of lexical entries (see, for example, Pollard and Sag (1987), Flickinger (1987), Fraser and Hudson (1992), and Flickinger and Nerbonne (1992)). Moreover, van der Linden (1992) points out that, in addition to eliminating the lexical redundancy, hierarchical lexicon design may also be exploited to improve the efficiency of language processing, and proposes certain techniques along these lines, incorporated in an incremental processing mechanism for Categorial Grammar. It may merit further consideration to investigate the applicability/adaptability of those techniques in the framework of HPSG.

Recall that the grammar in Chapter 2 (Section 2.3) makes use of a number of default constraints on the type *hd-ph*, which specify certain default feature values and reentrancies that are assumed to be overriden in case of a conflicting non-default value specification in a more specific subtype of *hd-ph*. As noted then, the use of such default constraints follows the framework for default unification outlined in Lascarides *et al.* (1996). In the actual implementation of the grammar in LIFE (which only supports ordinary unification), each one of those default constraints is rather imposed as an ordinary (hard) constraint on the most specific phrase types that actually satisfy the given default constraint. To capture generalizations, in each case I introduce a new phrase type that is specified to be a supertype of the most specific phrase types that actually satisfy the default constraint in question, and impose the respective ordinary constraint on that new phrase type. Consider, as an example, the Empty COMPS Constraint (ECC), repeated below in (7.19).

(7.19) Empty COMPS Constraint (ECC):
hd-ph ⇒ [HD-DTR [COMPS / <>]]

Recall from Table 2.1 in Chapter 2 (Section 2.3) that the ECC is assumed to be overriden in objects of type *hd-comp-ph*. In the implementation, the ECC is therefore redefined as an ordinary constraint imposed on a new type called *ecc-ph*, as shown in (7.20), where *ecc-ph* is specified to be a supertype of the types *hd-adjunct-ph*, *hd-spr-ph*, *hd-subj-ph*, *hd-mark-ph*, and *hd-fill-ph* (i.e. the phrase types that actually satisfy the defeasible constraint in the original (default) ECC in (7.19)).

(7.20) *ecc-ph* ⇒ [HD-DTR [COMPS <>]]

Similarly, the Marking Principle is implemented as an ordinary constraint on a new type called *mark-pr-ph*, a supertype of the phrase types *hd-adjunct-ph*, *hd-spr-ph*, *hd-subj-ph*, *hd-comp-ph*, and *hd-fill-ph*. The Valence Principle, on the other hand, is implemented on each of the most specific phrase types in the appropriate way.

One point to note is that the processing mechanism of the parsing system relies totally on LIFE's depth-first Prolog-like resolution strategy. There is no additional copying of data structures at non-deterministic choice points, other than that done by LIFE's own backtracking mechanism.

Among the constraints on phrase types introduced in Chapter 2 (Section 2.3) the Nonlocal Feature Principle (NFP) is rather distinct from the rest in one important respect: it is formulated in terms of union and difference operations on set objects (rather than straightforward structure-sharing of values).[5] It is therefore essential that satisfiability of the NFP on a phrase object is checked only after all of its daughters have already been

[5] In fact, the latest version of the Semantics Principle of Pollard and Sag (1994) (cf. fn. 4 in Chapter 2, page 19) also relies on a set union operation, and is again subject to the same restriction mentioned below.

encountered. This constitutes a problem for the incremental processing mechanism of the parser, since certain phrase objects during processing are specified/constructed before some of their daughters are actually encountered (which are then predicted to come, by being pushed onto the stack). Therefore, in the implementation, the NFP is not imposed as a constraint on type *hd-ph* in the type hierarchy (i.e. the grammar), but is rather imposed on every phrase object during processing only after (and as soon as) *all* its daughters have been *fully* processed. To that end, I have implemented the stack PRED-PHRASES as a stack of a new data type called *entry*, whose structure is given below:

$$(7.21) \quad \begin{bmatrix} \textit{entry} & \\ \text{NODE} & \textit{hd-ph-pr} \\ \text{MOTHER} & \textit{none-entry} \end{bmatrix}$$

As shown in (7.21), an *entry* object has two appropriate fields, namely NODE and MOTHER. The NODE field is of type *hd-ph-pr*, and actually corresponds to a predicted phrase in the original algorithm. The MOTHER field is of type *none-entry*, with the subtypes *none* and *entry*. The idea is outlined below:

1. The only entry on the stack at the beginning of the parse has a NODE field pointing to the global structure STRUCT, and a MOTHER field set to be *none*.

2. Whenever a daughter is predicted by a phrase *specification* routine, a new *entry* object is constructed, whose NODE field points to the predicted daughter, and whose MOTHER field points to the *entry* object that was on the top of the stack when the specification routine in question was first called (i.e. the *entry* object whose NODE field points to the mother of that predicted daughter).

3. Whenever a daughter is predicted by a phrase *construction* routine, a new *entry* object is constructed whose NODE field is again a pointer to the predicted daughter, but (this time) whose MOTHER field is set to another newly constructed *entry* object, with the NODE field pointing to the mother of the predicted daughter (i.e. the *phrase* object newly constructed by the construction routine in question), and MOTHER field simply set to be *none*.

4. Whenever a phrase *specification* routine succeeds, and does *not* predict any new daughters (to be pushed onto the stack), a *recursive* process is initiated that imposes the NFP on the just (further) constrained active phrase itself, and on any of its ancestors until it reaches a MOTHER field that is set to be *none*.

In fact, LIFE itself provides a *residuation* mechanism in *function* calls which suspends the execution of a function if any of its arguments is not yet instantiated in the way specified in the function head.[6] Use of that mechanism could enable one to impose the NFP on type *hd-ph* in the grammar in the usual way, rather than being concerned with the details of the processing mechanism as discussed above, which would obviously be more in line with the declarative nature of HPSG. However, functions in LIFE are *deterministic*, as opposed to predicates that are *non-deterministic* (just like those in Prolog). The NFP (being defined in terms of operations on sets) should therefore be implemented using predicates (rather than functions) to guarantee that all possible analyses would be returned in cases of ambiguity. Note also that residuation is a computationally complex and expensive operation, and implementing a pseudo-residuation mechanism such as the one discussed above may actually turn out to improve the processing efficiency (with the obvious price of losing declarativeness).

7.4 Summary

In this chapter, I have been concerned with certain computational, psycholinguistic and implementational issues concerning the incremental parsing approach presented in Chapters 5 and 6. In Section 7.1, I focussed on the issue of computational complexity, and proposed ways of improving the efficiency of a parser implemented using this approach. In particular, I suggested certain strategies that would enable a serial parser for a head-final language to come up with the correct analysis/analyses with the least possible number of transitions in most cases (and with minimal reanalysis in the remaining cases). In Section 7.2, I discussed the impact of underspecification in structure and the high degree of non-determinism in processing, on the psychological plausibility of the present approach. Finally, in Section 7.3, I briefly discussed certain implementational aspects of a parsing system implemented for Turkish in the LIFE programming language using the parsing algorithm presented in Chapter 6.

[6] A residuated function call returns a temporary result, which is later unified with the actual result once the function fires (after all its arguments are properly refined). Thus, the temporary result may be used in further function calls as if it were the actual result (Aït-Kaci *et al.* (1994)).

Chapter 8

Conclusions

In this dissertation, I have presented an incremental parsing algorithm for HPSG grammars. I have specifically focussed on incremental parsing of Turkish, a "free" word order, head-final language, and explored a number of points to improve the incrementality of structuring while processing such a language using a constraint-based grammar.

To that end, the first part of the dissertation has been concerned with theoretical accounts of several distinguishing phenomena in Turkish grammar, within the framework of HPSG. In Chapter 3, I examined the issue of word order variation in Turkish, focussing mainly on the syntactic aspects of that variation. I presented a detailed characterization of the word order variation in simple and complex Turkish sentences, which may involve both local and long-distance scrambling. I outlined the assumptions I make, in HPSG, to deal with the "free" word order in Turkish, and proposed ways of capturing certain restrictions on local and long-distance scrambling, either in the form of LP constraints, in the usual way, or as constraints imposed on certain phrase types in the grammar. In Chapter 4, I examined the issue of relativization in Turkish, and proposed a lexical account of the phenomenon within HPSG. The account exploits the identifying morphology on verbal heads of Turkish relative clauses, and is based upon an assumption that those clauses have lexically specified MOD values (encoded in the lexical entry of the verbal head of the clause).

Then in the second part of the dissertation, I presented the incremental parsing algorithm I have developed for HPSG grammars, which parses an input string from left to right, attaching every word of the input to a global structure as soon as it is encountered, thereby dynamically changing the structure as the parse progresses. In the process of attaching a word to the structure, the parser makes systematic use of the selection features of the HPSG formalism (in determining the type(s) of the phrase(s) that the word in question may be a part of), as well as taking into account the linear precedence constraints employed by the language being parsed. In order to overcome the processing problems due to left recursive structures in a language, certain sub-structures of the structure are non-deterministically left underspecified during processing, whose daughters may be nonmonotonically lowered in the structure at subsequent stages of processing, if need arises. Despite a high degree of non-determinism in processing in the *worst* case, I proposed certain strategies that would enable a serial parser for a head-final language (like Turkish) to provide the correct analysis/analyses with the least possible number of processing steps in most cases (and with minimal reanalysis in the remaining cases).

As mentioned above, I further explored certain issues that arise in incremental processing of a "free" word order, head-final language such as Turkish. In terms of parsing language word by word, head-final languages, in general, compensate for the late arrival of the head information by providing the parser with other means, such as CASE values, to construct the structure incrementally. I examined how the parser can exploit certain restrictions on the co-occurrence of constituents with particular CASE values in a Turkish clause, to foresee the existence of an embedded phrase/clause while processing embedded constructions in Turkish. I also presented an incremental view of long-distance topicalization in Turkish, which enables the parser to incrementally recover the long-distance dependencies while processing such structures, as soon as encountering a lexical head with a "missing" argument. To that end, the processing of a newly encountered word with a non-empty INHER|SLASH value triggers a search in the structure for each of the "missing" arguments of the new word, which results in restructuring of the global structure in case of a successful search. I further pointed out that such incremental recovery of long-distance dependencies further enables one to capture certain (strong) preferences that humans exhibit in processing certain cases with potentially ambiguous long-distance dependency relations.

In Chapter 7, I discussed the impact of underspecification in structure and the high degree of non-determinism in processing, on the psychological plausibility of the present approach. I argued that as long as the underlying constraint satisfaction operation is unification, the only way of employing nonmonotonicity in a parsing approach for HPSG – without violating its soundness – is to exploit underspecification in structure and a certain degree of non-determinism in processing, as is the case for the approach presented in this dissertation. I therefore suggest that efficient incremental satisfiability of HPSG in a model of human language processing is most likely to be realized by adopting some constraint satisfaction operation more complex than unification. The issue is left wide open for further consideration.

Bibliography

Abney, S. P. (1987). Licensing and parsing. In *Proceedings of NELS 17*, pages 1–15, Amherst. University of Massachusetts.

Abney, S. P. (1989). A computational model of human parsing. *Journal of Psycholinguistic Research*, **18**, 129–144.

Ades, A. and Steedman, M. J. (1982). On the order of words. *Linguistics and Philosophy*, **4**, 517–558.

Aït-Kaci, H. and Lincoln, P. (1988). Life: A natural language for natural language. Technical Report ACA-ST-074-88, Systems Technology Laboratory, Austin, Tex.

Aït-Kaci, H., Dumant, B., Meyer, R., and Roy, P. V. (1994). *The Wild LIFE Handbook*. Digital-Paris Research Laboratory, prepublication edition edition.

Barker, C., Hankamer, J., and Moore, J. (1990). Wa and ga in Turkish. In D. Katarzyna, P. Farrell, and E. M. Bikandi, editors, *Grammatical Relations*, CSLI Lecture Notes, pages 21–43. Center for the Study of Language and Information, Stanford, Ca.

Berwick, R. C. and Weinberg, A. S. (1984). *The Grammatical Basis of Linguistic Performance: Language Use and Acquisition*. Current Studies in Linguistics. MIT Press, Cambridge, Mass.

Borsley, R. (1993). Heads in Head-driven Phrase Structure Grammar. In G. Corbett, N. Fraser, and S. McGlashan, editors, *Heads in Grammatical Theory*, pages 186–203. Cambridge University Press, Cambridge.

Bouma, G. (1992). Feature structures and nonmonotonicity. *Computational Linguistics*, **18**(2), 183–203.

Brew, C. (1995). Stochastic HPSG. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, pages 83–89, Dublin, Ireland. University College.

Briscoe, T. and Carroll, J. (1993). Generalised probabilistic LR parsing of natural language (corpora) with unification grammars. *Computational Linguistics*, **19**(1), 25–60.

Carpenter, B. (1993). Skeptical and credulous default unification with applications to templates and inheritance. In T. Briscoe, A. Copestake, and V. de Paiva, editors, *Inheritance, Defaults, and the Lexicon*, pages 13–37. Cambridge University Press, Cambridge.

Chomsky, N. (1981). *Lectures on Government and Binding*. Foris Publications, Dordrecht.

Cooper, R. (1975). *Montague's Semantic Theory and Transformational Syntax*. Ph.D. thesis, University of Amherst, Mass.

Cooper, R. (1983). *Quantification and Syntactic Theory*. Number 21 in Synthese Language Library. D. Reidel, Dordrecht.

Cooper, R. (1986). Swedish and the Head-Feature Convention. In L. Hellan and K. K. Christensen, editors, *Topics in Scandinavian Syntax*, pages 31–52. D. Reidel, Dordrecht.

Copestake, A. (1993). Defaults in lexical representation. In T. Briscoe, V. de Paiva, and A. Copestake, editors, *Inheritence, Defaults, and the Lexicon*, Studies in Natural Language Processing, pages 223–245. Cambridge University Press, Cambridge.

Crocker, M. W. (1996a). *Computational Psycholinguistics: An Interdisciplinary Approach to the Study of Language*. Kluwer Academic Publishers, Dordrecht, The Netherlands.

Crocker, M. W. (1996b). Mechanisms for sentence processing. Research Paper EUCCS/RP-70, University of Edinburgh.

Csató, E. (1985). A syntactic analysis of participle constructions in modern Turkish. In *Beşinci Milletlerarası Türkoloji Kongresi*, pages 39–56, İstanbul.

Dede, M. (1978). Why should Turkish relativization distinguish between subject and non-subject head nouns? In *Proceedings of the Fourth Annual Meeting of the Berkeley Linguistics Society*, Berkeley, Ca.

Dede, M. (1986). Definiteness and referentiality in Turkish verbal sentences. In D. I. Slobin and K. Zimmer, editors, *Studies in Turkish Linguistics*, volume 8 of *Typological Studies in Language*, pages 147–164. John Benjamins Publishing Co, Amsterdam.

Diesing, M. (1990). *The syntax of roots of semantic partition*. Ph.D. thesis, University of Massachussetts, Amherst, Mass.

Dini, L. (to appear). Null complements in Italian. In S. Balari and L. Dini, editors, *Romance in HPSG*, CSLI Lecture Notes, pages 215–260. Center for the Study of Language and Information, Stanford, Ca.

Dörre, J., Eisele, A., Wedekind, J., Calder, J., and Reape, M. (1990). A survey of linguistically motivated extensions to unification-based formalisms. In J. Wedekind, editor, *A Survey of Linguistically Motivated Extensions to Unification-Based Formalisms*, pages 1–43. Centre for Cognitive Science, University of Edinburgh. DYANA Report R3.1.A.

Enç, M. (1991). The semantics of specificity. *Linguistic Inquiry*, **22**, 1–25.

Erguvanlı, E. E. (1984). *The Function of Word Order in Turkish Grammar*. University of California Publications in Linguistics. University of California Press, Berkeley and Los Angeles. UCLA PhD dissertation 1979.

Erguvanlı, E. E. (1987). The role of semantic features in Turkish word order. *Folia Linguistica*, **21**, 215–227.

Flickinger, D. (1987). *Lexical Rules in the Hierarchical Lexicon*. Ph.D. thesis, Stanford University, Stanford.

Flickinger, D. and Nerbonne, J. (1992). Inheritence and complementation: A case study of *easy* adjectives and related nouns. *Computational Linguistics*, **18**(3), 269–309.

Fodor, J. D. (1978). Parsing strategies and constraints on transformations. *Linguistic Inquiry*, **9**, 427–473.

Fraser, N. and Hudson, R. A. (1992). Inheritance in Word Grammar. *Computational Linguistics*, **18**(2), 133–158.

Frazier, L. (1984). Modularity and the representational hypothesis. In *Proceedings of NELS 15*, pages 131–144, Providence, Rhode Island. Brown University.

Gazdar, G. and Pullum, G. K. (1981). Subcategorization, constituent order and the notion 'head'. In M. Moortgat, H. van der Hulst, and T. Hoekstra, editors, *The Scope of Lexical Rules*, pages 107–123. Foris Publications, Dordrecht.

Gazdar, G. and Pullum, G. K. (1982). Generalized phrase structure grammar: A theoretical synopsis. Ms., distributed by Indiana University Linguistics Club.

Gazdar, G., Klein, E., Pullum, G., and Sag, I. (1985). *Generalized Phrase Structure Grammar*. Basil Blackwell, London.

Grosz, B. J., Joshi, A. K., and Weinstein, S. (1995). Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, **21**(2), 203–225.

Hankamer, J. and Knecht, L. (1976). The role of subject/non-subject distinction in determining the choice of relative clause participle in Turkish. In Aissen and Hankamer, editors, *Harvard Studies in Syntax and Semantics*, volume 2, pages 197–219. Cambridge.

Heim, I. (1982). *The Semantics of Definite and Indefinite Noun Phrases*. Ph.D. thesis, University of Massachusetts, Amherst, Mass.

Hoffman, B. (1995). *The computational analysis of the syntax and interpretation of "free" word order in Turkish*. Ph.D. thesis, Philadelphia, Pa.

Inoue, A. (1991). *A comparative study of parsing in English and Japanese*. Ph.D. thesis, University of Connecticut.

Kamp, H. (1981). A theory of truth and semantic representation. In J. A. G. Groenendijk, T. M. V. Janssen, and M. B. J. Stokhof, editors, *Formal Methods in the Study of Language: Part 1*, pages 277–322. Mathematisch Centrum, Amsterdam.

Kaplan, R. and Zaenen, A. (1988). Long-distance dependencies, constituent structure, and functional uncertainty. In M. Baltin and A. Kroch, editors, *Alternative Conceptions of Phrase Structure*. Cambridge University Press, Cambridge.

Kaplan, R. M. (1987). Three seductions of computational psycholinguistics. In P. J. Whitelock, M. M. Wood, H. L. Somers, R. Johnson, and P. Bennett, editors, *Linguistic Theory and Computer Applications*, pages 149–188. Academic Press, London.

Kaplan, R. M. and Bresnan, J. (1982). Lexical-Functional Grammar: A formal system for grammatical representation. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*, MIT Press Series on Cognitive Theory and Mental Representation, chapter 4, pages 173–281. MIT Press, Cambridge, Mass.

Karttunen, L. and Kay, M. (1985). Parsing in a free word order language. In D. R. Dowty, L. Karttunen, and A. M. Zwicky, editors, *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, pages 279–306. Cambridge University Press, Cambridge.

Kasper, R. (1994). Adjuncts in the mittelfeld. In J. Nerbonne, K. Netter, and C. Pollard, editors, *German in Head-Driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, Ca.

Kasper, R., Kiefer, B., Netter, K., and Vijay-Shanker, K. (1995). Compilation of HPSG to TAG. In *Proceedings of the 33th Annual Meeting of the Association for Computational Linguistics*, Cambridge, Mass. Massachusetts Institute of Technology.

Kay, M. (1985). Parsing in functional unification grammar. In D. R. Dowty, L. Karttunen, and A. M. Zwicky, editors, *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, pages 251–278. Cambridge University Press, Cambridge.

Kay, M. (1989). Head-driven parsing. In *Proceedings of the International Workshop on Parsing Technology*, pages 52–62, Pittsburgh, Pa. Carnegie Mellon University.

King, T. (1995). *Configuring Topic and Focus in Russian*. CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, Ca.

Knecht, L. (1979). The role of the genitive suffix in relative clauses in Turkish: A reply to Dede. In *Proceedings of the Fifth Annual Meeting of the Berkeley Linguistics Society*, pages 180–197, Berkeley, Ca.

Knecht, L. E. (1986). *Subject and Object in Turkish*. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge.

Konieczny, L. (1996). *Human sentence processing: A semantics-oriented parsing approach*. Ph.D. thesis, University of Freiburg, Freiburg.

Konieczny, L. and Hemforth, B. (1994). Incremental parsing with lexicalized grammars. In G. Strube, editor, *Current Research in Cognitive Science*, volume 1/94, pages 33–54. University of Freiburg, Freiburg.

Konieczny, L. and Strube, G. (1995). SOUL: A cognitive parser. In J. D. Moore and J. F. Lehman, editors, *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*, pages 631–636. Distributed by Lawrence Erlbaum Associates, Hillsdale, N.J.

Kornfilt, J. (1984). *Case marking, agreement, and empty categories in Turkish*. Ph.D. thesis, Harvard University, Cambridge, Mass.

Krieger, H.-U. and Nerbonne, J. (1993). Feature-based inheritance networks for computational lexicons. In T. Briscoe, V. de Paiva, and A. Copestake, editors, *Inheritence, Defaults, and the Lexicon*, Studies in Natural Language Processing, pages 90–136. Cambridge University Press, Cambridge.

Kuhn, J. (1996). An underspecified HPSG representation for information structure. In *Proceedings of the 16th International Conference on Computational Linguistics*, pages 670–675, Copenhagen, Denmark.

Lascarides, A., Briscoe, E. J., Asher, N., and Copestake, A. (1996). Order independent and persistent typed default unification. *Linguistics and Philosophy*, **19**, 1–89.

Lee, Y.-S. (1993). *Scrambling as case-driven obligatory movement*. Ph.D. thesis, University of Pennsylvania, Philadelphia, Pa.

Lees, R. (1965). Turkish nominalizations and a problem of ellipsis. *Foundations of Language*, **1**(2), 112–121.

Lewis, D. (1975). Adverbs of quantification. In E. L. Keenan, editor, *Formal Semantics of Natural Language: Papers from a Colloquium Sponsored by King's College Research Centre, Cambridge*, pages 3–15. Cambridge University Press, Cambridge.

Marcus, M. P., Hindle, D., and Fleck, M. (1983). D-theory: Talking about talking trees. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 129–136, Cambridge, Mass. Massachusetts Institute of Technology.

Marslen-Wilson, W. and Tyler, L. K. (1980). The temporal structure of spoken language understanding. *Cognition*, **8**, 1–74.

Marslen-Wilson, W. D. (1973). Linguistic structure and speech shadowing at very short latencies. *Nature*, **244**, 522–523.

Milward, D. (1992). Dynamics, dependency grammar and incremental interpretation. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 1095–1099, Nantes, France.

Milward, D. (1994). Dynamic dependency grammar. *Linguistics and Philosophy*, **17**, 561–605.

Milward, D. (1995). Incremental interpretation of categorial grammar. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, Dublin, Ireland. University College.

Mohanan, K. P. (1982). Grammatical relations and clause structure in Malayalam. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*, MIT Press Series on Cognitive Theory and Mental Representation, chapter 8, pages 504–589. MIT Press, Cambridge, Mass.

Nerbonne, J. (1983). *German temporal semantics: Three dimensional tense logic and a GPSG fragment*. Ph.D. thesis, Ohio State University, Columbus, Ohio.

Nilsson, B. (1985). *Case Marking Semantics in Turkish*. Ph.D. thesis, University of Stockholm.

Oflazer, K. (1994). Two-level description of Turkish morphology. *Literary and Linguistic Computing*, **9**(2).

Pereira, F. C. N. (1985). A structure-sharing representation for unification-based grammar formalisms. In *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics*, pages 137–144. University of Chicago.

Pickering, M. and Barry, G. (1991). Sentence processing without empty categories. *Language and Cognitive Processes*, **6**(3), 229–259.

Pollard, C. (1990). On head non-movement. In *Tilburg Symposium on Discontinuous Constituency*, Tilburg, The Netherlands. Institute for Language Technology and Artificial Intelligence, Tilburg University.

Pollard, C. and Sag, I. A. (1987). *Information-Based Syntax and Semantics: Fundamentals*. Number 13 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, Ca.

Pollard, C. and Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar*. CSLI and University of Chicago Press, Stanford, Ca. and Chicago, Ill.

Pollard, C. and Yoo, E. J. (to appear). A unified theory of scope for quantifiers and *wh*-phrases. *Journal of Linguistics*.

Pritchett, B. L. (1992). *Grammatical Competence and Parsing Performance*. University of Chicago Press, Chicago.

Pullum, G. K. (1991). English nominal gerunds as noun phrases with verb phrase heads. *Linguistics*, **29**, 763–799.

Reape, M. (1990). Getting things in order. In *Tilburg Symposium on Discontinuous Constituency*, Tilburg. Tilburg University.

Reape, M. (1994). Domain union and word order variation in German. In J. Nerbonne, K. Netter, and C. Pollard, editors, *German in Head-Driven Phrase Structure Grammar*, number 46 in CSLI Lecture Notes, pages 151–197. Center for the Study of Language and Information, Stanford, Ca.

Russell, G., Ballim, A., Carroll, J., and Warwick-Armstrong, S. (1993). A practical approach to multiple default inheritance for unification-based lexicons. In T. Briscoe, V. de Paiva, and A. Copestake, editors, *Inheritence, Defaults, and the Lexicon*, Studies in Natural Language Processing, pages 137–147. Cambridge University Press, Cambridge.

Sag, I. (1995). Taking performance seriously. Unpublished manuscript, CSLI, Stanford University.

Sag, I. (to appear). English relative clause constructions. *Journal of Linguistics.*

Sag, I. and Fodor, J. D. (1994). Extraction without traces. In *West Coast Conference in Formal Linguistics 13.*

Schabes, Y. (1990). *Mathematical and computational aspects of lexicalized grammars.* Ph.D. thesis, University of Pennsylvania, Philadelphia, Pa.

Sezer, E. (1986). The unmarked sentential subject constraint in Turkish. In D. Slobin and K. Zimmer, editors, *Studies in Turkish Linguistics*, volume 8 of *Typological Studies in Language*, pages 123–135. John Benjamins Publishing Co, Amsterdam.

Sezer, E. (1991). *Issues in Turkish syntax.* Ph.D. thesis, Harvard University, Cambridge, Mass.

Shieber, S. M. (1984). The design of a computer language for linguistic information. In *Proceedings of the 10th International Conference on Computational Linguistics and the 22nd Annual Meeting of the Association for Computational Linguistics*, pages 362–366, Stanford, Ca. Stanford University.

Shieber, S. M. (1986). *An Introduction to Unification-Based Approaches to Grammar.* Number 4 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, Ca.

Shieber, S. M., Uszkoreit, H., Pereira, F. C. N., Robinson, J. J., and Tyson, M. (1983). The formalism and implementation of PATR-II. In B. Grosz and M. E. Stickel, editors, *Research on Interactive Acquisition and Use of Knowledge*, pages 39–79. Artificial Intelligence Center, SRI International, Menlo Park, Ca.

Strömbäck, L. (1995). User-defined nonmonotonicity in unification-based formalisms. In *Proceedings of the 33th Annual Meeting of the Association for Computational Linguistics*, Cambridge, Mass. Massachusetts Institute of Technology.

Sturt, P. and Crocker, M. W. (1995). Incrementality and monotonicity in syntactic parsing. In D. Milward and P. Sturt, editors, *Edinburgh Working Papers in Cognitive Science: Incremental Interpretation*, number 11, pages 23–66. University of Edinburgh.

Tomita, M. (1986). *Efficient Parsing for Natural Language: A Fast Algorithm for Practical Systems*. Kluwer Academic Publishers, Hingham, Mass.

Tomita, M. (1987). An efficient augmented-context-free parsing algorithm. *Computational Linguistics*, **13**(1-2), 31–46.

Trueswell, J. C. and Tanenhaus, M. K. (1994). Towards a lexicalist framework of constraint-based syntactic ambiguity resolution. In C. Clifton, L. Frazier, and K. Rayner, editors, *Perspectives on Sentence Processing*, chapter 7, pages 155–180. Lawrence Erlbaum Associates, Hillsdale, N.J.

Tura, S. (1986). Definiteness and referentiality in Turkish non-verbal sentences. In D. I. Slobin and K. Zimmer, editors, *Studies in Turkish Linguistics*, volume 8 of *Typological Studies in Language*, pages 165–194. John Benjamins Publishing Co, Amsterdam.

Turan, U. (1995). *Null vs. overt subjects in Turkish discourse: A Centering analysis*. Ph.D. thesis, University of Pennsylvania, Philadelphia, Pa.

Underhill, R. (1972). Turkish participles. *Linguistic Inquiry*, **3**, 87–99.

Uszkoreit, H. (1986). Categorial unification grammars. In *Proceedings of the 11th International Conference on Computational Linguistics*, pages 187–194, Bonn. Bonn University. Also CSLI Report CSLI-86-66.

Uszkoreit, H. (1987). *Word Order and Constituent Structure in German*. Number 8 in CSLI Lecture Notes. Center for the Study of Language and Information, Stanford, Ca.

Vallduví, E. (1992). *The Informational Component*. Garland Publishing, New York.

Vallduví, E. and Engdahl, E. (1995). Information packaging and grammar architecture. In J. Beckman, editor, *Proceedings of the Twenty-Fifth Annual Meeting of the North Eastern Linguistics Society*, pages 519–533. University of Pennsylvania.

Vallduví, E. and Engdahl, E. (1996). The linguistic realization of information packaging. *Linguistics*, **34**(3), 459–519.

van der Linden, E.-J. (1992). Incremental processing and the hierarchical lexicon. *Computational Linguistics*, **18**(2), 219–238.

Young, M. A. and Rounds, B. (1993). A logical semantics for nonmonotonic sorts. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 209–215, Columbus, Ohio. Ohio State University.

Zeevat, H., Klein, E., and Calder, J. (1987). An introduction to unification categorial grammar. In N. J. Haddock, E. Klein, and G. Morrill, editors, *Edinburgh Working Papers in Cognitive Science: Categorial Grammar, Unification Grammar and Parsing*, volume 1, pages 195–222. Centre for Cognitive Science, University of Edinburgh.