

INFORMATION TO USERS

The most advanced technology has been used to photograph and reproduce this manuscript from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

U·M·I

University Microfilms International
A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
313/761-4700 800/521-0600

Order Number 9108147

**On the learnability of a version of generalized phrase structure
grammar**

Maxfield, Thomas Lane, Ph.D.

City University of New York, 1990

Copyright ©1990 by Maxfield, Thomas Lane. All rights reserved.

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

A

ON THE LEARNABILITY OF A VERSION OF
GENERALIZED PHRASE STRUCTURE GRAMMAR

by

THOMAS L. MAXFIELD

A dissertation submitted to the Graduate Faculty
in Linguistics in partial fulfillment of the
requirements for the degree of Doctor of
Philosophy, The City University of New York.

1990

© 1990

THOMAS L. MAXFIELD

All Rights Reserved

This manuscript has been read and accepted for the Graduate Faculty in Linguistics in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

September 25, 1990

Date

Janet Fodor
Janet Dean Fodor,
Chair of Examining Committee

Sept. 25, 1990

Date

Richard S. Kayne
Richard S. Kayne,
Executive Officer

D. Terence Langendoen

John A. Moyne

Pauline Jacobson
Supervisory Committee

The City University of New York

Abstract

ON THE LEARNABILITY OF A VERSION OF GENERALIZED PHRASE STRUCTURE GRAMMAR

by

Thomas L. Maxfield

Adviser: Professor Janet Dean Fodor

This study examines the version of Generalized Phrase Structure Grammar (GPSG) proposed by Gazdar, Klein, Pullum and Sag (GKPS, 1985). The theory is impressive in the scope of its coverage of English and in the subtlety with which its principles interact to determine sentence structure. However, these principles are also complex. It is argued here that the complexity can be greatly reduced if, following Shieber (1986), we impose an ordering on the principles, rather than allowing them to apply simultaneously as GKPS did. This forces us to reject GKPS' view that the various principles only check feature values in favor of the view that the principles actually supply the features.

Shieber's modifications of GKPS are not sufficient. Other changes are needed, not only to achieve simplicity but also to maintain descriptive adequacy. The ordering of principles which is developed leads to the conclusion that non-default values must be inherited from the ID rule and that there is almost no 'free instantiation'. Instead, the principles of feature

instantiation are procedures that copy feature values from one node to another, within a local tree.

The Control Agreement Principle (CAP), which is the locus of much of the complexity of GKPS, is examined. GKPS use the CAP in describing many different linguistic phenomena. It is argued that the grammar can be simplified, and more realistic generalizations can be expressed, by separating the analyses that the CAP collapses. Once this is done, the principles of feature instantiation become very similar: they all copy features from mothers to daughter.

This theory is very similar to the theory Fodor and Crain (in press) argued for on the basis of learnability considerations. We examine the changes that must be made to make the theory developed so far learnable. The only changes required are elimination of the parochial constraints. Once this is done it can be shown that the grammar proposed here for English could be learned by a child, assuming no negative data and a general, symbol counting simplicity metric.

ACKNOWLEDGEMENTS

The thought of having to try to thank so many people for so much makes the thought of writing of these acknowledgements almost as intimidating as the thought of writing a dissertation. But, it must be done, so here is my attempt at expressing my gratitude to those who shared my life.

First, I need to thank my thesis committee, who were the best first, last and always. Without their various expressions of belief in me, I could have never come this far. Janet Fodor has gently pushed, pulled and prodded me to strive for excellence since the beginning of our acquaintanceship. I have yet to leave a meeting with Janet without feeling equal to the task and eager to start work again. Janet first insisted on seeing a draft when I felt unable to put two connected ideas on a page. And she continued reading them after I *thought* I could. The quality of this thesis owes much to the number of drafts Janet read and the quality of the comments she made. Although the actual writing was often less than joyful, the hours spent with Janet discussing the ideas involved have made the whole process quite enjoyable. I can only hope that some day I can pass on to my students the same level of linguistic depth and human kindness.

Terry Langendoen has been there to encourage and excite me from the beginning of my graduate education. My indebtedness to Terry is immense. He advised me when I was unsure about decisions, and has supported my decisions consistently. Terry taught me that it is alright to attack a new area; that understanding will come with persistence. Terry held my hand and told me that everything would work out - and then sent me back to write it up. John Moyne was there to support me; to listen and then ask "why?" Thanks to John, several of my positions had to be rethought and defended. I could not have asked for (or imagined) a better outside reader than Polly Jacobson. Polly's insightful comments improved much of my thinking (and hopefully the content of this work). For going out of her way to work within my time constraints, I cannot thank Polly enough.

My education at CUNY was made exciting and meaningful by the diversity of our faculty and students (not to mention two of the best in support staff: Judith Tucker and Liz Hogan). The faculty who seem to have had the biggest influence on my training are: Helen Cairns, Bob Fiengo, Janet Fodor, Mike Hall, Arnie Koslow, Terry Langendoen, and Sam Levin. I hope to be able to measure up to the standards they have set. My fellow students have also done a lot to educate me and teach me how to "do linguistics". For friendship (both linguistic and

otherwise), I would especially like to thank Duku Anokye, Mike Barnett, Ed Battistella, Judy Bernstein, Jill Burstein, Wayne Cowart, Amy Diauta, Lorna Feldman, Tracey Forrest, Cindy Greenberg, Alice Greenwood, Sheila Haber, Alice Hopkins, Jennifer Hsu, Nina Hyams, Kris Johnson, Kathleen Kane, Nancy Kelly, Elaine Klein, Claudia Leacock, David Man, Debbie Mandelbaum, Marco McCormick, Dana McDaniel, Pam McPartland, Sheila Meltzer, Ria Olsen, Carmen Picallo, Judy Pierpont, Norie Sanchez, Gloria Schlisselberg, Marcy Serkin, Katherine Steinberger, Nina Wacholder and Tom Yohannan.

During my time in New York, in graduate school, many people have taken me into their lives and helped me through the ups and downs of my own life. Without the emotional support that they have provided (not to mention the grammaticality judgments), I don't think that I would have made it through. Something more than a simple "thanks" is needed to express what I owe to Alice, Bob, Dara, David and Stephanie Greenwood, Maureen Herman, Nina Hyams, Kris, TJ, Torsten, Alisa, Mariah and Gus Johnson, Judy Klavans, Ivan and Lisa Klavans-Rekosh, Dana McDaniel, and Carmen Picallo. I hope that some day some one does for you what you all have done for me.

Finally, I need to thank my families. When I started all of this, I had no idea how much I would rely on them. They were there to support and encourage me, to

comfort me or push me back to work (depending on what was needed), to share with me the laughs, tears and desserts of life. What can I say but "thanks", to Kris and TJ Johnson, who let me in and shared with me the best; to Torsten, Alisa, Mariah and Gus Johnson, who laughed and played and slept in my arms; to the rest of the family that they took me into, August and Marilyn Gjerlow and Dorothy, Ray, Stephanie and Douglas Nygard? Words cannot express what you have and do mean to me and how much I owe.

My biological family has done so much for me throughout the years of my education that I cannot ever hope to repay them. They have continually expressed their belief that I would make good choices in my life, when I had serious doubts about that. They have supported me in every way possible while I have tried to find my way through the maze of graduate school. They have put up with me through exam anxiety and tried to understand why it mattered if that sentence was grammatical or not. They have picked me up at 3 a.m. and seemed happy to celebrate Christmas two weeks late. They have listened to me kvetch for hours when they didn't have a clue where or why I wanted to move- α anyway. Without Catherine, Dude, Dawn Marie and Lane Underwood, my life would have been much bleaker. Phyllis and Lane Maxfield gave me life, and then a whole lot more. Words

do not suffice, so I will only say that, if I had had a choice, I couldn't have chosen better and wouldn't have chosen different.

To Phyllis, and the memory of Lane

TABLE OF CONTENTS

ABSTRACT	iv
ACKNOWLEDGEMENTS	vi
Chapter	
1. 1.1 <u>Arguments against PSGs</u>	1
1.2 <u>Countering those arguments</u>	3
1.3 <u>Linguistics and Learnability</u>	11
2. STANDARD GENERALIZED PHRASE STRUCTURE GRAMMAR . .	15
2.1 <u>Overview of GKPS</u>	15
2.1.1 Categories and Features	17
2.2 <u>Immediate Dominance and Linear Precedence</u> .	20
2.2.1 Immediate Dominance Rules	20
2.2.2 Linear Precedence Statements	22
2.2.3 Metarules	25
2.2.4 Subcategorization	27
2.2.4.1 <u>Projection</u>	30
2.3 <u>Feature Co-occurrence Restrictions</u>	33
2.4 <u>Feature Instantiation</u>	34
2.4.1 The Foot Feature Principle	37
2.4.1.1 <u>The Normal Case</u>	37
2.4.1.2 <u>'Exceptions' to the FFP</u>	39
2.4.2 The Control Agreement Principle	43
2.4.2.1 <u>Control and Control Features</u>	43
2.4.2.2 <u>The CAP at Work</u>	49
2.4.3 The Head Feature Convention	53
2.4.3.1 <u>The Normal Case</u>	53
2.4.3.2 <u>'Free' Feature Specifications</u>	55
2.4.3.3 <u>The HFC at Work</u>	57
2.4.3.4 <u>'Exceptions' to the HFC</u>	59
2.5 <u>Feature Specification Defaults</u>	61
2.5.1 Local Admissibility	61
2.5.2 Lexical Insertion	68
2.5.3 Non-local Trees	69
3. SOME PROBLEMS WITH FEATURE INSTANTIATION	73
3.1 <u>Complexity</u>	74
3.1.1 Complexity in Linguistics	74
3.1.2. Complexity in GPSG	75

Chapter	Page
3.1.2.1. <u>Subcategorizational complexity.</u>	75
3.1.2.2 <u>Complexity in UFI.</u>	82
3.2 <u>Simplification Based on Ordering</u>	84
3.3 <u>Ordering the HFC</u>	96
3.3.1 The HFC and the FCRs	99
3.3.2 The HFC and the FFP	100
3.3.3 The HFC and the CAP	102
3.3.4 The HFC precedes the CAP	107
3.3.4.1 <u>Reordering the FSDs.</u>	108
3.4 <u>The Revised Ordering</u>	109
3.4.1 Eliminating some non-lexical FSDs	119
4. CONCERNING THE CAP	121
4.1 <u>Focusing on the CAP</u>	121
4.1.1 What the CAP does	122
4.1.2 How the CAP does it	123
4.1.3 Problems for the CAP	124
4.1.3.1 <u>What features the CAP affects.</u>	124
4.1.3.2 <u>Subject-aux inversion.</u>	137
4.1.3.3 <u>A top-linking problem.</u>	141
4.1.3.4 <u>Connectivity.</u>	144
4.2 <u>Data Covered by the CAP</u>	149
4.2.1 Verbal agreement	150
4.2.2 Irregular Subjects	158
4.2.3 Object control	162
4.2.4 Subject control	163
4.2.5 <u>Tough</u> movement	165
4.2.6 SLASH termination metarule 2	172
4.2.7 Top-Linking	177
4.2.8 Features needed	182
4.2.9 Summary	184
4.3 <u>The ADP</u>	186
4.3.1 Genitive Nouns	188
4.4 <u>Ordering revisited</u>	191
4.4.1 The HFC	191
4.5 <u>Conclusion</u>	196
5. LEARNABILITY	197
5.1 <u>A model of learning</u>	198

Chapter	Page
5.2 <u>Marking Rules</u>	210
5.3 <u>Back to GKPS</u>	216
5.3.1 Feature Instantiation	216
5.3.2 The FCRs	219
5.3.3 FSDs	243
5.3.3.1 <u>The function of FSDs.</u>	243
5.3.3.2 <u>The FSDs.</u>	250
5.4 <u>Learning Rules</u>	255
6. CONCLUSIONS	275
6.1 Summary of Arguments	275
6.2 Further Development of the Theory	278
6.2.1 Eliminating extrinsic ordering	278
6.2.2 A general theory of features	292
6.2.3 Implementation	294
REFERENCES	296

CHAPTER ONE

INTRODUCTION

One of the concerns of the early work in Transformational Generative Grammar (TG) was motivating the need for transformations. To do that, it was necessary to show that phrase structure grammars (PSGs) alone were insufficient for the description and explanation of natural languages. The usual position that was taken was that context free phrase structure grammars (CFPSGs) were formally too weak and that context sensitive phrase structure grammars (CSPSGs) were too powerful. Therefore a theory whose formal power was between that of CFPSG and CSPSG was needed. TG was proposed to fill this need. Various arguments were put forth to prove this point, and they soon became accepted as linguistic truth.

1.1 Arguments against PSGs

The arguments usually came in one of two forms. The first kind says that although CFPSGs could in

principle describe the facts about natural languages, to do so would be cumbersome and would therefore miss the relevant generalizations. This kind of argument is exemplified in Chomsky (1957, 1965). Chomsky shows how the elements in AUX (i.e., those verbal elements that can follow the subject and precede the main verb) are consistently ordered with respect to one another and how the presence of one will influence the morphological form of the following verb. Since each of these elements (except tense) is optional, the number of CFPS rules necessary to generate all of the possible combinations is quite large. Even if one were to accept that, there is still the fact that the grammar as a whole would miss the generalization of how each element consistently affects the form of the following element. That generalization would have to be encoded directly into rule after rule and thus would seem to be an accident; it could easily have been different. This consequence was felt to be unacceptable and so CFPSGs were insufficient as a theory of grammar.

The second kind of argument against CFPSGs was that they could not even describe certain phenomena that occur in natural languages. That is, while arguments of the first type claimed that CFPSGs were descriptively inadequate, arguments of the second type claimed that they were observationally inadequate. One example of

this kind of argument involves WH-questions in English. The syntactic distance between a fronted WH-word and its associated gap is unbounded. That is, there can be no longest English sentence beginning with a WH-word and ending with the position where that word is required by subcategorization. Since a PS rule can only describe a local relationship (i.e., one involving only a mother and its daughters), if this unbounded relationship is not local, then its description is beyond the power of PSGs. Given the fact that we want to say that there is no grammatically based limit on the number of clauses that may be embedded within any matrix sentence, the relationship between a WH-word and its gap will not be local. Therefore, CFPSGs are inadequate as a theory of grammar. However, a TG could describe, and arguably explain, these facts about English. Thus, transformations were taken to be well motivated.

1.2 Countering those arguments

By the mid to late sixties, these basic arguments were generally accepted without debate and the general assumption was that transformations were in fact necessary to linguistics. But, by the early seventies, several things were beginning to change. One important event at this time was the finding of Peters and Ritchie (1973) that not only were TGs formally more powerful than

CFPSGs (which were regarded as formally too weak) but TGs were also formally more powerful than CSPSGs (which were held to be formally too strong).¹ That is, TGs (at least as developed in the Standard Theory, cf. Chomsky, 1965) were much too powerful to be a reasonable theory of grammar. This finding served to reinforce the trend in TG of restricting the function (and therefore the power) of the transformational component. But, it also served to reinforce another trend that was happening in syntax.

In the mid to late seventies, various linguists were becoming convinced that TG was not the appropriate approach to syntactic theory. Therefore, by the late seventies, one finds in the literature (or in widely circulated, but unpublished works) the beginnings of work in Lexical-Functional Grammar, Relational Grammar, Categorical Grammar, Montague Grammar, and Generalized Phrase Structure Grammar. Fueled by Peters and Ritchie's results, these approaches to syntax all share the property of being non-transformational. They differ, however, in how they compensate for the lack of transformational devices. One of the things that makes Generalized Phrase Structure Grammar (GPSG) unique in

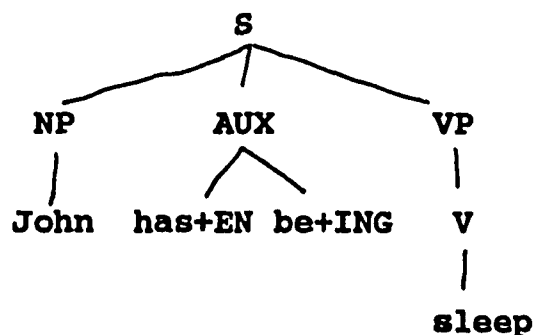
¹As discussed below, the 'proofs' that CFPSG is formally too weak to describe natural languages have been shown to be invalid. Shieber (1985) and Culy (1985) have argued, based on new evidence, that natural languages do in fact slightly exceed the limits of CFPSG. However, these arguments do not affect the points being made here.

this group is that it started by going back and re-examining the standard arguments against CFPSGs and questioning their validity.

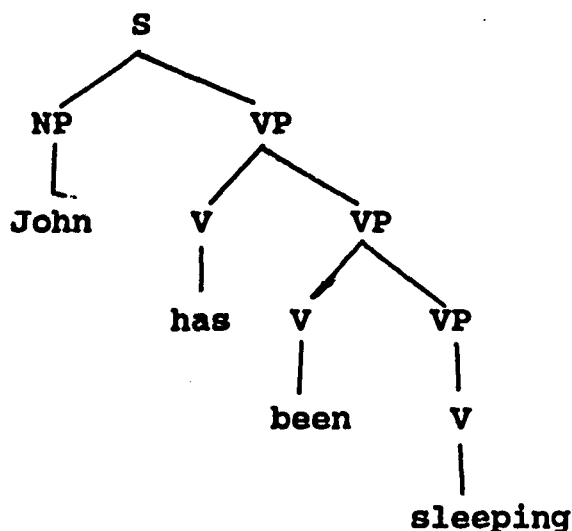
The motivation for this is quite simple. Formally (mathematically) speaking, we understand a lot about CFPSGs, and even CSPGs. Also, for CFPSGs, there are algorithms that will parse their sentences in real time (that is, in an amount of time proportional to the cubed length of the string, at worst; see Earley 1970). Neither of these are true for the other linguistic theories. The suggestion is that, since we understand CFPSG, and it is taken to be formally too weak, if we base our theory on that, and then extend its formal power only when the linguistic data force us to, we will end up with a theory of the appropriate formal power. Methodologically, this is a better (safer) approach than starting with an over-powerful theory and trying to constrain it. However, if CFPSGs are much too weak, then they will be of little use. What we want is a theory of grammar that is only just too weak, very close to the appropriate formal power. The question then is, are the initial arguments against CFPSG valid? If we must extend the power of CFPSG even to describe adequately the English auxiliary system, then we should probably start from a more powerful base.

The argument against CFPSGs from the English auxiliary system is based on the assumption that there is an AUX node and that the elements within it exhibit a series of discontinuous dependencies (see Chomsky, 1957). If we assume that there is in fact an AUX node, then this system can be described by a PSG system since the relationships are local. However, the discontinuous dependencies make the analysis cumbersome and unrealistic. But this is a problem with the analysis and not a problem with the theory it is based on. An alternative analysis which is simple and which works quite well takes the auxiliary verbs to be verbs which are subcategorized for a VP sister whose head verb is of a certain morphological form. That is, instead of the structure in (1), we have the structure in (2).

1)



2)



Thus, instead of assuming lexical items like have+EN or be+ING and a transformation like AFFIX HOPPING, we can instead assume that the auxiliary verb have is subcategorized for a VP whose head must have past participial morphology. We need then a (percolation) device to get the information about morphological form from the subcategorized-for VP down onto its head. But that kind of device is independently needed for verbs like require, which are subcategorized for an S-bar whose head V is uninflected (the 'subjunctive' verbs). The device that gets that information from S-bar down to its head verb, can also work for the VPs subcategorized for by the auxiliary verbs. It is not uncommon for a lexical head to place restrictions on the morphological form of its complements and that can be described in terms of local relationships. Thus, we have a workable solution

to English auxiliaries that is completely within the bounds of CFPSG.

Gazdar (1981) has shown that it is in fact possible to describe unbounded dependencies with a CFPSG.² Actually, what Gazdar uses is not a traditional CFPSG; instead, he uses a CFPSG with complex symbols and a metagrammar. The innovation that Gazdar proposes, in terms of complex symbols, is to generate a set of derived symbols of the form α/β , which is taken to represent a category of type α which is missing a constituent of type β , for all categories α and β . Thus, the category S/NP is just like the category S , except that it has an NP-shaped hole in it. He then uses a metagrammar to generate the CFPS rules that make use of these derived symbols. A metagrammar, in this sense, is simply a finite system for generating an infinite (or potentially infinite) system (here the infinite system consists of CFPS rules). Essentially, on this view, a metagrammar is a way of stating generalizations across PS rules. Langendoen (1976) shows that an infinite CFPSG that is generated by a finite metagrammar (of the appropriate type) still has all of the properties of a CFPSG.

²Much, if not most, of what was proposed in Gazdar (1981) has since been replaced with different formalisms or reinterpreted, as will be discussed in Chapter Two. The point being made here is still valid, however.

The metarule that Gazdar proposes says, informally, that for all nonterminal symbols α and β , if there is a rule in the grammar that is of the form in (3a), there is also another (derived) rule with the form in (3b).

- 3) a. $\alpha \rightarrow X Y Z$
 b. $\alpha/\beta \rightarrow X Y/\beta Z$

There is nothing special about these derived rules: they have the same subcategorizational and semantic properties as the basic rules they were derived from. Therefore they do not need to be separately specified or listed; they are completely predictable from the metarule, given the basic rules. All that is needed now is rules to introduce the first 'slashed category' (called "top-linking rules") and rules to realize a slashed category at the lexical level (called "bottom-linking rules"). Top-linking rules have the general shape given in (4a) and bottom-linking rules can (in general) be handled by the rule schema in (4b). (In either rule, both instances of XP are taken to be identical category specifications.)

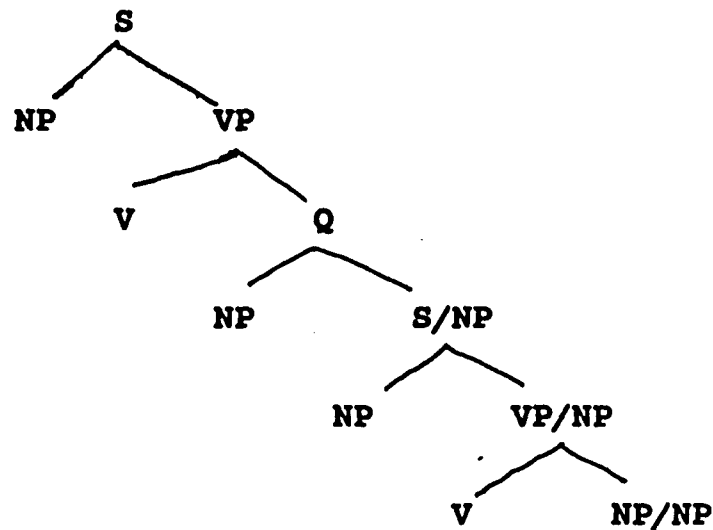
- 4) a. $YP \rightarrow XP ZP/XP$
 b. $XP/XP \rightarrow e$

Given this approach, the sentence in (5a) will have the structural description in (5b).³

³Using Gazdar's terminology, "Q" is essentially an S-bar that is marked +WH.

5) a. I wonder who they saw.

b.



On this analysis, an 'extracted' element is connected to the 'extraction site' by a series of slashed categories. The first slashed category is licensed by its sister NP (via a top-linking rule), the medial slashed categories are licensed by PS rules derived from the metarule, and the last slashed category is cashed out as a trace via the bottom linking rule. Thus, this unbounded relationship is broken up into a series of local relationships and therefore falls within the bounds of CFPSG. Gazdar shows how this PSG approach to unbounded dependencies can in fact work, how it can incorporate various island constraints, and how, in conjunction with an analysis of coordinate structures, it can account for the 'across-the-board' phenomena. (For fuller discussion of this version of the theory, see Gazdar 1981, 1982, and Gazdar and Pullum 1982.

Although the original arguments against the context-free nature of natural languages have been refuted (see Gazdar, 1982 and Gazdar and Pullum, 1982), new analyses have been proposed to show that natural languages are not completely context-free (see Shieber, 1985 and Culy, 1985). These arguments, however, do not contravene the approach of GPSG. The point of the GPSG program is that languages are only slightly beyond the formal power of CFPSGs, if they are beyond at all. Therefore, the best research strategy is to push CFPSGs as far as possible, and only extend that mechanism when the data from natural languages absolutely necessitate it. Of course, this program will be valid only if there are ways to extend CFPSG *slightly* in order to cover the data from natural language. Otherwise a move to a theory completely different from CFPSG would be justified.

1.3 Linguistics and Learnability

Chomsky (1965) extensively discusses the notions of observational, descriptive and explanatory adequacy. These notions have been influential in the development of the field over the last 25 years, giving us a terminology to use in comparing different grammars or different theories of grammar. It is in general held to be true that there is an implicit tension between the requirements of descriptive adequacy and those of

explanatory adequacy. That is, in order to describe languages, linguists seem to need more, and more powerful, devices, while to explain how a child acquires a given grammar, linguists seem to need fewer, and simpler, devices. In fact, Chomsky set this tone for the field in the following:

A theory may be descriptively adequate . . . and yet provide such a wide range of potential grammars that there is no possibility of discovering a formal property distinguishing the descriptively adequate grammars, in general, from among the mass of grammars compatible with whatever data are available. In fact, the real problem is almost always to restrict the range of possible hypotheses by adding additional structure to the notion "generative grammar." For the construction of a reasonable acquisition model [= explanatory theory, TLM], it is necessary to reduce the class of attainable grammars compatible with given primary linguistic data to the point where selection among them can be made by a formal evaluation measure. This requires a precise and narrow delimitation of the notion "generative grammar" -- a restrictive and rich hypothesis concerning the universal properties that determine the form of language. . . .
(1965, p. 35)

One can see this tension in the transformational literature of the late 60's and early 70's. As linguists were describing more languages in more detail within TG, the number and kind of transformations being posited continued to increase, in order to make the description provided by each individual grammar more elegant. It became obvious, however, that given the kinds of choices that the theory was making available, a child simply could never determine what the appropriate grammar is for

her language.⁴ Thus there was an effort to reduce the number of, and the operations performed by, possible transformations. This effort (coupled with motivation from the Peters and Ritchie finding) resulted in the transformational component being reduced to the single rule "move α ". One can take the move to non-transformational theories to be the logical conclusion of this effort: in these grammars, the transformational component has been completely reduced; it does not exist.

Thus, it can be seen that there are at least two reasons to investigate GPSG. On the descriptive level, it has approximately the correct formal power. Therefore, it might be able to provide a descriptively adequate grammar for all natural languages and methodological considerations would suggest that it is the theory to start with, at least. On the explanatory level, since GPSG is not overly powerful, it should not give the child too many choices when learning a grammar. That is, GPSG offers the potential of being an explanatorily adequate theory.

Given the (assumed) tension between descriptive and explanatory needs, it is then significant when arguments based on these different needs converge on the

⁴Throughout this work, I will use feminine pronouns to refer to the generic child. I do this simply for expository ease.

same analyses. The present work is a case study of such an occurrence. It is organized in the following manner. Chapter Two examines the most thoroughly worked out version of GPSG, that of Gazdar, Klein, Pullum, and Sag (1985).⁵ It is simply a detailed exposition of their extensive work. Chapters Three and Four are dedicated to simplifying the theory that Gazdar, Klein, Pullum, and Sag propose, based on purely linguistic considerations. A proposal by Shieber (1986) is examined and modified in Chapter Three, resulting in significant simplification. In Chapter Four, a particularly difficult principle of GPSG (the Control Agreement Principle) is scrutinized and revised, which also allows for considerable simplification. Chapter Five explores the interactions of the theory of language learning proposed by Fodor and Crain (in prep.) with the theory developed in Chapters Three and Four.

⁵A more recent development of GPSG, namely Head-driven Phrase Structure Grammar (HPSG) (see Pollard and Sag, 1987) will generally be ignored here. This is done for two reasons. First, only part of the theory is available in published form, and the missing portion is important to the work at hand. Second, HPSG is formulated completely within the framework of Situation Semantics (see Barwise and Perry, 1983) and thus is dependent on the success of that theory for its own success. GPSG, on the other hand is much more independent of any particular theory of semantics, and can be judged on its own merits.

CHAPTER TWO

STANDARD GENERALIZED PHRASE STRUCTURE GRAMMAR

Chapter One briefly gave the motivation for developing GPSG and gave the structure of the theory in those early days (see Gazdar 1981, 1982 and Gazdar and Pullum 1982). Much work has been done with the GPSG framework since that time. As with all active theoretical frameworks, GPSG has changed radically over the years. This chapter will explore in detail the most thoroughly worked out version of GPSG to date, namely that of Gazdar, Klein, Pullum and Sag (1985).

2.1 Overview of GKPS

Gazdar, Klein, Pullum and Sag (1985) (henceforth GKPS) propose that a grammar for a natural language is a Generalized Phrase Structure Grammar (GPSG)¹ with the components in (1).

¹In the following discussion I will gloss over or ignore aspects of GPSG which are not central to the following chapters. For example, I will not discuss the rules of semantic interpretation or the motivation for metarules.

- 1)
 - a. immediate dominance rules
 - b. metarules
 - c. linear precedence statements
 - d. feature co-occurrence restrictions
 - e. feature specification defaults
 - f. universal feature instantiation principles
 - g. semantic interpretation rules

The first five components are (at least partially) language particular (parochial), i.e., they are where languages can differ from one another to some extent. The last two components are taken to be universal.²

Immediate dominance (ID) rules are like context-free phrase structure (CFPS) rules, except that the righthand side of the rule is taken to be unordered. Metarules map ('basic') ID rules into ('derived') ID rules. The linear precedence (LP) statements are what supply the orderings among sister categories that were missing in the righthand side of the ID rules. Feature Specification Defaults (FSDs) and Feature Co-occurrence Restrictions (FCRs) help to determine what feature specifications occur within a category. FCRs limit the feature specifications that can co-occur within a category. FSDs specify what value a feature will have if nothing else causes it to have a different specification. The principles of universal feature instantiation (UFI) also limit the feature specifications in a category but

²Sells (1985) says that the feature instantiation principles may be subject to finite variation across languages, but he does not provide an example of how this might be so.

they take into account more than one category in a local tree. That is, like LP statements, they are relational in nature.

All of these components conspire to determine all and only the local trees (i.e., trees consisting of only a mother and its immediate daughters) that are used in a language. Global trees are built up by putting together acceptable local trees. The connecting node between two trees is the mother of one tree and a daughter of the other, so these two nodes must bear identical category labels. Before moving to a more thorough discussion of the grammar components, it will be useful to consider the notions of 'category' and 'feature' used by GKPS.

2.1.1 Categories and Features

Following Chomsky (1970), the categories that make up the ID rules are not taken to be monadic, but are sets of <feature, feature-value> pairs, with the possibility of internal structure among the pairs. (As an example of internal structure, GKPS cite work by Stucky (1981) and Horrocks (1983) that argues for the use of ordered pairs of sets of agreement feature specifications in languages where verbs are marked for both subject and object agreement.) Many of the features used are standard features used in almost any linguistic analysis, for example the features for person (PER),

number (PLU), tense (PAST) and so forth. The other features used are either newly proposed for GPSG analyses, such as 'NFORM' (which distinguishes expletive there and it from other NP's) and INV (which marks whether a sentence has an inverted auxiliary verb) or they are familiar attributes of categories that have been reinterpreted as features, such as BAR (which marks the X-bar level of categories). The label 'NP', then, is just a convenient short-hand notation for a category which has at least the specifications: {<N,+>, <V,->, <BAR,2>}. (Following GKPS, I will use certain abbreviatory conventions when it will cause no confusion in dealing with <feature, feature-value> pairs. For example, [+N] stands for {<N,+>}, X^2 stands for a category specified as {<BAR,2>} and [ACC] abbreviates {<CASE,ACC>}.)

It is important to be able to pick out certain subsets of features, as they seem to act alike. This is done by assigning a name to the subset, for instance: HEAD = {N, V, PLU, PER, VFORM, SUBJ, PFORM, AUX, INV, PAST, PRD, SLASH, AGR, SUBCAT, BAR, LOC}. We can then talk about a mother and its head daughter having to agree in HEAD feature specifications. The other group of features that will play a significant role in the following discussion is the FOOT features, defined as

{RE, SLASH, WH}.³ There is also a third set of features, which Sells (1985) calls "non-propagating" features, to emphasize the fact that they are usually only on one category in a local tree and do not 'pass' through the tree. This set is defined in GKPS as {CASE, COMP, CONJ, GER, NEG, NFORM, NULL, POSS, REMOR, WHMOR}.

GKPS regard categories as partial functions from features into feature-values, which are both finite sets. This insures that no category can include any one feature more than once, where the different occurrences of that feature have distinct values. Since a category C is a partial function, C can be undefined for some feature f (represented by ' $\sim f$ ') and we say that f is not in the domain of C ($f \notin \text{DOM}(C)$). Fully specified categories are not given a special status in GPSG: both rules and trees may contain underspecified categories. However, trees which are admitted by the grammar will, in general, contain only fully specified categories (see discussion below). The category [$\langle \text{BAR}, 2 \rangle$] is, therefore, as legitimate as the category [$\langle \text{BAR}, 2 \rangle, \langle +N \rangle, \langle -V \rangle, \langle \text{NOM} \rangle, \langle \text{PER } 3 \rangle$, etc.]. This then allows the empty set, $\{\}$, to be a possible category.

³The values of RE represent reflexive and reciprocal morphology, SLASH is used in the analysis of unbounded dependencies, and WH marks WH-phrases and has the ability to encode morphological differences between relative pronouns, interrogative pronouns, free relatives and exclamatives.

Features, which are atomic symbols, can take as their values either an atomic symbol (such as those discussed above) or a category. The notion of category-valued features is less familiar than that of atomic-valued features, and causes some interesting wrinkles in the formal machinery (cf. discussions below on extension and unification). The SLASH ("/") annotation of earlier work in GPSG is reinterpreted as a category-valued feature and it is used in the GPSG analysis of unbounded dependencies (long-distance movement or deletion in TG). Intuitively, the specification $\langle \text{SLASH}, \{ \langle \text{N}, + \rangle, \langle \text{V}, - \rangle, \langle \text{PLU}, + \rangle, \langle \text{PER}, 2 \rangle, \langle \text{CASE}, \text{NOM} \rangle \} \rangle^4$ as part of a category means that this category is 'missing' an accusative second person plural noun phrase; in the most usual case, (though not necessarily) the category would contain an empty (NULL) accusative second person plural NP.

2.2 Immediate Dominance and Linear Precedence

2.2.1 Immediate Dominance Rules

One of the major innovations in linguistics in the 1970's was the introduction of X-bar syntax (cf. Chomsky, 1970; Jackendoff, 1977). The insight that it is no accident that phrasal categories have heads and that

⁴Given the abbreviations above we can write, $\langle \text{SLASH}, \text{NP}[+\text{PLU}, \text{PER } 2, \text{NOM}] \rangle$ or, even more simply, $/\text{NP}[+\text{PLU}, \text{PER } 2, \text{NOM}]$. I will use both of these abbreviations for convenience.

those heads agree with the mother in most features is now generally accepted and assumed in linguistic discussions.⁵ Since it is the case that PS rules have heads, it is redundant to have to specify certain features (such as major category features, number and tense) on both the mother and the head daughter(s) since they are always identical; these redundant features can be omitted from the rules. To achieve both of this, GKPS use the metagrammatical variable H, on the righthand sides of ID rules, as a place-holder for the head of the rule. It is then the job of the Head Feature Convention (see the discussion below) to insure that the appropriate HEAD features on the mother and head daughter(s) are the same. So, the rule in (2a) is the GPSG version of the more standard rule in (2b).⁶

- 2) a. N[BAR 2] --> H[BAR 1], DET
 b. N' --> DET . N'

⁵See Pullum (1985a) for a discussion of what is and is not essential to the assumptions of X-bar theory.

⁶Except, of course that the daughters in (2a) are unordered with respect to each other. Note, also, that the decrease in bar level is explicit in (2a). This is because GKPS take BAR to be a HEAD feature. Therefore, their version of X-bar theory does not meet the condition that Pullum (1985a) calls 'succession'. That is, mother and head daughters will have the *same* BAR feature specification unless the ID rule explicitly dictates otherwise.

(Following GKPS, when discussing rules or trees, I will omit many feature specifications, if they are not relevant to the point at hand. I do this simply for expository ease.)

2.2.2 Linear Precedence Statements

A GPSG can be converted into an equivalent CFPSG, but not all CFPSG's can be converted into GPSG's, as GKPS define them.⁷ Thus, the weak generative capacity of GPSGs is equivalent to that of CFPSGs. GKPS split standard CFPSG rules into two parts, Immediate Dominance (ID) rules and Linear Precedence (LP) statements. Removing linear precedence from the statement of the rules has the consequence that the rule in (3) simply means that the category A may dominate the categories B, C and D, but they can be in any order.

3) A --> B, C, D

(The daughter categories must be written down in some order, but that is a reflection of the orthography; it

⁷Earlier versions of GPSG could not be converted to CFPSG's as the metarules they contained could generate an infinite set of derived rules. This is not the case with the GKPS version, since they restrict the application of metarules such that a metarule may not apply to rules that have resulted from the application of that metarule, thereby obtaining the finite closure of the ID rules under metarule application. GKPS do use an infinite set of rules in their grammar; those induced by the coordination schema. However, these rules are Finite State, which guarantees that the resultant language will be context-free (see Langendoen, 1976).

has no theoretical import.) Therefore, (3) licenses all six of the possible orderings of the three daughters. However, many languages do not allow free ordering of sisters, and the orderings that are to be ruled out must be excluded by the LP statements.

Assume that the preterminal strings of the language that we want to generate are as in (4).

- 4) a. B . C . D
 b. B . D . C
 c. C . B . D

The generalization here is that B must always precede D and other than that, the order is free. A CFPSG could generate this by having three rules, one for each terminal string. But in ID/LP format, all that is needed is the rule in (3) and the LP statement in (5), which should be read as: 'A node labelled D cannot appear to the left of a node labelled B in a local tree'.

- 5) B < D

That is, all B's must precede all D's in any local tree. While the relative simplicity of the ID/LP format is not striking in this trivial example, it becomes much more obvious as grammars get more realistic. This, of course, would not be so if natural languages did not exhibit word order generalizations across different local trees.

Thus, on top of its value in terms of formal simplicity, the separation of ID and LP allows GPSG to explicitly capture generalizations that the CFPSG format

misses. An LP statement can require that some given category always precede another, regardless of which ID rule introduces them. But a CFPSG cannot; it must stipulate the order again in every PS rule that introduces those constituents, and the grammar does not anywhere express the generalization.

The ID/LP format commits GPSG to the claim that all natural languages have what GKPS call the Exhaustive Constant Partial Ordering (ECPO) property.⁸ ECPO means that if in the expansion of any category the daughter categories observe a partial ordering, then that ordering must also be observed by those daughter categories in the expansion of all other categories. For example, since NP and VP are consistently ordered when they are daughters of S, ECPO requires that they have the same order whenever they are sisters, regardless of what category is their mother. All GPSG's have the ECPO property while not all CFPSG's do. In this respect GPSG's are more restrictive than CFPSG's. However, in other respects GPSG's have more expressive power than CFPSG's. GKPS use

⁸Note that the ECPO property makes a strong empirical claim about possible human languages. If it turns out to be false, that would invalidate the ID/LP format for natural language description. Pauline Jacobson has pointed out to me that it would be possible to circumvent ECPO violations with the appropriate use of features. However, if we assume that the features in LP statements must be independently motivated, then ECPO is not vitiated.

an infinite schema for coordination which allows a flat structure analysis for an infinite number of conjuncts. This is strictly beyond the strong generative power of CFPSGs (i.e., it allows the generation of 'weakly CS languages') in that it is the schematization of an infinite set of rules.⁹

2.2.3 Metarules

Metarules map from ('basic') ID rules to ('derived') ID rules. In earlier versions of GPSG (cf. Gazdar 1981, 1982, Gazdar and Pullum 1982 and Jacobson 1984, for example) metarules were allowed to apply to any of the CFPS rules, but much of the descriptive work that this did has been taken over by the ID/LP format and by the principles of feature instantiation that GKPS postulate, and so it is possible to restrict metarule application to lexical ID rules. This restriction is called the 'Lexical Head Constraint'. Given this constraint, metarules are more or less equivalent to what would be lexical rules in other theories (e.g., LFG and GB).¹⁰

⁹It is not uncommon to find traditional CFPSG analyses of conjunction that also use the Kleene notation. In fact, in *Aspects*, Chomsky (1965) argues natural language coordination can be structurally flat, although he does not explicitly propose the use of the Kleene-star, which is formally not available to CFPSGs.

¹⁰Note, however, that these lexical rules are generative not merely redundancy rules.

Taken together, the ID rules and metarules define all the local trees potentially accepted by the grammar. As an example of the use of metarules, I will briefly consider the 'passive' metarule proposed by GKPS. There is a fairly regular and well-known correspondence between verb phrases that contain (at least) a verb and an object noun phrase, and verb phrases that contain the verb marked with passive morphology with no object noun phrase and an optional agent phrase. The metarule in (6) will capture this correspondence.

6) VP --> NP, W

↓

VP[PAS] --> W, (PP[by])

This says that for every rule in the grammar that allows a VP to dominate an NP (and, optionally, other material), there is another rule that allows that same VP, with the addition of the passive marking, to dominate all of the same material, minus the NP, plus an optional by-prepositional phrase. W is a metavariable and stands for a multi-set (also called a 'bag', technically) of categories (a 'multi-set' is just like a set, except that elements may be repeated a finite number of times).

The fact that no head verb is expressed in (6) is not a problem since metarules can only apply to lexical ID rules and all lexical VP expansion rules must

introduce a V.¹¹ The only features mentioned in the output of a metarule are those that it changes. Otherwise, all feature specifications remain the same; even the rule number, which expresses subcategorization information (see below) remains. This incorrectly predicts that lexical rules (or lexical generalizations) are exceptionless, which is clearly false (for example, compare own and have, which are very similar semantically, yet the latter has no passive form).

GKPS note (p. 106) that the distinction between basic and derived ID rules is not relevant to feature instantiation. Henceforth, it will not be mentioned except when it is pertinent to the point at hand.

2.2.4 Subcategorization

An ID rule is a 'lexical ID rule' if it has a head that is an extension of a SUBCAT category. GKPS assign all and only preterminal categories (i.e., those categories that are either BAR 0 or undefined for BAR) the feature SUBCAT, since only preterminals may have subcategorizational requirements.¹² The values of SUBCAT are integers (for mnemonic reasons, however, the SUBCAT

¹¹Note that GKPS do not posit a V-bar level, but rather have VP directly dominating V. They do, however, make use of single-bar projections for N, A, and P.

¹²I will, for the time being, follow GKPS in ignoring lexical items, which are the terminal symbols and are supplied by the lexicon.

value for a lexical formative (determiners, complementizers and prepositions, for example) is listed as that formative, which is less obscure than some arbitrary number). The integers that are used as the values for SUBCAT are pointers to the rules that can be used to introduce the lexical item. SUBCAT is used, for example, to insure that verbs are inserted into trees which are consistent with their subcategorizational requirements. For example, the rules in (7) (where 'V[n]' is an abbreviation for 'V[SUBCAT n]') will insure that verbs occur in the appropriate environment, given that intransitive verbs are lexically specified as [SUBCAT 1] and transitive verbs as [SUBCAT 2].¹³

- 7) a. VP --> V[1]
- b. VP --> V[2], NP

GKPS also postulate the LP statement given in (8).

- 8) [SUBCAT] < ~[SUBCAT]

This LP statement says that categories undefined for the feature SUBCAT cannot precede categories marked for SUBCAT, in any local tree. This gives many of the same

¹³Note that there is a certain amount of redundancy here because the lexical item which is inserted under the head of rule number 2 will also be specified as [SUBCAT 2]. Both the preterminal and the terminal symbols must be specified for SUBCAT, the former in the syntax and the latter in the lexicon. This redundancy is necessary, given the way local trees are put together to build up non-local trees (see below).

results as the 'Head First Parameter' (Stowell, 1981, Travis, 1984), but covers additional data. The Head First Parameter generalizes over heads, whether lexical or phrasal, whereas the LP statement (8) generalizes over lexical items, whether they are heads or non-heads (such as specifiers and conjunction words). For instance, DET must be initial in NP, since DET is a lexical category (hence, marked for SUBCAT) and N-bar is not. (The Head First analysis captures this only if NP is reanalyzed as Determiner Phrase (see Abney, 1987).) Similarly, words such as and, either, and or will precede their phrasal sisters, even though they are not heads (and could not be reanalyzed as heads without damage to the universal conventions for conjunction structures).¹⁴

Most of the ID rules that GKPS posit in their grammar of English are in fact lexical ID rules; they specify a value for SUBCAT on their head daughter. Of course there must also be nonlexical ID rules (rules which rewrite S, and put adjuncts onto N-bar, for example). The total set of ID rules for the grammar, then, is this set of nonlexical ID rules combined with

¹⁴However, the GKPS rule in (8) does not capture the fact that X-bar heads typically precede adjunct sisters in English. In fact, (8) cannot account for this generalization since BAR 1 categories cannot have the SUBCAT feature. Fodor and Crain (in press) propose a broader generalization: $[BAR\ n] < [BAR\ m]$, where $n < m$ or n is undefined.

the maximal set that is the result of taking each metarule and applying it to the set of rules which have not themselves arisen from the application of that metarule. Given that the set of lexical ID rules is finite and the set of metarules is finite, the complete set of ID rules for the grammar will also be finite. Recall that GKPS do in fact include an infinite schema for coordination. While this might seem to conflict with the claim that the grammar is finite, GKPS say ". . . the set induced by the schema for coordination, has the Type-3-generable property that guarantees context-freeness for the resultant language . . ." (p. 73) (see also Langendoen, 1976). Therefore, they claim that there is no danger to finiteness here, as GPSGs will still be weakly equivalent to CFPsGs.

2.2.4.1 Projection. The ID rules are part of the definition of the set of local trees which the grammar can potentially use to form the set of global trees that are structural descriptions for all and only the sentences in the language. The categories mentioned in the ID rules tend to be fairly underspecified so that each rule allows many local trees to be accepted by the grammar. The projection from ID rules to local trees is given as a one-to-one, onto mapping function (cf, p.78). What this function says (informally) is that an ID rule

admits a particular tree if every node in the tree is labelled by an extension of one category in the rule and every category in the rule maps onto (corresponds to) one node in the tree. A category C_i is an extension of a category C_j if C_i has at least all the atomic-valued feature specifications that C_j has and the value of every category-valued feature in C_i is an extension of the value of that feature in C_j (C_i may have more feature specifications than C_j , but the intersection of C_i and C_j must equal C_j). This is basically a superset relation, except for the additional clause about the category-valued features and the fact that C_i (the extension) must itself also be a category (function).

A tree that is admitted by a rule is called a projection of that rule. Features that are mentioned in the ID rule and are thus required to be in the projections of that rule are called 'inherited' features. A category C_i in an ID rule is called the 'licensing' category for C_i in a tree if C_i inherits feature specifications from C_i . Features which occur in the tree but are not specified in the rule are 'instantiated' features. Instantiation occurs 'freely'; that is, features not mentioned in the ID rule are unreservedly added to the categories in the projection process, up to the limitation that the category must be a function.

GKPS define the notion 'legal', where a category is *legal* if and only if all FCRs (see below) are true of it; and a category C_i is a *legal* extension of a category C_j if and only if C_i is legal and C_i is an extension of C_j . Legitimate projections must then contain legal extensions of the ID rule.

To get a sense of how the GKPS rule-to-tree algorithm works, consider first the space of all local trees whose nodes are labelled by the categories. Since nothing requires that categories be fully specified, there are a great many categories, and so this space is very large indeed. Intuitively, what the grammar must do is to limit this space to all and only those local trees that occur in the language. The ID rules narrow down this space somewhat in that certain local trees will not be compatible with any ID rule. The ID component thus defines a smaller space of potential local trees which are legitimate projections of the ID rules. Many of these local trees are ill-formed because some of the categories in the tree are incompatible with each other (an example of this would be a tree with a +PAST VP mother and a -PAST V daughter); others can never surface as part of a structural description of grammatical sentences in the language because they are not completely specified where they need to be. It is the job of the

other four syntactic components to narrow down still further the number of local trees accepted by the grammar.

2.3 Feature Co-occurrence Restrictions

FCRs and FSDs limit the features that occur within any given category. They are formally identical, but they differ in the role that they play in the grammar. FCRs are absolute restrictions; they must be true of all categories in all trees in a language. They can be either universal or parochial, but they are all negative; they state what must be excluded.

Some FCRs help define what is universally acceptable as a category. For example, GKPS postulate the FCR in (9) (which is GKPS FCR 2).

9) [VFORM] $\supset$ [+V, -N]

This says that if a category is marked for VFORM (whose values are the various forms that a verb can take, i.e., infinitive, passive, past participle, etc.) then it must be a verb. In other words, it just is not possible to mark anything except a verb (or a verb phrase) as passive, infinitive, etc. This is a universal FCR; it is true of all languages. It seems to be similar to the marking conventions given for phonology by Chomsky and Halle (1968) such as (10).

10) [+VOC, -CONS] $\rightarrow$ [-LATERAL]

Chomsky and Halle say that features like LATERAL are simply not available for the marking of vowels.

By contrast an FCR like (11) (= GKPS FCR 18) says that if a category is marked as having a COMP that is for, then it must also be marked as being infinitive.

11) [COMP for] $\supset$ [INF]

This is not only not definitional, but is also not universal; it is a parochial fact about English.

Formally, however, all FCRs are formulated in the same way and apply in the same way. Given the definition of projection (see discussion above), the FCRs must be true of all categories that occur in legitimate trees.

FSDs are concerned with feature specifications internal to a category, as FCRs are, but I will postpone discussion of them until later, as they are defined in terms of categories that label nodes of local trees that meet the principles of universal feature instantiation.

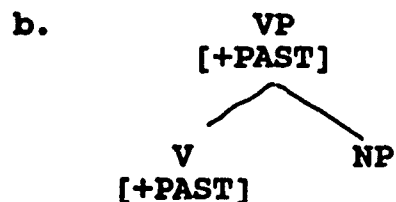
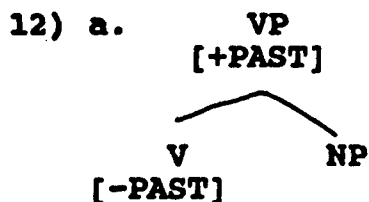
2.4 Feature Instantiation

The principles of universal feature instantiation (UFI) and LP statements are distinct from FCRs and FSDs in that they take into account more than one category. They are expressed in terms of 'sisters' or 'mothers' and 'daughters'. That is, they are necessarily relational. Feature specifications that are determined by an ID rule are 'inherited' feature specifications; the rest of the

propagating features in a tree are 'instantiated' and hence are governed by UFI. Non-propagating features occur on only one node, either by inheritance or instantiation. Thus the relational principles of UFI will not affect them.

In an optimal grammar, the ID rules will be very simple and general i.e. will mention few feature specifications. They will therefore permit many local trees to be projected by free instantiation of features not mentioned in the rule. The LP statements and UFI will eliminate some of these, reducing the set of local trees accepted by the grammar.

It is recognized in most theoretical frameworks that many features must be marked identically on mothers and head daughters (i.e., the HEAD features, such as agreement features), and these are often regarded as passing down from the mother to the head. GKPS propose the Head Feature Convention to handle these. The Head Feature Convention (HFC) says, roughly, that the HEAD features on the mother and the head daughter in a local tree must be identical (see the discussion below for a more precise formulation). This is what insures that the tree in (12a) is ruled out, but the tree in (12b) is accepted.



PAST is a HEAD feature and the HFC requires identity of HEAD features between a mother and its head daughter. This identity obtains in (12b) but not in (12a), and therefore (12b) is accepted by the grammar while (12a) is not.

However, there are other features that do not seem to fit this pattern. There are features that are shared by a non-head daughter and the mother. As an example, consider the feature WH, as in (13).

- 13) These reports, [the wording on the covers of
 which] has caused so much controversy, are
 to be destroyed.

It is clear, as GKPS point out, that it is the WH feature of which that makes the bracketed NP a WH-NP and allows it to occur in the position in which it does. Yet which is also clearly not the head of that NP (or of any subtree within that NP). Therefore the Head Feature Convention cannot guarantee that the mother and this non-head daughter agree on the feature WH. Another principle of UFI is needed for this, and GKPS propose the Foot Feature Principle (FFP).

2.4.1 The Foot Feature Principle

2.4.1.1 The Normal Case. FOOT features (SLASH, WH, and RE) are all category-valued and tend to have an overt morphological/phonological effect on a non-head daughter of the category on which they occur.¹⁵ The FFP governs only instantiated FOOT features; FOOT features specified in ID rules are simply inherited in the admissible trees which they project and the FFP ignores them. The FFP must ignore inherited FOOT features, given the GKPS analysis of top-linking and SLASH Termination Metarule 2 (STM2). If FOOT features did not stop percolating up at the point at which they were inherited in the tree, then all FOOT features in any tree would have to be on the root node. Also, the GKPS analysis of 'subject extraction' could not go through. This will be discussed more fully in Chapter Four.

Note that the discrimination between inherited and instantiated feature does not hold of the HFC; the HFC governs the occurrence of both instantiated and inherited HEAD features. This difference between the HFC and the FFP exists because the operation of the HFC

¹⁵It may well be that these two properties are essential to the notion of FOOT features, or it may be that it is just a coincidence. At present, their status is unclear. Descriptions of other languages could conceivably reveal additional FOOT features besides those needed for English.

determines (most of) the HEAD feature specifications of the head daughter(s) in a tree. That is, the HFC gives content to the metagrammatical variable H while the FFP does not. Many of these HEAD feature specifications are inherited on the mother from the ID rule, yet they must still be identical on the head(s).

A projection meets the FFP on a rule if the instantiated FOOT features on the mother category are the unification of the instantiated FOOT features on its daughters. Unification is similar to set union, with the additional requirement that the result must be a category (that is, a function). The unification of a set of categories is the smallest category that contains all of the feature specifications of the categories in the set, if such a category exists. If two or more of the categories in the set conflict in their specifications for some feature, then the unification (a category) of that set is undefined (i.e., it does not exist). So, what the FFP requires is that any FOOT feature that is instantiated on the mother category must also be instantiated on at least one of its daughters and that any FOOT feature instantiated on a daughter is also instantiated on the mother.

Intuitively, the FFP insures that certain information, namely information about SLASH, WH and RE, does not get lost in a tree. Informally, we can think of

the FFP as governing the passing of instantiated FOOT features both up and down the tree between daughters and mothers (though strictly speaking the UFI principles do not actively pass information around a tree, but rather filter out trees which do not meet certain conditions). It does not permit a trail of instantiated FOOT features to terminate partway up or partway down a global tree. However, because it disregards any *inherited* FOOT features, the FFP does allow the usual 'passing' of a FOOT feature through a tree to be blocked by mentioning the feature in the appropriate ID rule. Let us now consider how this comes about.

2.4.1.2 'Exceptions' to the FFP. The inherited FOOT features in a tree come from the 'top-linking' and 'bottom-linking' rules. These are ID rules that either introduce the FOOT feature (top-linking) or 'cash-out' that feature in terms of whatever (morphological/phonological) effect it is going to have (bottom-linking). They mark the highest and lowest nodes in the tree at which the FOOT feature must occur. They are what would be exceptions to the FFP if it were formulated in terms of FOOT features and not instantiated FOOT features.

Consider the top-linking rule in (14), the 'topicalization' ID rule.

14) $S \rightarrow X^2, H/X^2$

This rule licenses the occurrence of a SLASH and will indirectly cause a SLASH feature to be instantiated in the local tree below. But it does not cause instantiation of a SLASH feature on its own mother. That is, the tree in (15a) meets the FFP on rule (14) but (15b) does not, as there would be an instantiated FOOT feature on the mother and no corresponding *instantiated* FOOT feature on any daughter.



Clearly (15a) is what is needed for top-linking in a topicalized clause. The whole clause is a complete S, but within it there is an X^2 'filler' for an X^2 gap in the S. In some sense then, the [SLASH X^2] in (15a) is licensed by the other occurrence of X^2 in the rule. (The equivalence between the antecedent phrase and the value of the SLASH is supposed to be guaranteed by the CAP, which is discussed below in 2.4.2, though see the discussion in Chapter Four also.)

The topicalization rule (14) is a typical top-linking rule for SLASH: it introduces both an antecedent phrase and a SLASH feature on its sister which will ultimately dominate the associated trace. But other top-

linking rules are also possible. For example, rule (16) handles 'tough-movement' constructions.

16) $A' \rightarrow H[42], V^2[INF]/NP[-NOM]$

Here the SLASH is not licensed by an occurrence of the same category in the rule, but rather by an adjective with [SUBCAT 42]. The understood antecedent for the empty category dominated by the SLASH category is the subject of the clause, not a sister to the SLASH category. This fact is also established by the CAP, which will be discussed below.¹⁶

The typical bottom-linking rule for SLASH introduces an empty element of the same category as the value of the SLASH. For example, consider (17):

17) $VP \rightarrow H[2], NP[+NULL]$

The FCR in (18) (= GKPS' FCR 19) will insure that the NP in a projection of (17) gets instantiated with a SLASH specification and therefore the VP mother must be instantiated with the same SLASH specification.

18) $[+NULL] \supset [SLASH]$

The SLASH can travel no further down in the tree, because the NP[+NULL] will be realized as the empty string at the lexical level. Thus, this rule marks the point at which the trail of instantiated SLASH features must stop.

¹⁶I will consider GKPS' analysis of the tough-constructions in more detail in Chapter 4.

Rule (17) does not count as an exception to the passing of FOOT feature information since there is no structure below the NP[+NULL] to pass the information into. However, the ID rule (19) (which is the output of the 'Slash Termination Metarule 2') is exceptional.

19) VP/NP --> H[40], VP[FIN]

Here, the SLASH specification on the mother node in the rule stops SLASH from being instantiated on the daughter node in the local trees that the rule projects.¹⁷ This is exceptional because the trail of SLASH features 'dies' in mid-tree and is not cashed out as an empty category.¹⁸ (See Gazdar, 1981; Chung & McCloskey, 1983; Sag, 1983; and Schachter, 1984, for arguments that the lack of SLASH in these constructions makes the appropriate predictions.)

¹⁷GKPS allow only one SLASH feature per category and allow SLASH to take only one value, but they note that this will undoubtedly have to be relaxed in order to account for multiple gap constructions (see Engdahl, 1980). If this were the case, then both the mother and daughter in a projection of (19) could be instantiated with another, different SLASH feature.

¹⁸It seems that this exceptional bottom-linking rule may be needed only in order to deal with the that-trace effects. Since subject-extraction is generally regarded as marked, it may not be inappropriate that GPSG needs extra rules in order to deal with it.

2.4.2 The Control Agreement Principle

2.4.2.1 Control and Control Features. The Control Agreement Principle (CAP) handles what is standardly thought of as 'agreement' (i.e., subject-verb or verb-object agreement in person, number, and gender) and also aspects of what is in transformational terms 'raising', 'equi-deletion', 'tough-movement' and (the top part of) long-distance movement. The CAP is based on Keenan's (1974) insight that 'functors may agree with nominal arguments'. This is taken to prohibit any agreement system it does not explicitly sanction. It is a universal and only says what can occur, not what must occur; the actual agreement facts in a particular language will be specified by parochial statements. To take an example from GKPS: they claim that, in English, only projections of A and V show morphological agreement¹⁹, and this can be enforced by the FCR in (20) (their FCR 12):

20) [AGR] $\supset$ [+V]

(Note that GKPS are binding together the presence of a syntactic agreement feature with the existence of

¹⁹They note that one might argue that the difference between a book and *a books is also agreement. They do not indicate what the parochial FCRs/FSDs would have to be like to restrict agreement to just determiners and projections of V and A.

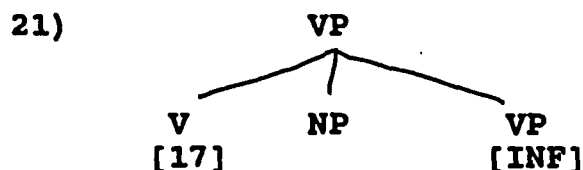
agreement morphology. This connection is not a logical necessity, as will be discussed in Chapter Four.) This is a fact about English and must be included somewhere in the grammar in order to block prepositions, for instance, from also agreeing with their objects (which they say appears to be the case in Welsh and Arosi).

Following Bach and Partee (1980), GKPS define a notion of 'control' which subsumes Keenan's notion of function-argument application. Control is a relation that holds between two categories in a local tree if they meet certain semantic criteria. There are two subcases of control: If there are only two daughters in a tree and they semantically combine as a function and an argument, then the argument controls the functor. (This is illustrated by subject-verb agreement, where the subject controls the verb phrase, and possibly determiner-noun concord, where the N-bar controls the determiner.) The functor (controllee) is a category of the appropriate semantic type²⁰ and contains a 'control' feature (which is inherited SLASH, if present, otherwise it is AGR), whose value must be equal to certain feature specifications on the controller (see below for discussion), which has the appropriate semantic type of

²⁰The 'appropriate semantic type' means, roughly, that its type is that of a function from the type of objects denoted by the controller to the type of objects denoted by the mother.

an argument for that functor (usually, though not always, an NP).

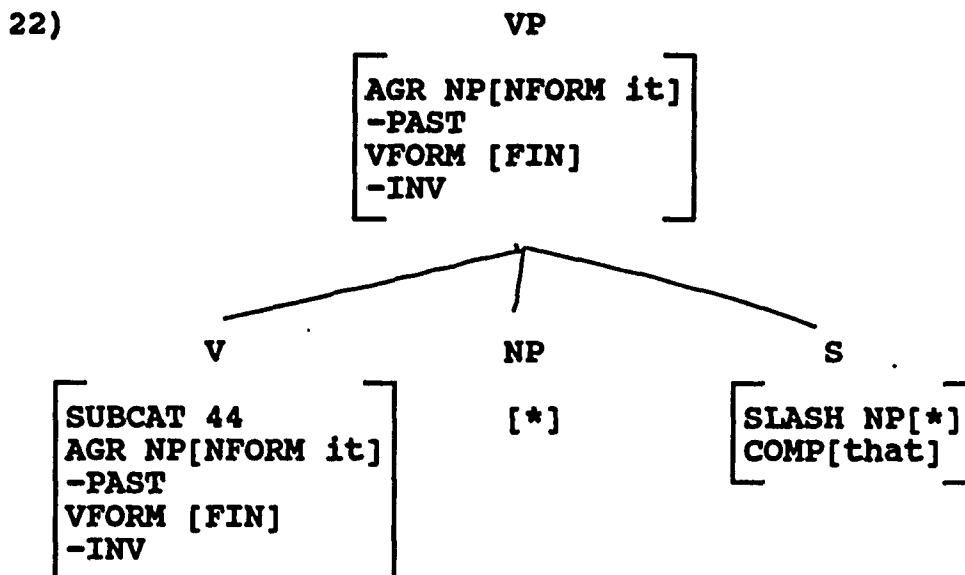
The second sort of control has to do with 'control predicates', that is, a lexical head subcategorized for complements that are predicative categories (where 'predicative category' is semantically defined as a function on NP denotations). Control predicates include auxiliary verbs, 'raising-to-object' verbs, object-controlled 'equi' verbs and adjectives, and 'tough' adjectives. In 'object-controlled equi' and 'raising-to-object' the lexical head mediates control between an NP complement and the predicative category. Here, the control predicate must semantically combine with the controllee and the controller, in that order, to yield a denotation that is of the appropriate semantic type for the mother. So in the tree in (21), the NP is the controller, the lower VP is the controllee, and the V is the control predicate.



This is a case of 'object control' (as with persuade); 'subject control' (e.g., promise) is handled differently.

The CAP states that if a category C has a controller, C', then the value of the control feature of C must equal C', or if C is a predicative category with

no controller, then the value of its control feature must equal the control feature of its mother.²¹ (This formulation of the CAP is inadequate and does not represent GKPS' final version. It will be refined below.) Consider the tree in (22), which is part of the tree underlying a sentence like (23). The features of the NP daughter (Kim in (23)) are not spelled out here but are abbreviated by *.



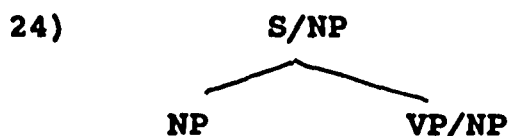
23) It is Kim that Sandy relies on.

The NP controls the S, where SLASH (since it is inherited) is the control feature. So the value of SLASH must have the same specifications as the NP does (again, this is an approximation that will be clarified below).

²¹Predicative categories will always have a control feature, either because SLASH was inherited from the ID rule or because of the lexical specification of the head of that category.

This pattern of agreement is appropriate since semantically the NP is the antecedent for the gap in the S.

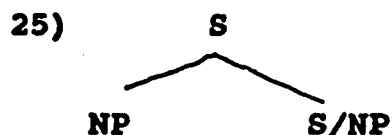
The presence of instantiated FOOT features complicates the definition of control by changing the semantic type of some functors and arguments. The NP in (24) must still control the VP although the instantiation of SLASH has changed the semantic type of the VP from a function from NP-type objects to S-type objects into a function from NP-type objects to VP-type objects.



The VP is no longer of the appropriate semantic type for the definition of control as it is now a function to VP-type objects, not S-type objects. Because of this, it must be stipulated that only the specifications for non-FOOT HEAD features (that is, all the HEAD features other than SLASH) and inherited (not instantiated) FOOT features determine the semantic type relevant for control.²² Inherited FOOT features, on the other hand, do help determine the semantic type relevant for control.

²²GKPS clearly mean for this set to include all inherited FOOT features including SLASH, even though it seems that this definition would exclude all SLASHes, as feature which is both HEAD and FOOT. I will assume that this is just an oversight, and that they are referring to inherited SLASH as well as all other HEAD features.

Consider a top-linking configuration, as in (15a) above, repeated here for convenience.

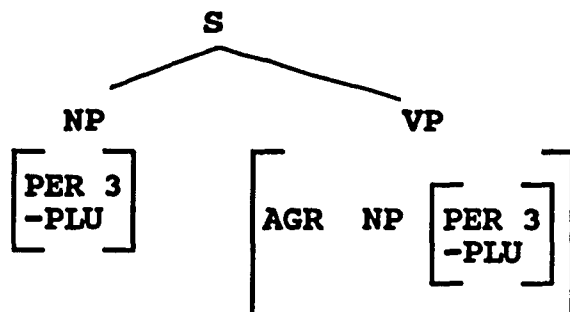


S/NP has the semantic type of a function from NP-type objects to S-type objects. Therefore, the S/NP and the NP in (25) can successfully combine semantically to form an S, whereas a VP/NP and an NP as in (24) cannot semantically combine to form an S/NP, they must form a VP semantically (even though they form a S/NP syntactically). That is why only the instantiated FOOT features are permitted to alter the semantic type of a category without affecting the determination of which categories undergo control.

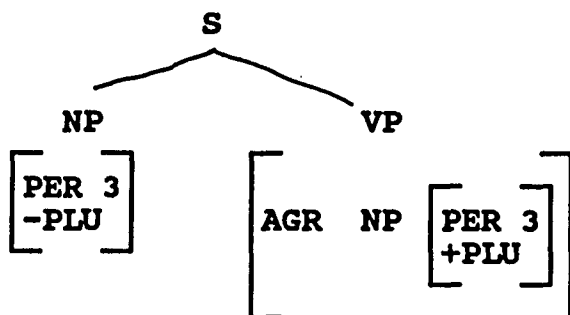
GKPS use the term 'chi-specifications' to denote the features relevant to determining where a control relation holds (i.e., non-FOOT HEAD and inherited FOOT features). (Chapter Four will return to the chi-specifications and examine the validity of the generalization that they express.) It is also the chi-specifications that the CAP forces to be identical on the controller and in the control feature on the controllee. (Again, this claim will be critically examined in Chapter Four.) For the present, we will now turn to an examination of how the CAP functions.

2.4.2.2 The CAP at Work. To illustrate the functioning of the CAP, consider the trees in (26) (= GKPS 24 and 25, p. 90).

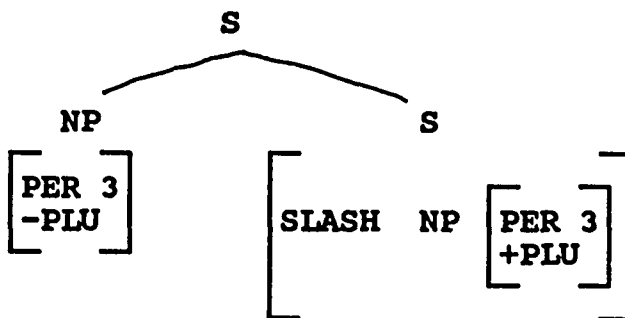
26) a.



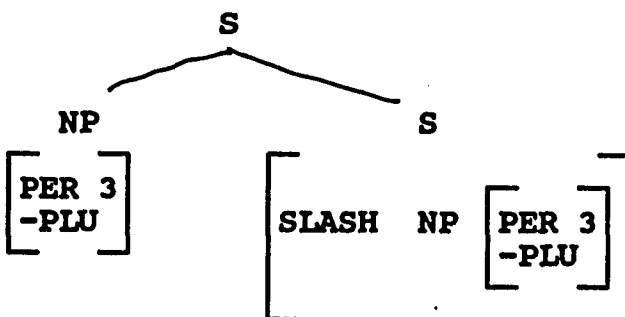
b.



c.



d.



(26a and b) are projections of the standard rewrite rule for S's ($S \rightarrow X^2, H[-SUBJ]$) and (26c and d) are projections of the topicalization top-linking rule in (14) above. In (a) and (b) the NP controls the VP, and AGR is the control feature on the VP. (26a) meets the CAP on the relevant rule because the value of the control feature equals the chi-specifications of the NP.

However, (26b) does not meet the CAP since the control feature does not match the chi-specifications of the controller. Turning now to (26c and d), the NP controls the daughter S, since they combine semantically as function (S/NP) and argument (NP) to form an S, and SLASH is the control feature on that S (it is inherited from the ID rule). (26c) fails to meet the CAP on (14) as the control feature does not match the chi-specifications on the controller, but (26d) does meet the CAP on (14).

It might seem as if requiring that all non-FOOT HEAD features on the controller be included in the control feature on the controllee is unnecessary, in English and also universally. Consider, for example, subject-verb agreement. The CAP requires that all of the non-FOOT HEAD features on the subject NP also be in the specification of AGR on the VP. Since SUBCAT is a HEAD feature, this seems to imply that agreement could be sensitive to the SUBCAT value of the noun. However, the

FCR in (27) (= GKPS FCR 9) will not allow the SUBCAT feature to appear on the subject NP; SUBCAT can only appear on BAR 0 categories.

27) [BAR 2] > ~[SUBCAT]

Therefore it cannot affect the process of agreement between an NP and a VP. Many of the FCRs and FSDs function like this, to restrict the features that can occur on a controller or controllee, although they were not initially proposed to do this. In this way, they constrain the features that the CAP can check.²³

The CAP based on chi-specifications works to a large extent (Chapter Four examines the exact working of the CAP in much more detail), but it predicts systems of agreement that we just do not find in the world's languages. GKPS cannot use a more restricted definition of chi-specifications because they are using a single principle (the CAP) to handle several phenomena. The CAP is responsible not only for standard subject-verb and verb-object agreement, but also subject and object control, top-linking, the distribution of 'dummy' NP's (there and it) and also sentential subjects. These various constructions are sensitive to the presence or

²³GKPS will need more FCRs and FSDs in order to do this completely, but for other reasons they need more than are presented in their book, e.g. FCRs to block features like LOC and ADV in irrelevant places.

absence of *different* features. For example, agreement between an NP and its VP sister requires only knowledge of the PER and PLU features of the NP (and GENDER, which GKPS do not include). Those would then be in the AGR feature of the VP. But the rule GKPS use to expand S allows any BAR 2 category to be the subject. It is the CAP's job to ensure that each verb gets the right kind of subject. Therefore, it must be sensitive not only to the major category features N and V, and to PER and PLU, but also to SUBJ (to differentiate VPs from Ss) and LOC (for 'locative inversion' sentences) and possibly others.²⁴

Top-linking requires information about BAR, LOC, N, NFORM, PER, PFORM, PLU, PRD, SUBJ, V, and VFORM. Thus top-linking uses a more complete subset of the HEAD features, but it is still a subset. In fact, it seems that none of the phenomena that the CAP is responsible for requires knowledge of all of the non-FOOT HEAD features. This suggests that the CAP is in fact expressing a spurious generalization. (I will return to this point in Chapter Four.)

²⁴All of these non-NP subjects, which require the inclusion of the features other than PER, PLU and GEND, show morphological agreement exactly like third person singular NPs. This is either a missed generalization (a default of some kind) or a strange coincidence.

2.4.3 The Head Feature Convention

2.4.3.1 The Normal Case. Let us now turn to the Head Feature Convention (HFC). The HFC is concerned only with head daughters and their mothers. But not all ID rules introduce only one head; for example, conjoined structures have multiple heads. This causes some complications in the definition of the HFC, which I will ignore for the moment.

It is the job of the HFC to equate HEAD features between mothers and head daughters. Simple equality will not work, however, since the mother (or daughter) may for other reasons be required to carry (or to lack) feature specifications that are incompatible with the daughter (or mother). These 'problematic' features may be inherited from the ID rule or required by the FCRs, the FFP or the CAP. GKPS claim (p. 95) that when the HFC conflicts with any of the other principles of UFI, it acquiesces to them.²⁵ Thus it functions not as an absolute principle, but as a default. This could seem to be a problem, as Shieber (1986) points out, since the CAP is defined partially in terms of instantiated HEAD features, and now we see that the HFC is defined

²⁵It is not clear that GKPS' claim is completely accurate. This question will be examined in more detail in Chapter Three.

partially in terms of trees that meet the CAP. However, this should never really be a problem as the HFC and CAP can never interact in a pernicious way. (This will be discussed more below, along with the notion of simultaneous application of these principles.)

What the HFC must capture is the fact that if the head or mother is independently required to have (or lack) a particular HEAD feature specification, and that specification is incompatible with the feature specifications on the mother or head respectively, then there need not be equality between those HEAD feature specifications; otherwise there must be equality. The case where it is easiest to see this kind of conflict is with the HEAD feature BAR. An NP will be specified as [BAR 2], but its head daughter will inherit [BAR 1] from the ID rule, and the FCRs will force the lexical head of that N-bar to be specified as [BAR 0], not [BAR 1], because the lexical head is specified for SUBCAT. If the HFC insisted on equal specifications here, trees with NP's dominating N-bars dominating lexical N's would never be generable (nor indeed any X'' dominating X' or X' dominating X⁰).

As another example, consider the HEAD feature AGR. According to GKPS, AGR will be lexically specified on V⁰ and must also occur on its VP mother. The HFC will insure equivalence between them. The HFC will also

ensure that S and VP have the same AGR specification. However, this seems incorrect. Allowing AGR to be specified on S's would predict that a lexical item could subcategorize for a complement clause with a third person singular subject, for example, or that the choice of complementizer could co-vary with agreement features of the subject, as it can with the tense of a complement clause. But this does not seem to happen in natural languages. Its impossibility could easily be established by adding to the GKPS system an FCR which blocks AGR from occurring on categories specified as [+SUBJ] (i.e., sentences). If this is right, then the HFC must be formulated to accommodate this kind of discrepancy in HEAD features while in general forcing mothers and daughters to have equivalent HEAD features (including BAR and AGR).

2.4.3.2 'Free' Feature Specifications. In order to allow special cases such as these, GKPS define the notion 'free' feature specifications. A feature specification is 'free' on some category C if it can appear on legal extensions (see section 2.2.4.1) of that category: a feature specification that conflicts with specifications that are required to be on the category (regardless of whether they are required by inheritance, the FFP, CAP or FCRs) is not free on that category. To

continue with the first example above, BAR is not free on an NP, since it is inherited from the ID rule, and it is not free on the N-bar head of NP since that specification is also inherited. The HFC will permit this situation if it is formulated so as to require that the HEAD feature specifications on the head be an extension of the intersection of the HEAD feature specifications on the mother and the free HEAD feature specifications on the head, and that the HEAD feature specifications on the mother be an extension of the intersection of the HEAD feature specifications on the head and the free HEAD feature specifications on the mother.

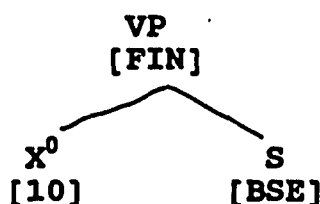
This formulation is successful for the cases where an ID rule introduces only one head, but it must be modified to handle trees with multiple heads. We will discuss this case only briefly, since it is not central to the topics addressed in the chapters that follow. To deal with the case of multiple heads, GKPS refine the definition of the HFC. They define it on the set of projections that meet the FFP and CAP on some rule and it says that: i) the HEAD feature specifications of each head must be an extension of the HEAD feature specifications of the category created by taking the intersection of the mother with the free feature specifications on that head and ii) the HEAD feature specifications on the mother must be an extension of the

HEAD features of the category created by taking the intersection of the intersection of each head with the free feature specifications on the mother.²⁶ What all this means is that the HEAD features on the mother and the head daughter(s) must be equal, up to the limitations imposed by the FFP, the CAP, and the FCRs.

2.4.3.3 The HFC at Work. One of the illustrations that GKPS use for the HFC is the rule in (28)²⁷ (= their 50) and the trees in (29) (= their 52 - 54).

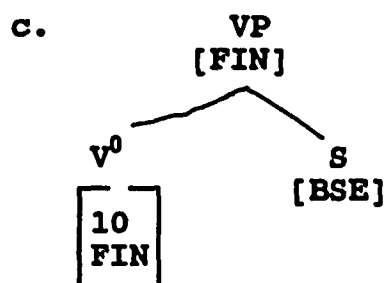
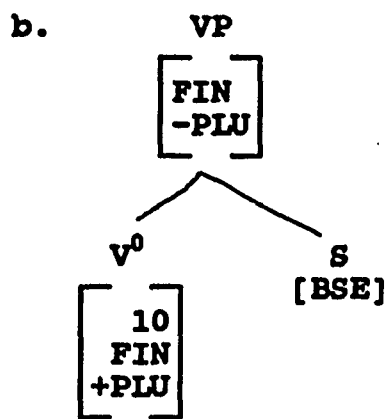
28) VP --> H[10], S[BSE]

29) a.



²⁶There is an additional clause which states that any projection that introduces no heads automatically satisfies the HFC. However, GKPS do not discuss any non-headed rules and this clause will be irrelevant to the discussion below.

²⁷The rule in (29) introduces verbs like 'insist' which subcategorize for a complement in the base form (cf. I insist that he leave now.).



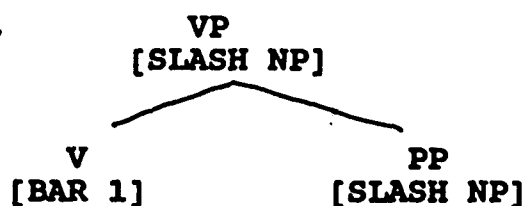
GKPS point out that both (29a) and (29b) fail to meet the HFC on rule (28). The set of HEAD features on the mother category in (29a) that are free with respect to the head daughter is $\{ \langle N, - \rangle, \langle V, + \rangle, \langle VFORM, FIN \rangle \}$. As noted above, $\langle BAR, 2 \rangle$, though a HEAD feature and present on the mother (implicitly in the VP label), is not a free HEAD feature here, since FCRs force the lexical head to contain $\langle BAR, 0 \rangle$. (A category is lexical if it is defined for SUBCAT.) Example (29a) fails to meet the HFC on rule (28) since none of the free HEAD features on the mother have been instantiated on the head daughter; it is unspecified for N, V, and VFORM. (29b), however, fails because the set of free HEAD features on the mother

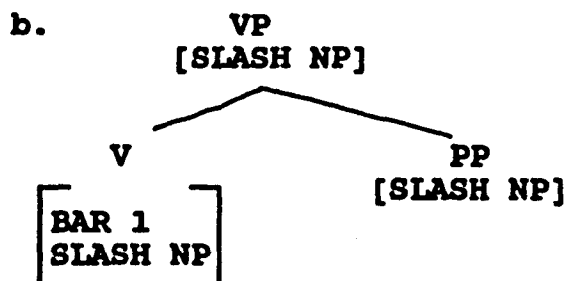
category is {<N, ->, <V, +>, <VFORM, FIN>, <PLU, ->} but the instantiated free HEAD features on the head daughter are {<N, ->, <V, +>, <VFORM, FIN>, <PLU, +>}. (29b) fails to meet the HFC with respect to rule (28) because although these two sets intersect, neither is an extension of the other. The tree in (29c) does, however, meet the HFC on rule (28) as the free HEAD features on the mother and head daughter are the same.

2.4.3.4 'Exceptions' to the HFC. We saw above that what appear to be exceptions to the FFP result from including FOOT features in the ID rules. A feature in a rule is inherited into a tree, and inherited FOOT features are not subject to the FFP. It is possible to sidestep the HFC in a similar manner. For example, the rule in (30) (which would give 'extraction' out of an adjunct) would admit the tree in (31a) but not the tree in (31b) in which SLASH has passed down to the head daughter.

30) VP/NP --> V', PP/NP

31) a.





SLASH is not in the set of free HEAD features on V' since 'free' is defined only on projections that meet the FFP and CAP. Instantiating SLASH on V' would violate the FFP (there is no corresponding instantiated SLASH on the mother) and therefore SLASH is not free on V'. SLASH is also not free on the mother category as it is inherited from the ID rule. Since SLASH is not free on the mother or the head daughter, the HFC does not require identity on that feature specification and we get what appears to be a violation of the HFC.

Note that it is the SLASH features in rule (30) that are responsible for the apparent violation of the HFC; if they were omitted from the rule, it would admit the tree in (31b) and not the tree in (31a). So, we can force apparent violations of the HFC and the FFP by including certain features in the ID rules. There are also apparent violations when the HFC runs up against the requirements of the FCRs. In addition, there can be apparent violations of the HFC in order to satisfy the CAP and the FFP. This difference between the HFC and the FFP is thus one of quantity (how many other principles

can override it) as opposed to quality (whether or not it can be overridden). This will be discussed more in the next chapter.

2.5 Feature Specification Defaults

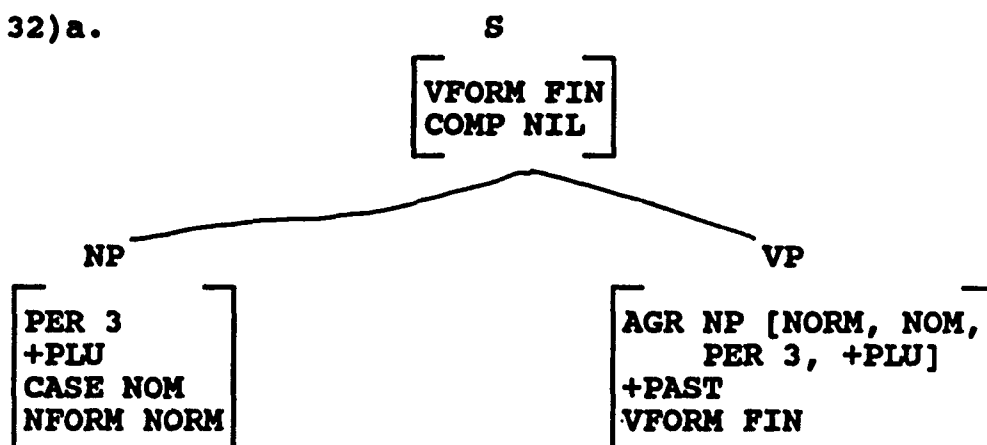
2.5.1 Local Admissibility

We have seen now how the ID/LP format for rule writing works, how the principles of feature instantiation apply and interact, and the role of FCRs. The only component we have left to consider is the FSDs. FSDs come into play in the grammar in the mapping from ID rules to local trees. Central to this mapping is the notion of 'local admissibility'. Local admissibility invokes the CAP, FFP, HFC and also the LP statements, FCRs and FSDs. Before we can see how FSDs function, we will need the notion of 'candidate projection' of a rule, which in turn requires the notion of 'LP-acceptability'. GKPS define a local tree as LP-acceptable if and only if, if a category C_i precedes a category C_j in the tree, then there are no categories C_i' and C_j' , where C_i is an extension of C_i' and C_j is an extension of C_j' , and $C_j' < C_i'$ is an LP-statement in the language. That is, the linear order of the daughters in the tree does not violate any of the LP statements in the grammar. A projection from a rule r is a 'candidate projection' of

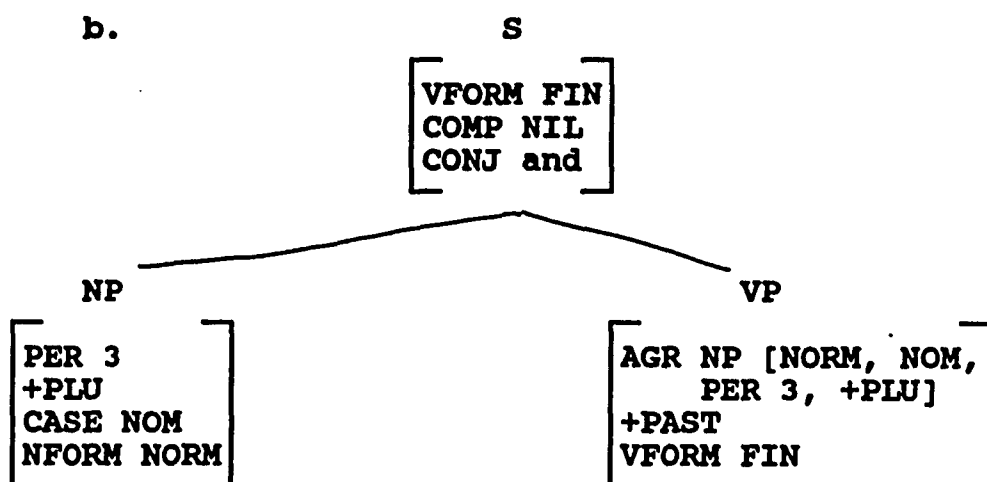
that rule if and only if it is LP-acceptable and it meets the HFC, the FFP and the CAP on r . This is all fairly straightforward. The situation, however, becomes more complicated when we now invoke the FSDs.

If a feature f in a category C has the same value in all candidate projections of r , then that must be because that value has been fixed by the ID rule or some principle of UFI (the HFC, FFP or CAP). C is then 'privileged' with respect to the FSDs that affect f and does not need to have its default value. (That is, default values may be overridden by all rules and principles in the grammar.) But, if f varies independently of any other feature in the candidate projections (that is, it is not inherited, nor is its value governed by UFI), then the grammar excludes those trees where it does not take the default value. GKPS use the device of comparing all candidate projections to formalize the idea that defaults should hold only if nothing else in the grammar requires a non-default value. For example, CONJ must be absent from a tree unless it is inherited from an ID rule; the reason is that it is possible for CONJ to be absent (it is a non-percolating feature, so UFI does not govern its occurrence), and its default is $\sim$ [CONJ]. So, the tree in (32a) meets this FSD on the rule in (33), but (32b) does not.

32)a.



b.

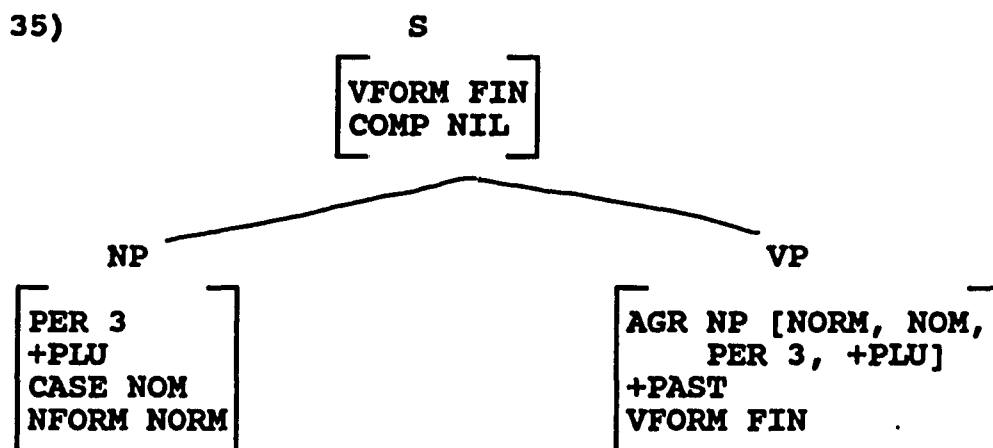
33) $S \rightarrow X^2, H[-SUBJ]$

However, the rule schema in (34) does permit expansion of a CONJ-marked S, because the CONJ feature is inherited.

34) $X \rightarrow H[CONJ \alpha_0], H[CONJ \alpha_1]$
 where α is in (<both, and>, <either, or>,
<NIL, but>)

Instances where the principles of UFI cause a feature to have a certain value, for example because it must co-vary with another category in a local tree, are also 'privileged' with respect to the defaults (but see

the caveat below regarding the HFC and the discussion in Chapter Three). For example, the default case for NPs is [ACC]. But this default does not apply to the daughter NP in the tree in (35) since an FCR forces all projections of finite Vs that agree with NPs to be specified as [AGR NP[NOM]] and the HFC then forces the VP also to be specified [AGR NP[NOM]].



GKPS claim that the CAP then insures that its NP sister is specified as [CASE NOM]. (I will return to this point in Chapter Four.) Since CASE has a value required by some principle of UFI (actually, the interaction of several principles), the default cannot apply. As these two examples show, FSDs must apply to all categories in all trees, whether those categories are mothers or daughters.

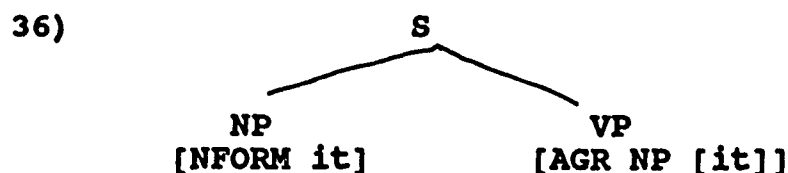
A candidate projection of a rule *r* is an 'admissible projection' of *r* if and only if it also meets the requirements of the default specifications. This

means, roughly, that for all categories in r and all defaults,

(i) the default is true of the category, or (ii) no candidate projections exist in which the default is true of the category, or (iii) the default is false of the category, but making it true would necessitate changing (in order to avoid violating UFI, TLM) some other category in the local tree (in particular, changing a sister if the category is a lexical daughter) (GKPS, p. 103)

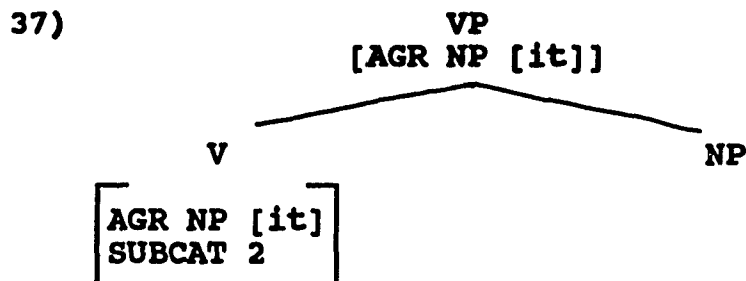
The last clause concerning lexical categories is a complication which is needed to avoid having all lexical heads be exempt from defaults on their HEAD features, which are either specified by rule or co-vary with their mother. I will now illustrate how this clause works.

The definition of 'admissible projection' makes lexical categories exempt from FSDs only if they co-vary with a sister category, though non-lexical categories are exempt if they co-vary with any other category in the local tree (either their sister or their mother). Consider the tree in (36), which is locally admissible from the rule in (33) above.



The NP in this tree is lexical (in that it is preterminal; it will dominate the lexical item it) and it does not have the default value for NFORM (which is NFORM NORM). This is allowed by clause (iii) of the definition

of local admissibility since its value for NFORM co-varies (via the CAP) with its sister VP's value for AGR (which has been freely instantiated). The tree in (37) is not locally admissible from rule (7b), repeated here as (38).²⁸



38) VP --> H[2], NP

The value for AGR on the mother has been freely instantiated and the HFC requires that it also be on the head daughter. But this is a non-default value for this lexical category and this tree is not an admissible projection of this rule since the lexical item co-varies with its mother and not with its sister.²⁹ Compare this with (39), which is an admissible projection from (40),

²⁸(38) is one of the local trees that would underlie a sentence like It hit John, where it is the expletive, and not the referential, pronoun. Expletive pronouns can only occur if they are licensed by a sister VP (or V). This will be discussed in more detail in Chapter Four.

²⁹For a tree to be admissible with a mother VP specified with [AGR NP[it]], it must inherit that specification from the ID rule which licenses it. AGR on the head daughter will be specified identically, due to the HFC. This allows the insertion of a verb that takes it subjects and allows the generation of sentences like 'It is raining'.

even though the default is for ADV to only occur on BAR 0 categories.

39) AP
 [+ADV]
 |
 A'
 [+ADV]

40) AP --> (AD[+ADV]), H'

Clause (iii) of the definition of admissible projection allows this because this co-variation is between a mother and its non-lexical head daughter. Therefore, they can both have the non-default value for ADV (due to the operation of the HFC). What all this boils down to, effectively, is that the HFC can override FSDs that apply to non-lexical categories, as can the FFP and the CAP, but the HFC cannot override FSDs that apply to lexical categories, only the CAP and FFP can.

If non-default values were allowed when mothers and head daughters co-varied on HEAD features, then there would be no way to force the defaults relevant to those HEAD features to apply. The grammar would then generate the (a) sentences in (41) and (42).

41) a. *John eaten the cake.
 b. The cake was eaten.

42) a. *There grew a riot.
 b. There ensued a riot.

On the other hand, if the grammar could not ignore the FSDs relevant to HEAD features when there is co-variation

between two daughters, then there would be no way to generate the (b) examples in (41) and (42). Thus the insight that it must be possible, in general, to override the FSDs, while mothers and head daughters cannot freely license each other to violate the FSDs seems valid. Chapter Three will return to this and question whether the GKPS implementation accurately depicts this insight.

2.5.2 Lexical Insertion

Before we can move on to a discussion of what it means for a non-local tree to be admissible we must briefly consider lexical insertion. GKPS assume a lexicon with entries that contain (at least) four kinds of information: 1) a phonological form, 2) a syntactic category, 3) an indication of any irregular morphology, and 4) a meaning. These lexical items constitute the terminal symbols of a GPSG sentence derivation. GKPS assume further that the output of the lexicon is a set of unary branching trees like (43) where the mother is a syntactic category that extends the category label in the lexical item and the daughter is the lexical item.

43)

+V
-N
SUBCAT 1
BAR 0
VFORM BSE

|

die

Such a tree is considered to be 'terminated' because 'every leaf node is labeled by a terminal symbol' (GKPS, p. 104). All terminated local subtrees are generated by the lexicon and are assumed to be available to the syntax.

2.5.3 Non-local Trees

The set of trees that are locally admissible from the ID rules, supplemented with the terminated trees generated by the lexicon, is then the total set of local trees that the grammar can work with. Admissibility for non-local trees is defined as follows:

Let R be a set of ID rules. Then a tree t is *admissible* from R if and only if

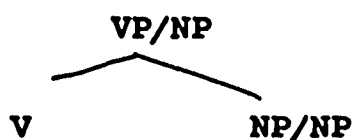
- (i) t is terminated, and
- (ii) every local subtree in t is either terminated or locally admissible from some r (GKPS, p.104)

For a sentence to be generated by a GPSG means that it must have a structural description that is an admissible tree from the ID rules in the grammar. This in turn means that every local subtree above the lexical level must simultaneously meet the FFP, the CAP, and the HFC on some

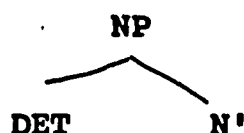
rule, be LP-acceptable and meet the requirements of the FCRs and FSDs.

GKPS never make explicit how the grammar is to build up non-local trees. The view that they seem to hold is that it is like lexical insertion. A category node can dominate a lexical item (or equally, a lexical item can be inserted under a node) just in case that node extends the category label in the lexical item. Similarly, two trees can join just in case a daughter of one tree is an extension of the mother of the other tree. However, this view is inappropriate. Consider the trees in (44).

44) a.



b.



On the approach that GKPS seem to take, these trees should be able to go together since the NP daughter in (44a) is an extension of the mother in (44b). But, this kind of loss of information cannot be allowed. It seems that local trees can only be allowed to go together to form a non-local tree if the mother of one tree is identical to a daughter in the other tree. These trees will usually be fully specified.

Trees in which categories are not fully specified can be locally admissible, but they would (almost) never be

a part of an admissible non-local tree as they could not be terminated.³⁰ GKPS assume that most items are fully specified for syntactic features in the lexicon. This assumption, taken together with the HFC and the FFP, insures that nodes in trees will, in general, be fully specified. That is, if a lexical N is specified as {+N, -V, PER 3, -PLU, +MASC, WHMOR[Q]}, then its mother NP must be specified identically, or the HFC and the FFP will be violated. Thus, lexical information will 'percolate up' through the tree and fully specified lexical entries entail fully specified nodes in trees, in general.

It is unclear just which features GKPS assume a lexical item can or must be specified for. Some of the FCRs and FSDs could act to fill in certain redundant information in lexical items. For example, given the universal FCR given above in (9), repeated here for convenience, the fact that the lexical item given above in (43) is <+V, -N> is predictable from the fact that it is specified for [VFORM].

³⁰This is not true of trees containing conjunction. Sag, Wasow, Gazdar and Weisler (1985) have argued that to account for sentences like His father was well known to the police and a devout catholic (where an AP is conjoined to an NP), the mother node must be underspecified. This node will be specified simply as [BAR 2, +PRD]. Categories can be conjoined under this node just in case their intersection is an extension of [BAR 2, +PRD]. This can only occur because he is subcategorized for an XP complement that is +PRD; in most constructions a category must be fully specified.

45) [VFORM] $\supset$ [+V, -N]

Looking at the few lexical entries that GKPS do supply, it appears that they are assuming that (at least some) lexical items can be marked for: N, V, PER, PLU, NFORM, AGR, PAST, AUX, INV, PFORM, VFORM, LOC, ADV, BAR, SUBCAT, WH, RE, WHMOR, REMOR, POSS, NULL, NEG, GER, CONJ, and CASE. That is, it appears that all the proposed features, except COMP and SLASH (which can only appear on phrasal categories and never appear on lexical categories), must be used in the lexicon at some time. Thus, a tree from the lexicon will be fully specified for all features (possibly by the operation of the FSDs and FCRs). Any other tree that this tree can hook up with will then also have to be fully specified. In this way it is guaranteed that the admissible non-local trees will contain only local subtrees that are fully specified.

CHAPTER THREE

SOME PROBLEMS WITH FEATURE INSTANTIATION

GKPS is impressive both in the detail of the various analyses and in the extent to which the proposed analyses account for the constructions of English. However, as Shieber (1986) points out, at least some of the complexity of GKPS' theory is a consequence of their formalisms:

Much care is taken in the definitions of the feature instantiation principles (and their ancillary notions such as controller, controllee, free features, privileged features, etc.) to control the complex interaction of the various constraints. For instance, the FFP admits local sets of nodes with . . . instantiation of slash features. . . . A simple definition of the CAP would ignore the source of the slash value, whether inherited, instantiated by the FFP, or instantiated in some other manner. However, the appropriate definition of control needed for the CAP must ignore instantiated slash features, but not inherited ones. . . . Thus, the CAP is in some sense blind to the work of the FFP. . . . The increased complexity of the principles resulting from this need for tracking the origins of feature values is evident not only in the CAP, but in the other principles as well. (pp. 3-4)

This chapter explores (and modifies) Shieber's proposal that the theory can be simplified by implementing an ordering statement.

3.1 Complexity

3.1.1 Complexity in Linguistics

As should be clear from the discussion in Chapter Two, the theory proposed by GKPS is quite complex. It is complex in two different senses. In one sense, it is complex in that the several components of the grammar interact in subtle ways to yield only the appropriate trees for a language. This is the 'deductive depth' that Chomsky (1982, 1986) argues is desirable, since it is the mark of a theory that is moving toward explanatory adequacy. As Chomsky says,

Because of the sometimes intricate connections among the various subtheories, small changes in the formulation of some principle or notion may have large-scale and wide-ranging consequences. . . . This is naturally a positive development, one that is inherent in any serious effort to deepen explanatory power (1986, p. 2)

The various principles and definitions that GKPS use are so interwoven that a small change in one can have repercussions in the predictions made throughout the entire theory.

The GKPS theory is also complex in a second sense in that some of its principles and definitions are quite intricate and complicated. This then seems to raise the

question: is all of this complexity necessary, or is some of it merely a sign that the correct principles have not yet been found? Or perhaps that they cannot be stated within this theoretical framework? Chomsky (1982) states,

Insofar as we succeed in finding unifying principles that are deeper, simpler and more natural, we can expect that the complexity of argument explaining why the facts are such-and-such will increase, as valid . . . generalizations and observations are reduced to more abstract principles. But this form of complexity is a positive merit of an explanatory theory, one to be valued and not to be regarded as a defect in it. . . . It is important to distinguish clearly between complexity of theory and complexity of argument, the latter tending to increase as theory becomes less complex in the intuitive sense.

(p. 15)

Complexity of argumentation or interaction, then, is a good achievement, while complexity which is inherent in the principles of grammar is not. In the rest of this chapter and the next I will explore whether or not some of the complexity in the GKPS system can be eliminated. That is, I will try to maintain the descriptive scope of the GKPS feature instantiation apparatus, while simplifying some of its principles and definitions.

3.1.2. Complexity in GPSG

3.1.2.1. Subcategorizational complexity. Some of the complexities that GKPS introduce are the result of their conception of subcategorization. As noted in

Chapter Two, GKPS' SUBCAT feature takes rule numbers as its values. The lexical head of a local tree must then contain a SUBCAT value that is identical to the number of the rule that induced that tree. While this version of subcategorization is not without its merits, it also has some serious flaws.

Jacobson (1987) points out three problems for the GKPS view of subcategorization. First, it forces a syntactic (metarule) analysis of constructions like Passive and Extraposition. These must be syntactic (and not lexical, as in LFG or Categorical Grammar) because, for GKPS, lexical items do not carry explicit information about what complements they are subcategorized for. Instead, they only carry a SUBCAT value and the ID rule which has the same value introduces the complements. Therefore, there is no way within the lexicon to refer, for example, to all and only those verbs which take an NP complement. Jacobson questions the use of metarules on two different grounds. First, GKPS must adopt the Lexical Head Constraint (which says that only lexical ID rules can be input to metarules), thus stipulating that metarules can only do what other theories do with lexical rules.¹ The second problem with metarules is their

¹If GKPS did not adopt the Lexical Head Constraint, then their use of metarules would allow their grammars to generate non-context-free languages (Gazdar and Pullum, 1982). In fact, metarules in GPSGs without

expressive power. It is not clear exactly what the formal power of a grammar with recursively applying metarules is (see Uszkoreit and Peters, 1985).

Therefore, given GKPS' commitment to stay within the bounds of devices whose mathematical properties are well-understood, they must constrain the application of metarules such that in a series of metarule applications, no metarule may apply more than once. Jacobson says, ". . . this is a rather unnatural constraint, which in no way follows from the formalism of metarules themselves; it is clearly preferable to dispense with metarules altogether." (p.397). But, to change the metarules into lexical rules, as Jacobson suggests, requires a more standard approach to subcategorization than that of GKPS.

The second problem that Jacobson notes for GKPS' theory of subcategorization, which requires their version of metarules, has to do with GKPS' position that the semantics of a phrase should be derivable from the semantic types of the mother and daughters and the syntax of the rule which induced that phrase. However, because the semantic type of a passive verb must differ from the semantic type of its associated active verb (in order to combine correctly with its arguments), GKPS must, in effect, postulate a lexical rule of Passive (which

the Lexical Head Constraint can induce grammars for any recursively enumerable set (Uszkoreit and Peters, 1985)

changes the semantics) as well as the Passive metarule (which is syntactic and changes the complement structure). GKPS cannot allow the metarule to adjust the semantic of the phrase without retreating from their principle that the semantics of a phrase should be derivable and not stipulated. Such a retreat would entail massive changes in their theory of the interaction of syntax and semantics. Thus, this duplication of work (having both a syntactic Passive metarule and a semantic Passive lexical rule) follows directly from their view of subcategorization and can be dispensed with if their approach to subcategorization is dispensed with. If lexical items explicitly contain information about their complement structure, then a lexical rule of Passive can generalize over transitive verbs (much as is done in LFG and GB).

The final problem that Jacobson discusses is that it is the use of SUBCAT that requires GKPS to complicate their notion of how FSDs are invoked. Recall that a feature on a given category C need not have its default value (the FSDs do not apply) if that value systematically co-varies with another category in its local tree (because systematic co-variation is due to the HFC, the FFP, the CAP or the FCRs). However, if C is lexical (BAR 0), and systematically co-varies with its mother, then the FSDs do apply to it. This complication

that Jacobson is objecting to arises largely due to the interaction of GKPS' notion of subcategorization and the use of metarules. Metarule application does not alter the SUBCAT feature, thus the passive verb kissed has the same SUBCAT value as the active kiss. Without the complication of the FSDs, GKPS have no way to avoid generating sentences like *Fred was kissed Bill (where the verb has the non-default value for passive because its mother has the same value). However, if passive verbs are marked in the lexicon with their own subcategorization features, then these sentences will not be generated and the FSDs may be invoked in a much simpler fashion, not distinguishing lexical from non-lexical categories.

There are also other problems for the GKPS version of SUBCAT. One prediction that is implicit in their theory is that metarules are exceptionless. However, it is clear that this is not the case. Consider again the Passive metarule. It says, roughly, that any verb which takes an NP complement has a passive counterpart without that NP complement. Yet there are well-known counterexamples to the Passive generalization (cf. want and get). Perhaps these can be excluded on semantic grounds, but there are other examples within GKPS which do not seem to be amenable to a semantic analysis. Consider for example the sentences in (1).

- 1)
 - a. John spared Bill a dime.
 - b. That Bill left bothered John.
 - c. There exists a problem in the homework.
 - d. It is Bill that Mary loves.
 - e. It is Kim Sandy relies on.

Given the GKPS analysis of passive, all of these sentences should have passive counterparts, but as is clear in (2), this is simply not the case.

- 2)
 - a. ?Bill was spared a dime (by John).
 - b. *John was bothered by that Bill left.
 - c. *A problem in the homework is existed (by there).
 - d. *Bill is been that Mary loves (by it).
 - e. *Kim is been Sandy relied on (by it).

These examples could easily be ruled out by a theory which allowed lexical items to be subcategorized independently for the input and output of metarules, which following Jacobson, we can take to be essentially lexical rules.

Most theories of subcategorization include some amount of redundancy between lexical specification and the phrase structure rules of the grammar (see Stowell, 1981 for an attempt to eliminate this redundancy in GB). Thus there is some tension in the theory over where to locate the redundancy, either in the syntax proper or in the lexicon. In their attempt to reduce the redundancy, GKPS kept their conception of the lexicon fairly simple and then were forced to incorporate certain redundancies in the syntax. For example, consider the rules in (3).

- 3) a. VP --> H[13], VP[INF] (tend)
 b. VP --> H[15], VP[INF, +NORM] (try)
 c. VP --> H[17], NP, VP[INF] (believe)
 d. VP --> H[18], NP, VP[INF, +NORM] (persuade)
 e. VP --> H[19], NP, VP[INF, +NORM] (promise)

Because of their view of subcategorization, GKPS must separate these five rules, even though they are very similar. A grammar that utilized a more standard version of subcategorization (eg., that of Aspects) would only need the rule in (4) to generate the same trees.

- 4) VP --> H, (NP), VP[INF]

The differences among verbs as to which verb could be inserted into trees with which complements would then be reflected in the verbs' subcategorization frames and not in the rules themselves. This seems especially reasonable in the case of (3d) and (3e), where the only difference is the way in which the verbs semantically combine with their complements. Syntactically, the structures seem to be exactly the same. The same point can be made with the verb perjure. Unless one can argue that it is a semantic requirement that perjure have a reflexive object, this requirement must be stated explicitly in the ID rule introducing perjure. Assuming that that is the case, then perjure will require a unique ID rule, one that only it is subcategorized for, as in (5).

5) VP --> H[74], NP[RE NP[RE MOR REFL]]

Obviously, it is undesirable to have to introduce a new ID rule for a single lexical item (even one as unusual as perjure) just because it has a slightly different subcategorization frame.

Given these problems with the GKPS approach to subcategorization, I will henceforth not assume such a theory, but will assume a more standard version of subcategorization (essentially that of Aspects), where it is an overt specification of the syntactic environments in which a lexical item may occur.

3.1.2.2 Complexity in UFI. Recall that GKPS had to complicate the definition of the HFC by defining 'free feature specification', which is itself an elaborate concept. The notion of 'free' restricts the HEAD features that the HFC can check and require to have identical values. This is taken to be unique to the HFC, but in fact it seems to be just a more intricate version of a kind of restriction that we find throughout the grammar, although it is generally written into the definition of a principle so it is not as obvious. Consider, for example, the distinction between inherited and instantiated FOOT features, which is an integral part of the GKPS definition of the FFP. This distinction restricts the features that the FFP checks and requires

to have identical values, just as the concept 'free' does for the HFC. Thus, the same restriction does not hold over all principles of the grammar, but the same *kind* of restriction does.

Almost every principle that can affect the feature structure of a category must apply only to certain features in certain circumstances or configurations, and not to the same features in slightly different circumstances or configurations. For example: the FFP can only check instantiated FOOT features, not inherited FOOT features; the CAP can only check non-FOOT HEAD and inherited FOOT features, not SLASH or instantiated FOOT features; the FSDs can only apply to certain features in certain categories (i.e., those features that are not 'privileged'). Only the FCRs are in effect in all categories in all trees. With the FFP and the CAP, the restriction to certain features is written into their definitions; whereas with the HFC and the FSDs, the restrictions on them are complicated and are stated independently of them (as 'free' and 'privileged', respectively). If some (or all) of the complexity of these restrictions can be eliminated then the theory as a whole will be simplified.²

²Of course, mere notational simplification is of little value. The goal here will be to reduce the theoretical apparatus as much as possible. For example,

3.2 Simplification Based on Ordering

Given that all these restrictions are doing the same kind of thing, one must wonder whether they could be combined and stated only once, or if they could be predicted somehow. Shieber (1986) has claimed that the latter is in fact the case, and that much of the complexity in the statement of the principles can be avoided if we move away from the simultaneous application of the principles of UFI and the FCRs, FSDs and LP statements. He argues that if instead these principles were sequentially ordered with respect to one another, the same consequences would result from a simpler formulation.

Shieber notes that the elaborate wording of the definitions in GKPS which restricts the principles from applying to the wrong features in the wrong configurations, induces an unexpressed ordering on the principles. To take one of his examples: trees with an instantiated SLASH feature must have the same value for SLASH instantiated on the mother and on a daughter in

reducing the number of lines it takes to write the HFC is of no theoretical interest if for example it merely exploits abbreviatory symbols, but if we could eliminate a special condition in the HFC or the whole HFC, that would represent a significant reduction. In Chapter Five, I will be examining a theory of the notational system used in the brain/mind, and then notational simplifications will also matter, at least for things that have to be learned.

order to meet the FFP. The CAP, as we saw above, is only sensitive to³ the presence of inherited, not instantiated SLASH. GKPS must complicate the definition of the CAP by mentioning which SLASH features are relevant and which are not. By ordering the CAP before the FFP, Shieber notes, we could simplify the definition of the CAP since we would not have to differentiate the two kinds of SLASHes; only inherited SLASHes would be present, not instantiated ones.⁴ This represents a real simplification of the grammar since the notions of inherited and instantiated will no longer be a necessary part of the definition of the CAP.

As another example, consider the application of FSD's. Recall that the definition of local admissibility was complicated by the fact that lexical categories are exempted from the FSDs only if they co-vary with a sister, but nonlexical categories are exempted if they co-vary with any other category in the local tree. (That is, the HFC cannot override some of the FSDs, but the

³'Sensitive to' means 'its application is (partially) determined by' these features.

⁴This analysis also relies on the idea that the principles actually introduce features into local trees. It is not clear that it would work on the GKPS view of checking features that have been freely instantiated, unless one were to break up free instantiation such that FOOT features were freely instantiated and checked, and then features relevant to the CAP were freely instantiated and checked. This point will be raised again below.

other principles of the grammar can override all of the FSDs.) Shieber proposes to split the FSD's into two groups, those that affect lexical categories and those that affect nonlexical categories. He then orders the lexical FSDs before the HFC and the nonlexical FSDs after the HFC. This allows for a simplification of the statement of how the FSD's apply. On his theory, HEAD features will not have been instantiated yet on lexical categories (the HFC has not yet applied), so if the HEAD feature specifications in lexical categories co-vary with some other category in a local tree, it must be with a sister.⁵ Of course, if a HEAD feature on a lexical daughter has a non-default value before the HFC applies, then that value must arise from the ID rules or the operation of the CAP or the FFP. In such a case, that feature is 'privileged' and the FSDs do not apply.

The ordering that Shieber finally arrives at is given in (6) (where >> means 'takes precedence over'⁶).

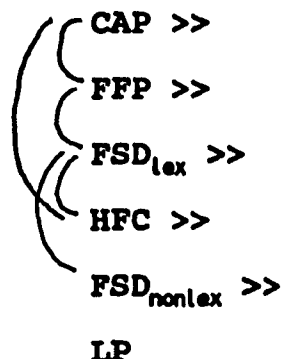
⁵It is not quite as clean as Shieber proposes. Consider the rule in (i), where the lexical head and its mother explicitly vary in the HEAD feature PLU.

(i) $V^2[+PLU] \rightarrow H^0[-PLU]$

This rule would circumvent the HFC and would not violate the FSDs since there seems to be no default for PLU. But surely it must not be a permissible rule. Perhaps its non-occurrence could be explained on the grounds that it is an unlearnable rule: what possible positive evidence could reveal to the child that the VP is plural while the head V is singular?

⁶It is somewhat unclear what Shieber means exactly by 'precedence'. Saying that the CAP 'takes

6) METARULES >>



Two points need to be made here. First, the output of the metarules is taken to be the total set of ID rules, both basic and derived. Second, Shieber motivates the appearance of metarules and LP statements in the ordering in the following way:

. . . all ID rules, both base and derived are subject to all these principles; yet metarule application is not contingent on instantiations of the base ID rules. Conversely, LP constraints are sensitive to the full range of instantiated features. (p. 6)

Therefore, metarules come first and LP statements come last, in the precedence ordering. (I have changed Shieber's mode of presentation in order to explicitly mark the orderings that he argues for, other than those

precedence over' the FFP might mean only that the CAP wins out in case the two conflict, or it might mean that the CAP applies *before* the FFP. In the latter case, the FFP might win out by *changing* the feature values assigned by the CAP; or the CAP might win out by establishing values which are *unchangeable* by later principles. In fact, Shieber appears to intend the latter interpretation.

involving metarules and LP statements. This will be useful later in this chapter.)

This precedence ordering allows the grammar to be simplified in terms of the record keeping that is otherwise necessary to control the interactions between components of the grammar. On the GKPS view, there are no stages of derivation. There is just a tree, with both inherited and freely instantiated feature specifications. These two kinds of feature specifications must be differentiated (for the FFP and FSDs at least). We could think of them (metaphorically) as written in different colored ink. The FSDs also need to know which feature specifications are required by the HFC and the CAP; and all the principles must be aware which feature specifications are required by the FCRs; so there must be more colors. It is as if each principle has its own colored ink and colors the features that it checks or requires to be present. Then other principles can be made sensitive to some features (some colors) and can ignore others. On Shieber's view, however, all of the feature specifications can be the same color (i.e., no color) and the order that the principles apply in takes care of the interactions. Earlier principles get a chance to determine a feature's value before later principles, which can refer to those values, but cannot

change them. Therefore, the record keeping is done automatically; it is free.

This is comparable to the use of rule ordering in early versions of transformational grammar. It does in fact allow a great deal of simplification in the statement of the principles of grammar (i.e., a real theoretical savings). It is also explanatory in that it makes sense of *why* the principles are sensitive to different features: a principle can only be sensitive to features present in the tree when that principle applies, and not to features instantiated by later principles. If the principles were unordered, then this sensitivity would be a complication. However, with ordering, it would be more complicated not to have the sensitivity. The ordering also predicts certain properties of the 'color sensitivity'. For example, it is transitive: if red is sensitive to pink and pink is sensitive to green, then red is sensitive to green.

However, it also represents a significant change in the view of the grammar from that proposed by GKPS.⁷

⁷Shieber's view is in fact very similar to the view which is assumed in Sag, Gazdar, Wasow & Weisler (1985). They also have much less elaborate and less formal definitions of the principles of UFI than GKPS, and they assume that the various principles actually instantiate the features, not just check them. In Sag, et al., and also in GKPS, there seems to be a trade-off between formality of discussion and precision in distinguishing whether features are instantiated freely

For GKPS, the principles of UFI check features that are present in trees and restrict the class of trees admitted by an ID rule if certain configurations of features do not hold. They do not add feature specifications, or copy them from category to category, but simply check ones which are there. If the kind of ordering that Shieber proposes is to work, one must assume that the principles are in fact supplying or writing in feature specifications, as is clear in the following passage. Shieber says that the definitions given by GKPS are such that they cause an

. . . implicit precedence ordering on the principles, principles earlier in the ordering being blind to the *instantiations from later principles*, which are themselves sensitive to (and exempt from applying to) *features instantiated by the earlier principles*.
[emphasis added; TLM] (p.5)

Obviously, this implies a sequential derivation of a local tree. GKPS are sometimes lax in their less formal discussions of the principles of UFI, and imply a similar conception of feature instantiation, as can be seen in the following quote:

The feature WH is a FOOT feature, as we have seen. Thus it will be *instantiated by the FFP* in exactly the same way as SLASH
[emphasis added; TLM] (GKPS, p. 155)

However, when they are being more precise (cf. p. 105), they clearly intend the principles of UFI to apply

or by the principles of UFI.

simultaneously, to trees which already contain all their inherited and freely instantiated features. They do not have the features being instantiated by the principles.

The change from GKPS' view to Shieber's view makes no real difference to how the local trees are then used to build up non-local trees. As we saw in Chapter Two, two local trees can join together to make a non-local tree just in case a daughter of one and the mother of the other are identical, and in the usual case a local tree will be able to be part of a complete non-local tree only if it is fully specified.

Shieber proposes a significant modification of the theory of GKPS, but his theory seems to be able to describe the same grammars and structures that the GKPS theory can. And, as we have noted, Shieber's proposal has advantages over GKPS. His theory is less complex (i.e., has less theoretical machinery) in that all of the restrictions that GKPS must either write into the principles or state separately, Shieber is able to do without; they follow from his precedence ordering.

This represents a real step forward for the theory, assuming that the ordering is innately specified. If it turned out that this precedence ordering could vary from grammar to grammar, then it would represent a big step backward as it would introduce a new range of

possible variation for grammars and grammar learners. (It would effectively make Shieber's version of GPSG unlearnable for the same reasons that the descriptively rich Standard Theory was unlearnable. See Fodor, in press.) This issue, however, can only be decided on the basis of detailed descriptions of many languages and that data is not available now. I will assume that Shieber's ordering is universally fixed, since there does not seem to be any *prima facie* evidence to the contrary, and follow him in assuming that the principles of UFI actually introduce features into the trees.

Shieber's proposal also represents an improvement on the explanatory level as well. It is easy to imagine how the GKPS system could be 'parameterized', similar to the parameterization of GB theory. For example, one can imagine a theory just like GKPS, only the child has to decide whether, for her language, the HFC affects all HEAD features (as GKPS propose), or only the inherited HEAD features and ignores the instantiated HEAD features.⁶ If this were the case, it would increase the number of choices that the child has to confront when acquiring a language. However, it is much harder to imagine how an ordered system, like Shieber's, could be

⁶Perhaps this is what Sells (1985) means when he says that the principles of UFI may be subject to finite variation across languages, as noted in Chapter Two.

subject to the same kind of variation. The stipulations of GKPS follow now from the ordering statement, and therefore are much less likely candidates for parametric variation. Therefore, an ordered system should present to the child fewer potential grammars to consider when acquiring a language.

Though in general a good move which we would like to be able to maintain, Shieber's proposal unfortunately does not work in full detail to solve all the problems. In particular, it has trouble with the CAP.⁹ Where the CAP is to apply is defined in terms of HEAD features that will be supplied by the HFC. But the HFC should be ordered after the CAP because (GKPS say) the HFC gives way to the CAP, when there is a conflict between them. (I will return to this claim below.) Thus there is an ordering paradox: the HFC needs to apply after the CAP, but the CAP cannot apply until the HFC supplies the features needed to define whether or not control holds.

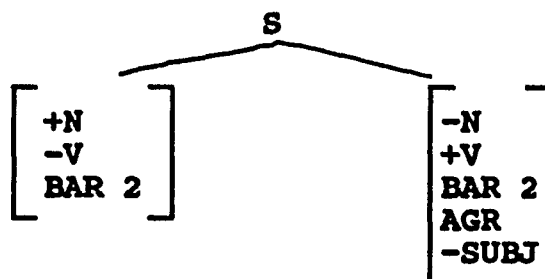
⁹Shieber also points out two other problems for his analysis. The first is how to implement the FCRs. It is unclear whether it would be better to revise the nature of the category structure that is used such that only categories that meet the FCRs are available, or to check the FCRs after each principle has applied, throwing out any trees which violate the FCRs. The latter is more complex than the GKPS proposal, but is probably more realistic for a psychological model, which is what Shieber is concerned with. Shieber also notes that the LP statements have not yet been incorporated into his system. He leaves them as a problem for future research and I will not address them here.

The HFC must apply both before and after the CAP.

Shieber suggests that possibly the HFC can be split into two parts, much like the FSDs, such that those HEAD features that are relevant to where the CAP is to apply (that is, relevant to semantic type determination, which has to do with the definition of control) are supplied before the CAP applies and the rest of the HEAD features are supplied after the CAP. He does not explicate this proposal any further, but this problem must be resolved in order to maintain the simplification that follows from his ordering.

Shieber notes that for this suggestion to work, the two subsets of HEAD features must be disjoint: there must be no overlap between the features which define where the CAP is to apply, and the features which the CAP instantiates. However, this simply is impossible. Not only must the features relevant for defining *where* control holds be instantiated before the CAP can apply, but also all of the features that are required to match. For example, in (7a), we have a tree licensed by the rule in (7b).

7) a.

b. $S \rightarrow XP, H[-SUBJ]$

Only those HEAD features necessary for the definition of control have been instantiated, and we can tell that the relationship of control holds here between the NP and its sister VP. But it is not enough to know that the AGR specification on VP must match the NP; we must also know (at least) the NUMBER and PERSON specifications of that NP in order for the CAP to copy them across to the VP. In sum, the HEAD features that must be specified before the CAP can apply are: N; V; PER; PLU; PFORM; SUBJ; and possibly BAR; LOC; and PRD. However, these are not just the features needed to specify the structures in which a control relationship holds, they are in fact the features that the CAP must apply to. Thus, instead of being two disjoint sets, they are identical.

Shieber's proposed solution to his ordering paradox will not work so we must find another solution to this problem. One way to start is by scrutinizing more closely Shieber's notion of 'precedence'. It can be argued that Shieber orders the CAP and the HFC backwards. That is, the HFC should precede, not follow, the CAP.

This becomes clear when one looks closely at his reason for ordering them as he does.

3.3 Ordering the HFC

Shieber says that the HFC must be sensitive to (that is, must have access to) feature specifications instantiated by the CAP, and hence the CAP takes precedence over the HFC. By this he means either that conflicts between the requirements of the HFC and the CAP are always resolved in favor of the CAP or that the HFC needs to copy AGR features that are supplied by the CAP. In this section, I will show that this assertion is inaccurate. Shieber is simply accepting GKPS' claim in the following passage, which Shieber quotes:¹⁰

Intuitively, the *free* feature specifications on a category is the set of all feature specifications which can legitimately appear on extensions of that category: feature specifications which conflict with what is already part of the category, either directly, or in virtue of the FCRs, FFP or CAP, are not free on that category. (GKPS p. 95)

GKPS offer no discussion of or motivation for including the CAP in this definition, they simply state it.

Jacobson (1987) in her review of GKPS also repeats this claim, discussing it briefly with respect to the HFC and FCRs, saying that this formulation of the HFC

¹⁰Recall that the HFC only checks HEAD features that are 'free' on a category (see Chapter Two for discussion).

is questionable at best. She says ". . . there is absolutely *no* independent motivation for this formulation of the HFC; it is used solely to solve theory-internal problems that arise strictly because SLASH is defined as a head feature." (p. 414). The problems that Jacobson notes with regard to taking SLASH as a HEAD feature are: 1) the fact that SLASH is never involved in agreement means that GKPS must say that only non-FOOT HEAD features are relevant to agreement (that is, SLASH does not have this important property of HEAD features); 2) GKPS must postulate an FCR (FCR 6) which stops SLASH from passing onto lexical heads and this is probably not independently motivated; 3) it makes the definition of the HFC more complex, in that it may be overridden by other principles of feature instantiation.

While I agree with Jacobson that the GKPS formulation of the HFC is faulty, I do not agree in points of detail. I will address her three issues in reverse order. First, it is not simply taking SLASH to be a HEAD feature that makes the HFC 'defaulty' in its nature. There are other places in which the HFC must give way to the requirements of the other principles (e.g., PAST must not be on the mother S even though it is on the head daughter VP, and BAR must be allowed to decrease down the projection path, even though it is a

HEAD feature). Therefore, the HFC will be 'defaulty' whether we take SLASH to be a HEAD feature or not. Second, GKPS' FCR 6 is unnecessary, given that a lexical category marked as having a phrasal gap in it is incoherent: since lexical categories are effectively the terminal level in a tree, they cannot have a phrasal gap in them.¹¹ Finally, we will see in the next chapter, that it is not necessarily the case that *all* HEAD features are involved in agreement processes. In fact, it seems that very few are, so the problem here could well be with GKPS' notion of 'agreement' and the CAP, and not with taking SLASH to be both a HEAD and a FOOT feature. In addition, it is exactly taking SLASH to be both HEAD and FOOT that provides for GPSG most of the Subjacency, ECP and CED effects of GB. Therefore, giving it up is a significant move that might be quite disadvantageous; it would require serious argumentation and investigation.

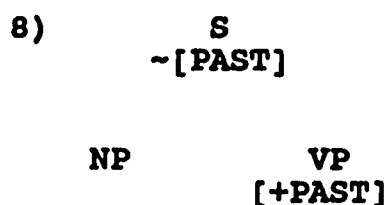
As we saw in Chapter Two, GKPS show why the HFC must give in to the requirements of the FCRs. However, GKPS (and therefore Shieber) do not discuss why the HFC must be sensitive to the requirements of the FFP and the

¹¹Alternatively, we could follow Gazdar (1981) in assuming that SLASHed categories of the form X/Y are legitimate only if X is capable of dominating Y in the grammar. Then the restriction of GKPS' FCR 6 follows as a particular case of this general restriction.

CAP. I will now examine this claim more closely and show that it is incorrect.

3.3.1 The HFC and the FCRs

It is easy to find a case where the HFC is in conflict with the FCRs and the FCRs have their way. (That is, the FCRs 'take precedence' over the HFC.) For example, consider the tree in (8).



The HFC should enforce equality between S and VP on the feature PAST since it is a HEAD feature and VP is the head of S. However, GKPS' FCR 5, given in (9), makes S undefined for PAST.

9) [PAST] $\supset$ [FIN, -SUBJ]

Therefore, the HFC ignores this feature when checking the correspondences between S and VP (i.e., PAST is not 'free' on S). This can also be seen in the lack of SLASH passing to lexical heads. This is enforced by GKPS' FCR 6, given in (10).

10) [SUBCAT] $\supset$ ~[SLASH]

In a tree like (11), the HFC would direct the SLASH to the V (the head daughter) but (10) does not allow that.

11)

VP/NP

V
[2]

NP/NP

(The FFP ensures that the SLASH is passed to some daughter, in this case the NP.) Again, we see that the requirements of the FCRs seem to be more compelling than (take precedence over) the requirements of the HFC.

3.3.2 The HFC and the FFP

It is somewhat more difficult (if not impossible) to show a clear case of the FFP and the HFC interacting, where the FCRs do not come into play. A clear case of such an interaction would have a HEAD feature on the head daughter and a FOOT feature on the mother (because it is also on another daughter), where those two features are incompatible. Unfortunately, there are few such pairs of incompatible features. The only ones that I can find are SLASH and WH, and SLASH and SUBCAT.

To make the first pair conflict, since WH is a FOOT feature, we must take SLASH as a HEAD feature. Since SLASH is also a FOOT feature, it would not be clear how to interpret its interaction with WH. That is, we could not tell if we were looking at the interaction of a HEAD and a FOOT feature, or at the interaction of two FOOT features, or at the interaction of a HEAD/FOOT feature with a FOOT feature, since SLASH is the only feature that has been proposed to be both a HEAD and a

FOOT feature simultaneously. Hence, any argument based on this pair would be unreliable. In the second pair, we can take SLASH as a FOOT feature. Unfortunately, SUBCAT is independently restricted to [BAR 0] categories, which cannot be mothers, so that pair is also untestable.

The clearest case where the FFP and HFC might have different requirements involves only one feature, not two. Consider top-linking rules, as in (12), where the SLASH is inherited on the head daughter.¹²

12) a. $S \rightarrow X^2, H/X^2$

b. S

NP S/NP

Given that SLASH is a HEAD feature, then the HFC should force the SLASH to be present on the mother and head daughter, but the FFP requires that it not be present on one if it is inherited on the other. Since (12b) occurs, and must be generated by (12a), it seems that the HFC gives in here to the FFP. However, this analysis is suspect, given certain problems with top-linking rules in general.¹³ I conclude, therefore, that there is no solid

¹²In the other cases of inherited SLASH (that is, tough and 'it clefts'), the SLASH is on a non-head daughter and so these constructions are not relevant here.

¹³Jacobson (1987) points out that top-linking cannot work as proposed by GKPS because they use the schematic rule in (i) and then have the CAP force the two instances of X^2 to be instantiated identically.

evidence to order the HFC and the FFP with respect to one another, since they affect disjoint sets of features, except for SLASH. (Note, however, that the FFP must follow the CAP as Shieber claims, because the CAP is only sensitive to inherited FOOT features and not to instantiated FOOT features, which arise due to the operation of the FFP.)

3.3.3 The HFC and the CAP

Returning now to the interaction of the CAP and the HFC, it will be simplest to look at the various possibilities in terms of heads and controllers. There are effectively three ways that the HFC and the CAP could interact. The first is that there might be no interaction at all (i.e., that the head daughter is neither the controller nor the controllee). This is the case with subject control, object control, STM 2, and tough constructions. In all of these, the HEAD feature specifications on the head daughter are irrelevant to the feature specifications of the control feature on the

However, the CAP can only apply if X^2 is instantiated identically in both of its occurrences. In order to avoid this circularity, Jacobson proposes that ". . . the extraction rule itself must stipulate that the dislocated constituent matches the category of the gap." (p.411). Fodor (in press) argues that this matching is the default in 'Chomsky adjunction' situation. This will be discussed in more detail in section 4.1.3.3.

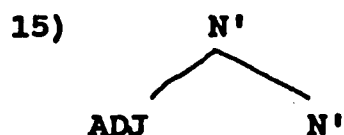
controllee. For example, consider a case of object control, as illustrated in (13).



The V is the head daughter, but the NP is the controller. The HFC will see to it that the mother and its head daughter will have the same HEAD feature specifications (including AGR), and the CAP will see to it that the VP daughter will have an AGR specification equal to the NP. The HEAD features on the head and the control feature on the controllee are unconnected. Therefore, these cases tell us nothing about the ordering of the HFC and the CAP.

The second possible relationship between the HFC and the CAP is where the head daughter is the controller. This is the case in GKPS' discussion (p.91) of DET and ADJ concord with the head noun. Consider the rule in (14)¹⁴, which projects the tree in (15).

14) $N' \rightarrow H, ADJ$

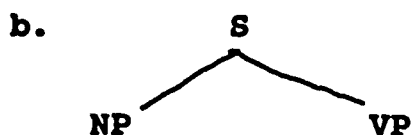


¹⁴GKPS do not actually list this rule, but given that the semantic type they associate with adjectives is $\langle N', N' \rangle$, they must assume a rule approximately like this.

As we saw in section 3.2, in order to get Shieber's ordering effects right, all of the HFC must apply before the CAP, so that the head daughter will be instantiated with the features included in concord (e.g., PER, NUM, GEND, and possibly CASE). If the CAP came first, the head would still be underspecified, so the CAP could not tell if agreement should take place and the features of the head would not be available for copying into (or on) the AGR feature on the ADJ.

The third possible way for that HFC and the CAP to interact is that the head daughter could be the controllee. This situation holds only for normal subject-verb agreement and expletive and non-nominal subjects; all of which require that certain features be identical on the subject XP and its sister VP. Again, it is necessary for (at least part of) the HEAD features to be instantiated onto the head in order to know that control holds and the CAP should apply. The HFC applying before the CAP would handle this. Consider, for example, subject-verb agreement, as in (16). (The same point could be made for both expletive and non-nominal subjects.)

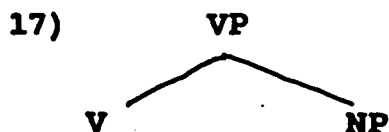
16) a. $S \rightarrow X^2, H[-SUBJ]$



The HFC will apply first to flesh out the features of VP, which then determine that the CAP must apply. The CAP then will pass features from NP to the AGR feature on VP. AGR is a HEAD feature, and the VP is the head of S. This would cause a problem if the AGR feature had to then be passed up to its mother, since that would require the HFC again. But in Chapter Two it was argued that AGR should not appear on S and a new FCR was proposed to prevent this. However, this fact would now follow as a consequence of the ordering of the CAP after the HFC, so the proposed FCR is not needed. This is a nice, if small, piece of evidence for this ordering, as opposed to the Shieber/GKPS ordering. There is, then, no conflict between the operation of the HFC and the CAP when the head daughter is the controllee and the HFC does not need to copy the information supplied by the CAP. It seems then that the HFC can come before the CAP and therefore that it should.

One other case that could still be a problem is that of verb-object agreement. In the tree in (17), the NP controls the V, and the V will show object agreement if the language has the morphology for it.¹⁵

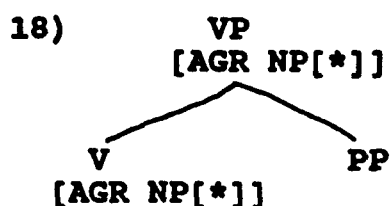
¹⁵Given that there are two types of possible verbal agreement (i.e., subject-verb and verb-object), the feature AGR should probably be split into SUBJAGR and OBJAGR, or allowed to take a stack of categories as its value (see Stucky, 1981, Horrocks, 1983 and Pullum, 1985b). I will assume the former analysis, with SUBJAGR



Here, the HFC will copy HEAD features down onto V from the mother, and the CAP will copy features over to V from the sister NP. There could be a problem here, if for some reason the VP also had to be marked for object agreement (that is, if 'OBJAGR' had to be a HEAD feature). However, this is hard to imagine. In fact, the passing up of object-agreement features should probably be universally prohibited, unless the languages of the world show cases where VPs are selected in terms of the PERSON, NUMBER or GENDER of their objects (as VPs are selected in terms of their VFORM, for example). That does not seem to happen, so object agreement should not be marked on VP. In general, agreement relations (in terms of PERSON, NUMBER and GENDER, at least) appear to hold between sisters, and though the agreement information passes downward from them, it seems never to pass upward. The downward passing does need the HFC, but the HFC will apply in the local tree immediately beneath the one in which the agreement relation holds, not in the same local tree. For example, (16b) above would link

and OBJAGR, is correct, although nothing hinges on this assumption.

with (18) beneath it (where * indicates matched feature values).



The agreement feature on VP would thus be 'passed down', but without there being any interaction of the CAP and the HFC in the same local tree. Therefore these constructions cannot be the source of any conflict between the CAP and the HFC. In fact, like the case above where AGR should be prohibited from occurring on S, ordering the HFC before the CAP will automatically block OBJAGR from occurring on VP. This is the right result and again supports the proposed ordering.

3.3.4 The HFC precedes the CAP

It seems, then, that GKPS had to complicate their definition of the HFC only because of their analysis of top-linking, and their allowing AGR to be specified on S. We can avoid this by using a version of Jacobson's analysis of top-linking (which was briefly noted in footnote 13 and will be made more precise in the next chapter), and by blocking AGR from appearing on S (which is needed for descriptive purposes, anyway, as noted

above).¹⁶ Once these changes have been made, there is no reason to order the HFC after the CAP. There remain only the reasons for ordering the HFC *before* the CAP, viz. that we can eliminate at least two FCRs and the fact that the CAP is sensitive to features instantiated by the HFC both for the triggering of control, and for the values of the features that must 'agree'.

3.3.4.1 Reordering the FSDs. There is another change that must be made in the GKPS system as modified by Shieber. Shieber argues that the lexical FSDs must be ordered before the HFC, as shown in (6), repeated here for convenience as (19).

```

19)      METARULES >>
          CAP >>
          FFP >>
          FSDlex >>
          HFC >>
          FSDnonlex >>
          LP

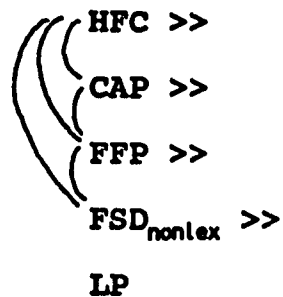
```

¹⁶GKPS might possibly have been aware of this. Although they say that the HFC must give way to the FFP and the CAP, as quoted above in the text, later they say:
 . . . the HFC, which requires that a mother's *HEAD* feature specifications are also part of the head's *HEAD* feature specifications (modulo FCRs)(p. 142)
 Note that they do not include the CAP and the FFP here in what the HFC can ignore.

Changing the ordering of the HFC relative to the CAP and the FFP will then also change the ordering of the lexical FSDs. The new ordering is given in (20).

20) METARULES >>

FSD_{lex} >> HFC


 HFC >>
 CAP >>
 FFP >>
 FSD_{nonlex} >>
 LP

(It will be modified again below.)

3.4 The Revised Ordering

Let us consider for a moment the question of how the system will generate trees, with the new ordering that has been proposed here. There will be a lexicon, with the set of possible SUBCAT feature values determined by lexical ID rules, including those derived by metarules. The ID rules will be quite schematic, like those in (21).

21) a.
$$\begin{bmatrix} -N \\ +V \\ \text{BAR } 2 \\ +\text{SUBJ} \end{bmatrix} \rightarrow \begin{matrix} H \\ [-\text{SUBJ}], \end{matrix} \begin{bmatrix} +N \\ -V \\ \text{BAR } 2 \end{bmatrix}$$

$$b. \begin{bmatrix} -N \\ +V \\ \text{BAR } 2 \\ -\text{SUBJ} \end{bmatrix} \rightarrow \begin{matrix} H \\ [\text{SUBCAT } 2], \end{matrix} \begin{bmatrix} +N \\ -V \\ \text{BAR } 2 \end{bmatrix}$$

c. NP $\rightarrow$ DET, H¹

d. N¹ $\rightarrow$ H[30]

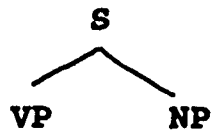
Clearly, these rules will not generate anything like the kinds of trees necessary for the description of a language. The rules must be instantiated by the ordered system given above, for the trees to be filled out in the appropriate ways.

On Shieber's view, the principles of UFI add on features, as opposed to GKPS' view where they just check features that are already in the trees/rules by free instantiation. Consider a rule like (21d). Given the ordering in (20), this rule cannot be instantiated appropriately (although GKPS' simultaneous application of all of the principles would work). The lexical FSDs are first in the ordering, but most of them cannot apply. The lexical category in a local tree projected by (21d) will be marked only by H⁰ and SUBCAT, and therefore the FSDs will not, in general, apply to that category.¹⁷

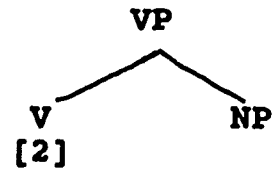
¹⁷Recall from Chapter Two that on GKPS' view, a [BAR 0] category is not a lexical item, but is the preterminal symbol that can dominate a lexical item just in case it is an extension of the category label in the lexical item.

The HFC, in its simplest conception, would derive the trees in (22) from the rules in (21), but clearly this is not all of the information about HEAD features that we need in trees.

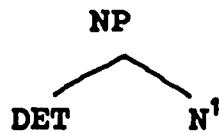
22) a.



b.



c.

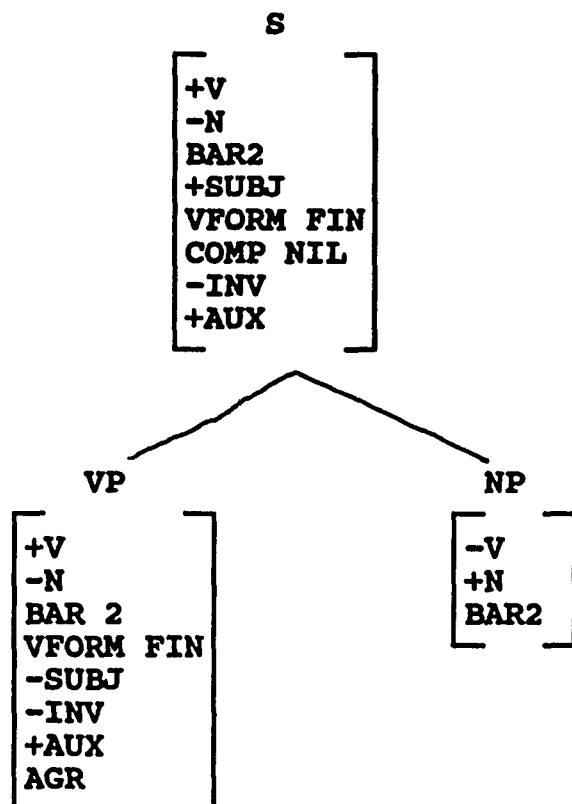


d.



This simple version of the HFC, which effectively only equates the HEAD features inherited on mother and head daughter(s), is not sufficient. We must, therefore, allow the HFC to go through the list of HEAD features that are not mentioned in the rule, and instantiate some value for each of them identically on mother and head, as long as it does not violate the FCRs. (22a) could then look like (23).

23)



Two things are noteworthy about this tree. First, notice that we have not yet added anything new to the non-head (NP) node; second, note that the AGR on VP is still unspecified because the CAP has not yet applied. However, the CAP will not apply appropriately until the HEAD features on the non-head daughter are instantiated. Clearly, what the HFC needs to say is: instantiate HEAD features on all nodes, and make sure that HEAD features on the mother and head have the same values. This will get the NP in (23) more fully specified, say with [NFORM NORM, +PLU, PER 1, CASE NOM].

The HFC has now done all that it can, and the CAP can now apply. The NP controls the VP, and therefore the

CAP applies. In this situation, it copies the relevant feature specifications (GKPS' chi-specifications, or, as argued in Chapter Four, some reduced set of what GKPS take to be HEAD features) from NP into AGR on VP. This is fairly straightforward. Next it is the FFP's turn to operate. Let us, for simplicity's sake, assume that in this instance no FOOT features are instantiated. We are left then to supply the nonlexical defaults. Notice that an interesting thing has happened. The HFC has supplied all the HEAD features, and the FFP has supplied all the FOOT features. Therefore, there is almost nothing left for the nonlexical FSDs (given in (24)) to do.

- 24) a. [-INV]
 b. ~[CONJ]
 c. ~[NULL]
 d. ~[NOM]
 e. [NFORM] $\supset$ [NFORM NORM]
 f. [INF, +SUBJ] $\supset$ [COMP for]
 g. [+N, -V, BAR 2] $\supset$ [ACC]

Consider, for example, (24a). When this default could apply, according to the ordering, the value of INV in any given tree will already have been established by the HFC. Thus, there can be no non-lexical defaults for HEAD or FOOT features, since those defaults would not apply until after the HFC and FFP have already instantiated the HEAD and FOOT features. Therefore, an FSD like (24a) cannot exist, even though it appears to represent a true generalization. All that will be left unspecified (and thus available to the FSDs) are the 'non-propagating'

features, if they have not been inherited from the ID rule. (The non-propagating features are: CASE, COMP, CONJ, GER, NEG, NFORM, NULL, POSS, REMOR, and WHMOR.)

An important and related question that was ignored above arises at this point: what happens if SLASH is instantiated in a tree, since it is both a HEAD feature and a FOOT feature? In general, the question is, what happens when one feature is controlled by the operation of more than one principle. The problem is that, assuming with Shieber that the principles instantiate features, not just check them, some features should be introduced more than once, which is untenable in a monotonic system. SLASH is the only feature that is both a HEAD and a FOOT feature, but the distribution of both HEAD and FOOT features will be partially determined by the FSDs. Of the 11 FSDs that GKPS propose, 9 of them concern what we are taking to be either HEAD or FOOT features. Given the approach here, these features should be introduced by the HFC or the FFP, but then they should also be introduced by the appropriate FSD.¹⁸ Obviously, this will not work. A feature can be checked any number

¹⁸On Shieber's model, the HFC essentially has two components: one which copies HEAD features from mother to head daughter (or vice versa) and one which instantiates pairs of HEAD features, one on the mother and one on the head. The FFP has an analog of the second component, but not the first. In the rest of this chapter, I will be ignoring the copying part of the HFC and will consider this difference again in Chapter Four.

of times, but it can only be introduced into any local tree once.

Intuitively, what we want is for the FSDs to apply last in the ordering. They are defaults and should supply their values to a feature only if nothing else has already supplied a value to that feature. Thus, we must constrain the principles of UFI such that they do not 'free instantiate' non-default values, which may be introduced into trees only by the ID rule. In some ways, this makes the FSDs operate much like the FCRs (in that they act like 'filters' on admissible categories), the difference being that the FCRs are absolute while the FSDs are overridable. This is quite reasonable given that FCRs and FSDs are formally identical: it seems strange then for them to operate in completely different manners.

What I am proposing is that the HFC and the FFP operate to insert HEAD and FOOT features, respectively, and that they do so freely, up to the limitations placed on them by the FCRs and the FSDs. If the FSDs are not interpreted in this fashion, then the only way to stop the HFC and the FFP from freely instantiating non-default values for HEAD and FOOT features, respectively, is to write the default value explicitly into each relevant rule. Thus, the generalizations that the FSDs express would be dispersed throughout the ID rules and not

expressed anywhere. One positive consequence of this interpretation of the FSDs is that in order to override the default, the non-default value must be explicit in the rule. If it is not, then there would be no way to get that value instantiated in the tree. GKPS also used explicit feature values in rules to override the FSDs, but in addition, they also allowed the HFC, the FFP and the CAP to override the defaults. That is what I am changing. This change is intuitively pleasing in that trees which include non-default values for features must be projected from rules which are more complex than the rules that project trees with only default values in them. That is, non-default values cause more marking in the grammar than do default values. This will entail some increase in the overall complexity of the grammar, but it is offset by the fact that grammar size will more closely reflect a markedness ranking.

The proposed change will also eliminate the need to separate lexical from non-lexical FSDs. GKPS had to have this distinction in order not to allow mothers and heads to arbitrarily have non-default values, just because they co-varied (cf. discussion in Chapter Two). Shieber then just translated this distinction into his system. However, it has never been clear what the theoretical motivation is for distinguishing lexical from non-lexical FSDs (other than it is necessary to make

either system work). In addition, Shieber's translation of the GKPS distinction is in fact flawed. Consider again the default in (24a). Is this a lexical or a non-lexical FSD? It must be both: it must apply not only to lexical Vs, but also to VPs, which are non-lexical. The same point can be made with the first five of the seven FSDs in (24). Thus Shieber would be forced to claim either that each of these five FSDs in fact represent two identical FSDs, one lexical and one non-lexical or that some FSDs apply more than once in his ordering. Neither of these are attractive positions to hold.

In my system, however, the FSDs are like the FCRs, in that they are in some sense always in effect. The difference between FCRs and FSDs is that ID rules can explicitly override an FSD, but not an FCR. (The explicit override from an ID rule can then be passed through the tree by the operation of UFI.) Thus, we do not need to try to separate the FSDs into two different groups and a distinction that is at best poorly motivated can be eliminated on the proposal here.

Changing how the FSDs apply is not only welcome because of its results for markedness, but it also seems to be necessary. That is, Shieber's view that the syntactic principles actually insert the features and not just check the instantiations that exist independently will not work out quite as simply as he assumes. It

seems that there is no natural translation from checking principles to instantiation principles. The reason for this is quite simple. Consider a feature like PFORM. The occurrence of PFORM is partially governed by an FSD which says that it occurs only on P^0 , in the default case. But PFORM is also a HEAD feature, therefore when it occurs, its distribution is governed by the HFC (i.e., PFORM must also occur on the P-bar mother). However, obviously it cannot be inserted by both of these principles. What we seem to need to say, then, is that GKPS' principles of UFI (that is, the CAP, HFC and FFP) insert the features, but their operation is governed by the FCRs and FSDs.

If the principles of UFI can only instantiate the default values of features, then how will non-default values ever be present in trees? The only option seems to be that they are inherited from the ID rules. That is, non-default values must be explicitly marked in the ID rules in order to occur in trees. The principles of UFI will be free to instantiate any value for a feature only if that feature has no default value. (Which features seem not to have any default value will be discussed in Chapter Five.) This proposal is a shift from Shieber's view, but it seems necessary, given that one feature cannot be introduced by two different principles. Having the FSDs act as overridable filters

on allowable categories while FCRs are absolute filters on admissible categories reflects the fact that FCRs and FSDs are formally indistinguishable. Shieber's ordering allows for vast simplification, which is desirable, but it must be modified in order to operate. This model is now very similar to that proposed by Fodor and Crain (in press). They, however, argue their case on learnability grounds, rather than simplicity.

3.4.1 Eliminating some non-lexical FSDs

Consider again how this system is going to work. The ID rules will have some features specified in them. Since this is a monotonic system (i.e., once a feature value is specified, it cannot be changed), none of the principles can change these values; they must remain in the tree. The HFC then instantiates those HEAD features that have not yet had a value specified for them. The CAP will then handle the agreement feature(s), and the FFP will instantiate any FOOT features that appear. All of this feature instantiation will be 'free' up to the limits imposed by the FCRs and the FSDs. All that is left now is to specify all the 'non-propagating' features. But, in fact, if we say nothing, then these features will not occur in the trees. This is exactly the right situation. These are basically special purpose features which mark such things as COMPLEMENTIZERS,

CONJunctions, GERundives, POSSessives, etc. These features should only occur in a tree if they are inherited from the ID rule that licenses that tree. We do not need to do anything special about them at all.

To summarize: we have seen that by accepting and modifying Shieber's ordering, we have been able to simplify the theory of GKPS in some interesting ways. We have eliminated the definitions of 'privileged', 'free', and at least some of the FSDs. However, as noted at the beginning of this chapter, the CAP is complex and the reordering does not simplify it much. Jacobson (1987) argues that there are serious questions about the validity of the generalizations that the CAP makes. She points out that the CAP requires three separate stipulations to account for control, while other linguistic theories need only one or two to handle the same array of data. This suggests that the GKPS analysis of control is missing something. So in the next chapter we turn to a closer inspection of the constructions that GKPS use the CAP for, and see if we can simplify this part of the grammar too.

CHAPTER FOUR

CONCERNING THE CAP

The theory proposed by GKPS is impressive in its coverage of the constructions in English and in the cleverness of its theoretical machinery. Our goal has been to streamline and, in as far as possible, to explain the interactions between the components of GPSG. As we saw in the previous chapter, if we establish an ordering among the principles of the grammar, those principles become very simple. That is, all of the principles except the CAP. Therefore, in this chapter, I will examine the CAP more closely, to see if perhaps it can be simplified too.

4.1 Focusing on the CAP

The CAP is in many ways the most complex and problematic principle that GKPS propose. Partly this complexity is because the CAP is a syntactic principle that is semantically driven; the interface between two components of a grammar often seems to engender difficulty. But also, the CAP is formulated to account

for a variety of data which at first blush seem to be very different.

4.1.1 What the CAP does

The CAP is designed to cover (at least) the eight different cases listed in (1).¹

- 1)
 - a. Subject-verb agreement (and nominative case)
 - b. Object control (and SAI)
 - c. Expletive subjects
 - d. Non-nominal subjects
 - e. Top-linking
 - f. Subject control (and Raising)
 - g. SLASH Termination Metarule 2 (STM 2)
 - h. Tough-constructions

The implicit claim is that these constructions share a similar syntactic structure, exhibit a particular featural relationship, and that information about that relationship is then passed through trees along the same paths. It is commendable to try and unify these various phenomena, as GKPS did, and it would be extremely interesting if such apparently disparate phenomena can be successfully unified. That would result in a significant simplification of the grammar and a real generalization. But it may be that this project is too ambitious, that in fact these phenomena do not all have something in common. I will suggest below that this is in fact the case, and

¹GKPS also assume that the distribution of reflexive and reciprocal pronouns in infinitives is a reflex of syntactic agreement (cf. GKPS, p. 84). Since they do not discuss this any further, and because anaphor binding is a thorny issue that would take us far afield, I will not discuss anaphor binding below.

that there is an overall gain in simplification and generalization to be had by relinquishing the attempt to unify all of (1).

4.1.2 How the CAP does it

In Chapter Two we saw that the CAP has two clauses. The first of these says, roughly, that in a control situation, the control feature *F* on the controllee must equal the chi-specifications² on the controller unified with the value of *F* on the controller (more about this below). The second clause says, again roughly, that, if a predicative category has no controller, then its control feature must equal the control feature of its mother. The control feature of a category is inherited SLASH, if that is present, otherwise it is AGR (which, in English, can only appear on projections of [+V]). The first clause of the CAP takes care of (1a), (1b), (1c), (1d), and (1e). Clause two of the CAP covers (1f), (1g), and (1h). The first five are cases of control between sisters and the last three involve mother to daughter relations.

²Chi-specifications are the non-FOOT HEAD and inherited FOOT features. These features do not change the semantic type of a category in such a way as to change its control relationships, so they form a natural class with respect to *where* the CAP applies. But they are not necessarily a natural class with respect to what features the CAP *affects*.

4.1.3 Problems for the CAP

Although the CAP does an impressive job in handling the phenomena in (1), there are still some problems with the analyses. This section will examine some of these problems and consider their consequences for the theory in general.

4.1.3.1 What features the CAP affects. GKPS use the term 'chi-specifications' (see 2.4.2.1 above) to denote the features relevant to determining where a control relation holds (where the CAP is to apply) and those features that must have equivalent values (due to the operation of the CAP). However, it seems as if the notion of chi-specifications expresses a spurious generalization. The definition of 'chi-specifications' embodies GKPS' claim that the non-FOOT HEAD features and inherited FOOT features determine the semantic type of a category relevant for control. This is because inherited SLASH, but not instantiated SLASH, influences the control relationships in a local tree. However, it is not at all clear that WH or RE could change the semantic type of a category, even if they were inherited, which would make them chi-specifications. When WH is inherited (as it is on the clausal sister to verbs like wonder, in relative clauses and in some it-clefts), it never affects the control relations. The other FOOT feature (i.e. RE)

appears to be always instantiated, it is never required in a construction.³ Thus, GKPS' stipulation "inherited FOOT features" in the definition of chi-specifications really means inherited SLASH and nothing else.

It is also the chi-specifications that the CAP forces to be identical on the controller and in the control feature on the controllee. GKPS say (p. 89) that it is just the chi-specifications that the controller and controllee must agree on. Once again, however, the class of chi-specifications appears not to capture the appropriate generalization. GKPS include in their formulation of clause (i) of the CAP (p. 89) some formalism that is supposed to force agreement on features that are neither HEAD nor FOOT features and hence not chi-specifications. This is to make the CAP constrain the distribution of the features CASE and NFORM, which are non-propagating features, but which are important to their analysis of agreement and control. Thus, it is not just the chi-specifications which must be identical in a control relation.

³Whether this claim can be maintained in its strong form, or must be weakened will depend on the appropriate analysis of a verb like perjure. If the fact that it requires a reflexive object is a semantic fact, then the syntax can say that RE is never inherited. However, if it is a syntactic requirement, not only will RE be inherited in this case, but, as mentioned in Chapter Three, perjure will then have to have a distinct ID rule in which RE is specified (that is, a new ID rule will have to be posited).

It is not clear, furthermore, that *all* chi-specifications must be matched. For instance, it seems that the issue never arises for FOOT features. Specifically, it is never the case that the controller has an inherited SLASH (as noted above, RE is never inherited, thus it is never part of the chi-specifications). If it did, then there could be a rule like (2) where the NP/NP controls its VP sister, but has no antecedent and it is not clear where it will get its interpretation from.⁴

2) VP --> H, NP/NP, VP

This rule would permit a sentence like "John persuaded to go", where the object/subject is a trace without an antecedent and therefore uninterpretable. (2) is distinct from the top-linking rules proposed by GKPS in that it is the controller, not the controllee, that is inheriting the SLASH. It seems plausible that this kind of rule should be ruled out universally.

Also, the controllee and the controller can never both have inherited SLASH, although they can, and often do, have distinct inherited values for HEAD features. If

⁴There are other cases where SLASH gets its value from the AGR value of its mother, due to the operation of the CAP (see the discussion below), but this could not happen here. The CAP only 'passes' the value of the control feature on the mother to the control feature of an uncontrolled daughter where that daughter is a predicative category, and NP/NP is not predicative.

SLASH were inherited on both categories, it would be both the control feature and one of the chi-specifications on each of the categories. Since SLASH would be the control feature on the controllee, its value would have to include the SLASH specification of the controller since SLASH is a chi-specification. GKPS do not allow a category-valued feature to take categories specified for that feature as its values. The controllee would have to be specified as <SLASH [X [SLASH ...]]> and for GKPS that is not a possible category (cf. GKPS p. 36).⁵ Also, since RE and WH never seem to change the control relationships in local trees, they seem to be irrelevant to the operation of the CAP, in English, at least. It seems that they are included in the formulation of the CAP in order to make it universal (cf. GKPS p. 89, footnote. 21). To sum up: SLASH is only relevant to the

⁵If category-valued features were allowed to contain categories specified for that feature, then the embedding of categories could go on infinitely. This would allow an infinite number of categories and would entail that the grammar is not equivalent to a context-free phrase structure grammar. It may well prove necessary for the grammars of natural languages to have more than context-free power (see Shieber, 1985 and Culy, 1985), but GKPS adopt the more restrictive position. However, they note (p.81) that English might in fact require categories specified as [SLASH X, SLASH Y] for sentences like (i).

(i) Which pots are those soups easy to cook in. Longobardi (1985) notes that Italian allows the extraction of categories containing gaps; these are described by Fodor and Jones (1986) using nested SLASH categories.

CAP if it is inherited on a category, then it is the control feature on that category. Otherwise, all FOOT features are irrelevant. Possibly no harm is done by including them, but they may not represent a genuine generalization.

In addition, contrary to what GKPS say in the text, they clearly mean for the CAP to enforce equality on more than the chi-specifications. They intend for it to also require equality on any feature that is specified in the control feature of the controllee. This is how they "force nominative case onto the subjects of tensed sentences" (p. 94) and how they account for the distribution of expletive subjects. However, these analyses do not work when examined closely.

GKPS' FCR 10, given in (3), requires that FINite verbs and VPs which agree with NPs, 'agree' with a NOMinative NP.

3) [FIN, AGR NP] $\supset$ [AGR NP[NOM]]

This then exempts the subject NP from FSD 11, given in (4), which says that NPs are normally in the ACCusative case.

4) [+N, -V, BAR 2] = [ACC]

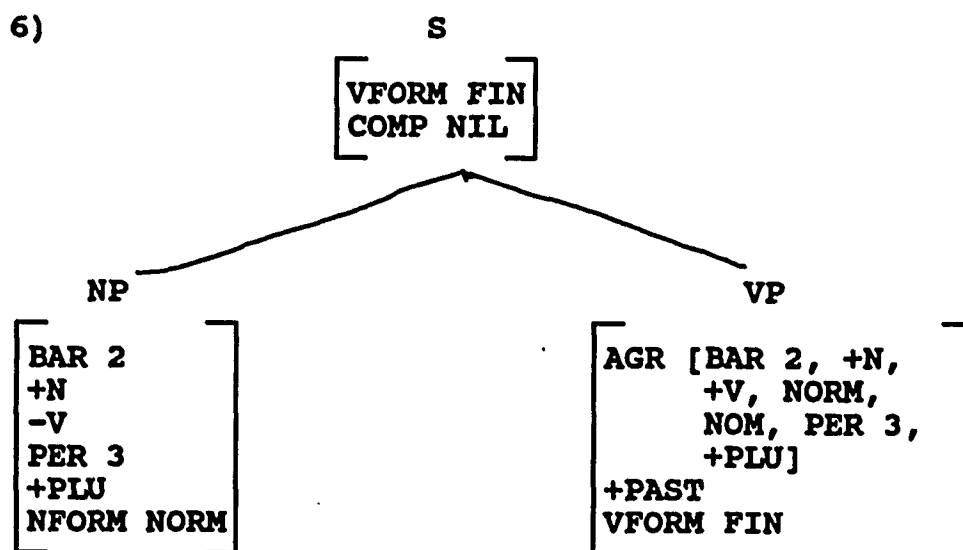
CASE is not a HEAD or a FOOT feature, but clause (i) of the CAP is supposed to ensure its presence on the subject NP since it is specified in AGR (the control feature) on the VP.

Even if we ignore, for the moment, the fact that CASE and NFORM are not a part of the chi-specifications and therefore GKPS are using the CAP to deal with features that it cannot, this analysis of nominative case assignment still does not quite work out. In order to see why, we must look closely for a moment at clause (i) of the CAP. GKPS formulate it as in (5).

- 5) if $\phi(C_i)$ controls $\phi(C_j)$, then
 $\phi(C_i)(f_i) = {}_x(\phi(C_j)) \sqcup \phi(C_i) \setminus \{f_i\}$, where f_i
 is the control feature of $\phi(C_i)$ (GKPS p.89)
 ($\sqcup$ means 'unified with')

Put into words, what (5) says is, if in a projection from a rule the category C_j controls the category C_i , then the control feature on C_i must equal the chi-specifications on C_j unified with those features on C_i which are also in the control feature on C_i .

Consider now the tree in (6), where the NP controls the VP.



Substituting the appropriate values into (5), we get: AGR of VP = [BAR 2, +N, -V, PER 3, +PLU] $\sqcup$ {}. (Note that $NP \upharpoonright \{AGR(VP)\}$ equals the empty set since there is no AGR on NP.) Chapter Two gave the definition for the unification of a set of categories: if it is defined for a set, it is the smallest category that is an extension of every member of that set. In this case, the CAP would require that the value of AGR on VP be [BAR 2, +N, -V, PER 3, + PLU].

Obviously, this is the wrong result since FCRs and FSDs (the FCR in (3), for example) will force AGR to also contain [NFORM NORM] and [CASE NOM]. Note that the FCR, for GKPS will not *supply* the required features; they must be freely instantiated into AGR in order to meet the FCR.

It appears that what GKPS must be meaning here is to take AGR as a set of features (like HEAD or FOOT features) rather than as an individual category valued feature (like SLASH) (cf. GKPS, p. 78). Otherwise, strict adherence to the CAP would mean that the AGR of VP should have to equal just [BAR 2, +N, -V, PER 3, +PLU], which it cannot because of the FCR in (3). However, we can assume that this is just a terminological problem, and that the appropriate formalism can be devised. Then we can take the CAP to say that the control feature of the controllee must equal the chi-specifications of the controller, unified with those features on the controller that are also in the

control feature on the controllee. The unification of those two sets will then be exactly the same as the AGR specification on the VP.

This comes closer to yielding the correct results, but it still gives no way to force CASE to be NOM on the subject. The problem is that, in rules, certain features (eg., CASE and NFORM) need to be specified in AGR of VP but should be unspecified on the subject, while other features (eg., PERSON and NUMBER) need to be specified on the subject but should not be specified in AGR of VP. What we want to be able to say is that AGR of VP and certain features on the subject will be identical. Given the definition of unification, all that is ensured now is that the subject is not marked ACC or DAT, etc.; it may be NOM or not marked for CASE at all.

We might just accept this, given that CASE is irrelevant to (i.e., not marked on) non-lexical NPs in English (it will be lexically specified on lexical NPs such as the pronouns). This would, however, violate the spirit of GKPS since it will not work universally (because there are languages which require overt CASE on all NP subjects and this system will not be able to rule out subjects not marked for CASE) and the CAP is taken to be part of Universal Grammar. In fact, it is also incompatible with the GKPS analysis of tough-movement. The ID rule that they postulate for the tough construction has to mention [-NOM]

on the SLASH that is introduced, to insure that the SLASH does not terminate as a subject. Otherwise, sentences such as (7) could be generated, where John is understood to be the subject of hits.⁶

7) *John is easy to think hits Bill.

If subject NPs could be unspecified for CASE, then the tough ID rule would have to make the SLASH non-NOMinative *and* not unspecified for CASE. While this could probably be formalized, it would be quite inelegant and would obscure the fact that the gap in a tough construction cannot be the subject of a tensed clause.

Another possible solution to the problem of getting the CAP to override the ACC default would be to say that the subject must be specified for all features that it is defined for. Generalizing, this means that features may not be unspecified on a category unless they are undefined for that category. If CASE is specified on the NP, then it must be specified for the same value on the NP and in AGR of VP or their unification will be undefined. Thus, a feature's being specified on the controller (either lexically, as is the case with pronouns, or by free

⁶This analysis is not without its flaws (see Jacobson (1987) and below for discussion). However, these flaws do not change the point being made here. Also, [-NOM] is uninterpretable, technically. [NOM] is an abbreviation for [CASE NOM] and not a Boolean feature. Clearly, the intended meaning is 'not [CASE NOM]', but GKPS have no formalism to say exactly that.

instantiation) avoids this problem. However, given GKPS's analysis of conjunction (essentially the same as that of Sag, et al., 1985), which crucially relies on the presence of underspecified categories in trees, it is unclear how they could enforce the full specification of features on subjects and yet permit underspecification elsewhere. (But see Jacobson (1987) for arguments against this analysis of conjunction.)

A third solution,⁷ and probably the correct one, is simply to make CASE a HEAD feature, as is done in Sag, et al. (1985) and Pollard and Sag (1987). Then the CAP would handle CASE just like N or PLU. That is, CASE will now be one of the chi-specifications of NP. Since NOM will be specified in AGR of VP, either the NP will also be specified as NOM, or the CAP will cause the tree to be thrown out. This is exactly the same as a case where the VP is specified as [AGR NP] and the subject is a PP. In this case also, the CAP is violated and no admissible tree is generated. The problem with this analysis is that if CASE is a HEAD feature, then conjoined NPs should have to share the same CASE, either NOM in subject position or ACC in

⁷There is another possible solution to this problem, based on intersection of AGR of VP and the chi-specifications of NP rather than their unification (this was pointed out to me by Janet Fodor). However, this is just another inelegant way of trying to force the CAP to regulate the distribution of features that are not part of the chi-specifications.

other positions, and that is not the case in some varieties of English. Consider (8).

- 8) a. John and me/I went for a walk.
- b. Him and her went for a walk.
- c. *Him went for a walk.
- d. She wanted him and I to attend the meeting.
- e. *She wanted I to attend the meeting.

(8a) shows that the case of a conjunct can vary, even though its position remains the same, (8b) and (8c) show that both conjuncts can be ACC even though the mother NP should be marked NOM, and (8d) and (8e) show that a conjunct can be NOM even though the mother NP must be ACC. This would be understandable on assumption that CASE is not a HEAD feature and therefore does not propagate. This behavior is anomalous on the assumption that CASE is a HEAD feature. However, I will assume that this is an idiosyncrasy of English, based on its general lack of a case system. In a regularly case-marked language, CASE will behave like any other HEAD features.

The way GKPS handle the distribution of expletive subjects with the non-HEAD non-FOOT feature NFORM is almost exactly analogous to their treatment of NOM. The only difference is that they have no FCR like (3) for NFORM; they have a FSD making NORM the default value for NFORM and another (lexical) FSD making AGR NP[NFORM NORM] the default for V^0 . This analysis of NFORM runs into almost exactly the same problem as the analysis of NOM above. The FSDs will insure that the VP is specified as [AGR NP [NORM]] (in

the default case) while the NP may or may not be specified as NORM. Therefore, the NP could be unspecified for NFORM while the VP inherits from the ID rule [AGR it] (or [AGR there]) and the resulting tree will not violate the CAP. It would also not violate the FSD which requires that, if an NP has a value for NFORM, then it is [NFORM NORM], since this NP is not marked for NFORM. However, if an NP subject is (lexically) specified as [NFORM it] or [NFORM there] then it must 'agree' with the AGR specification on the VP in terms of the marking for NFORM.

This is completely analogous to the CASE problem above; AGR of VP is specified with a non-HEAD, non-FOOT feature that the subject must bear and the CAP is responsible to guarantee that it does. If the subject NP is independently specified for this non-HEAD, non-FOOT feature, then the CAP ensures that AGR of VP has the same feature specification. But, if the subject is unspecified for this non-HEAD, non-FOOT feature, then the CAP cannot ensure that AGR of VP has the default specification. Thus, the problem with NFORM is the same kind of problem as the CASE problem, and it has the same kind of solution. Sag et al. also propose to include NFORM in the set of HEAD features.⁸ Once this is done, the CAP can account for the

⁸As presented, the analysis of Sag et al. is incompatible with the analysis of GKPS because Sag et al. propose to use NFORM to distinguish sentential NPs from non-sentential NPs and formulate the FSD in (i) to help

distribution of NFORM just as it does the other chi-specifications. I will assume, therefore, that NFORM should be a HEAD feature.

If we take NFORM and CASE to be HEAD features (bearing in mind the problem with nominative case assignment with conjoined NPs), we also solve the terminological problem noted above. In the definition of the CAP (see (5) above), the formula " $\bigwedge \phi(C_i) \mid \{f_i\}$ " was apparently included to force agreement on non-FOOT, non-HEAD features, i.e., CASE and NFORM. Now that these have been assimilated into the set of HEAD features, this part of the formalism is no longer needed. The assignment of nominative case and the distribution of expletive subjects now follow from the CAP with no extra stipulations added on. Thus the complexity of the CAP has been reduced somewhat, as it no longer needs to refer to the non-FOOT, non-HEAD features, even though the set of HEAD features has been increased slightly.

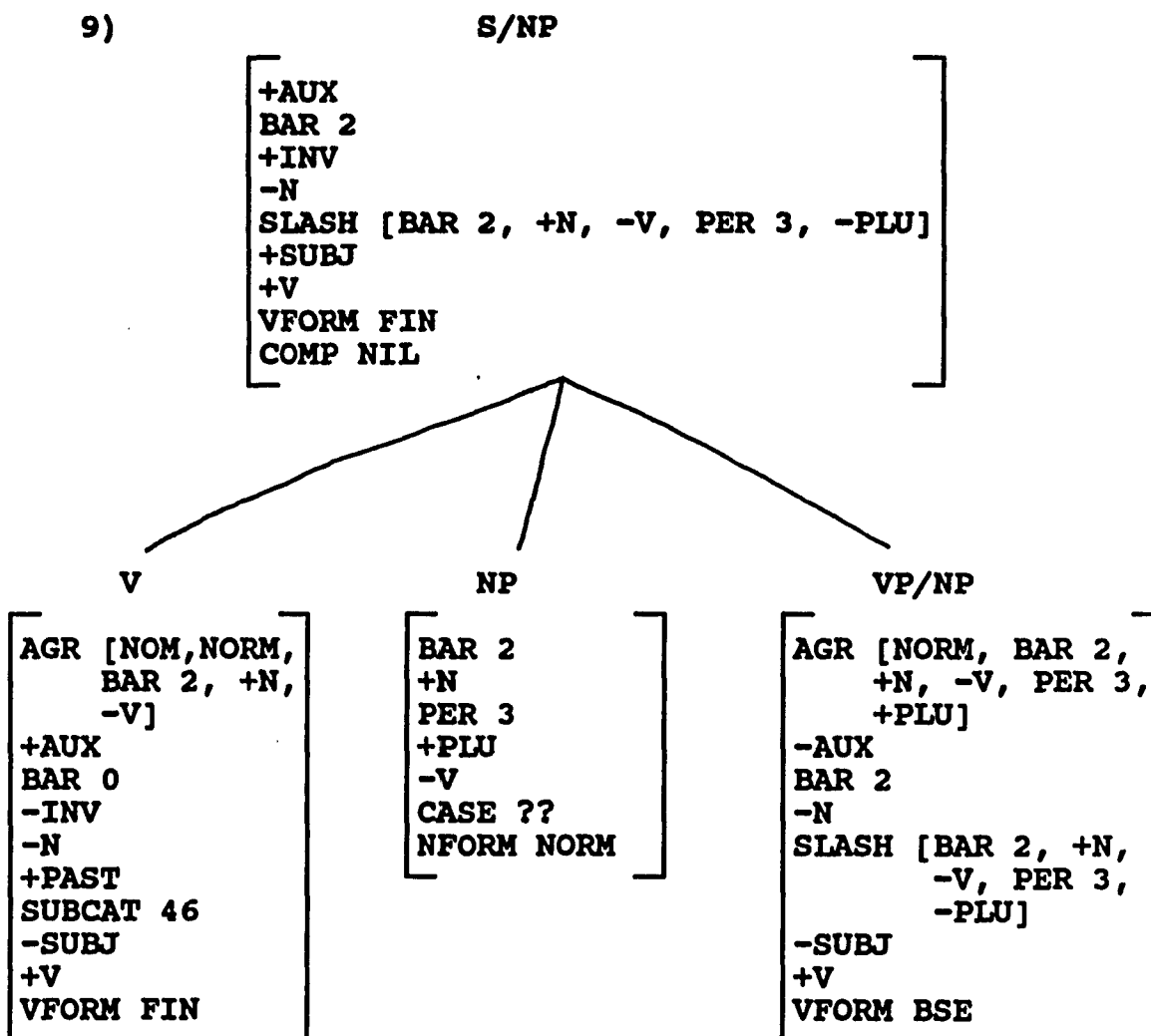
accomplish this.

(i) [NFORM] $\supset$ [CONJ]

They plan for this to restrict the distribution of sentential NPs to be just those sentences that are conjoined with NPs. However if this FSD is taken without modification into the GKPS analysis, it has the result that only conjoined NPs can be specified for NFORM and therefore it and there (and any ordinary NP which is marked as [NFORM NORM]) can only occur in conjoined structures. This is plainly false. However, if this FSD were formulated as (ii), then the right results would follow and the analysis of Sag et al. would mesh with that of GKPS.

(ii) [NFORM S] $\supset$ [CONJ]

4.1.3.2 Subject-aux inversion. Another problem with the CAP has to do with the output of the Subject-Aux Inversion (SAI) metarule. Consider the tree in (9).



The intention here is that the NP should control the head V, in order to insure agreement between them (cf. 'Who does John see?' vs. *'Who do John see?'). However, this does not follow from the GKPS definition of control. GKPS discuss two subcases of control. The first deals with binary branching trees and therefore it is irrelevant here.

The second subcase of control involves a control predicate, i.e. a lexical category mediating control between a NP and a VP complement (see Chapter Two for a discussion).

Obviously, this also is not relevant to the case at hand since we are concerned with an NP and its sister V, not VP.

What we have here is the canonical configuration for verbs and objects. Therefore, we could try to salvage the GKPS analysis by claiming this falls under the same principle as object agreement. This is not very plausible, given that English does not exhibit verb-object agreement. Also, this analysis predicts that in a language with object agreement, the subject agreement in (9) would surface as object agreement morphology on the verb. Again, this is not very plausible. Furthermore, this kind of verb-object agreement, in English, could be allowed only in inverted sentences, where the mother is S and so has no AGR features to pass down to the head verb. If noninverted verbs were allowed to agree with a following NP sister, then we would get the wrong agreement facts with verbs like kick, where a non-inverted verb has an NP sister which is really its object. Thus the grammar for English would have to stipulate somehow that it has agreement between a verb and its NP sister just in case the verb is inverted. It would be desirable to avoid this complication if possible.

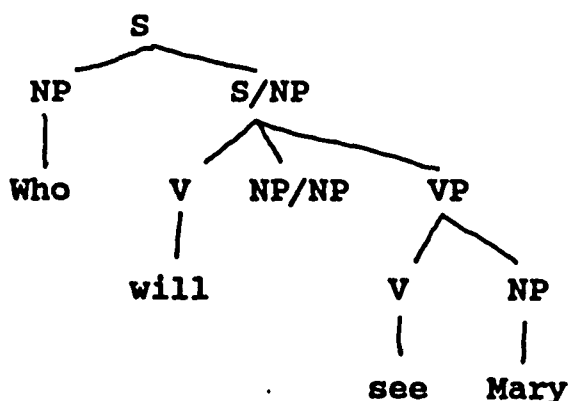
The NP in (9) will control its sister VP (since GKPS define invertible verbs as control predicates), which

it should. But it will not control its sister V, even though it must in order to account for the agreement facts. Also, if there were no control in this construction, there would be no way to guarantee that the subject will be assigned nominative case (even assuming that the problem noted in 4.1.3.1 above concerning nominative case assignment is remedied). In fact, the GKPS theory predicts that the subject in an inverted construction should be accusative not nominative, given the FSD in (3). ((3) will ensure that the subject is accusative since it controls no V (or VP) whose AGR is specified [CASE NOM].)

There are other problems for the GKPS analysis of SAI, also. Their analysis of SAI interacts with their analysis of unbounded dependencies to allow a 'string vacuous movement' analysis of (10a), given in (10b).

10) a. Who will e see Mary?

b.

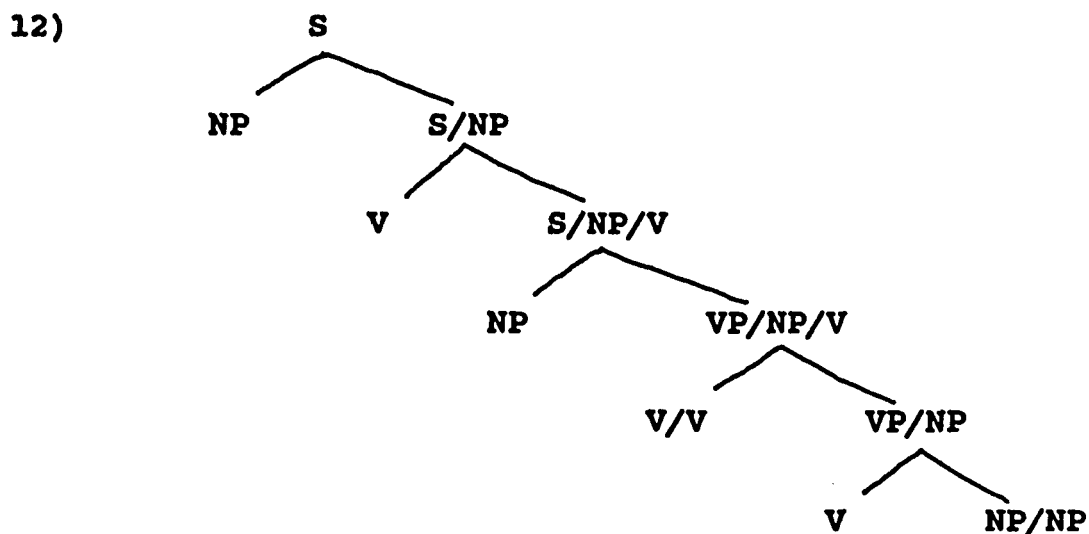


However, this kind of analysis has been argued against, because (10a) cannot conjoin with an object gap question (see Gazdar, 1981) and because it does not license

parasitic gaps (see Chung & McCloskey, 1983). Jacobson (1987) points out another problem with this analysis of SAI. As she shows, given their formulation of the HFC, GKPS have no way to rule out sentences like (11), where there has been extraction out of a subject phrase.

11) *Who did [a picture of e] bother Mary?

Fodor and Crain (in prep.) suggest an alternative approach to SAI in which it is analyzed as verb topicalization, which does not suffer from these problems since there will be a trace of the verb in its 'normal' position. Thus in place of (9), the structure will be (12).



Given this structure, the subject NP will control its sister VP. Therefore, subject-verb agreement and nominative case assignment can proceed as in non-inverted constructions. Also, since this tree so closely resembles the trees for non-inverted sentences, subject phrases maintain their

island status and there is no possibility of a 'string vacuous movement' analysis. Both of these are not possible due to the way in which the HFC governs the distribution of SLASHes in trees.

There are details of this analysis which remain to be worked out (e.g., the invertible verbs seem to require two different SUBCAT features in one tree, one for the syntactic context where the lexical verb occurs and one for the syntactic context of the trace), but this analysis does have the nice property that, in conjunction with the HFC, it predicts the boundedness of topicalization just in case it is a head (such as the verb) that is topicalized; the HFC will force the [SLASH V] feature onto the first head encountered, thus preventing it from passing further down through complement VP or S categories.⁹

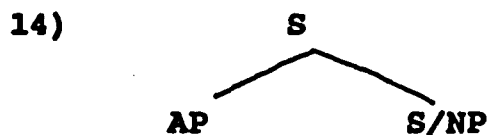
4.1.3.3 A top-linking problem. A third problem with the CAP stems from the top-linking rule that GKPS use for topicalization and WH-movement. It is given in (13).

$$13) S \rightarrow x^2, H/x^2$$

This rule could license a tree like that in (14), which GKPS say (p. 90) is ruled out by the CAP, essentially

⁹Note that this effect is quite similar to the effect of the Head Movement Constraint of Baker (1985) and Travis (1984). This is interesting in that it follows directly from the HFC (it needs no stipulations), which was not formulated to account for head movement.

because the value of the SLASH feature is not the same as its XP sister.



However, since the CAP only applies in control situations it does not apply here. For the AP to control the S/NP, the semantic type of the S/NP must be (roughly) a function from AP denotations to S denotations, and it is not (it is a function from NP denotations to S denotations).

Therefore, as Jacobson (1987) notes, there is no control and the CAP is irrelevant. For the CAP to apply in trees projected from (13), the two instantiations of X must be the same. But it is supposed to be the job of the CAP to ensure that they will be the same. The circularity here is obvious: in this situation, the CAP has to apply in order to set up the control relationship that will guarantee that the CAP must apply.

As far as I can tell, GKPS have no way to rule out trees like that in (14).¹⁰ They could, of course,

¹⁰It could be the case that GKPS assume that control always exists where a mother has exactly two daughters. This is one possible interpretation of the following quote.

There are two subcases [of control] to be considered. The first of these . . . arises when there are only two categories introduced by the rule. Such a pair of categories will nearly always combine semantically as function and argument. .

stipulate that in any one ID rule, all instances of X must be instantiated with the same features. This stipulation is the usual convention for interpreting variables in rules; all occurrences of the same variable within any rule are interpreted identically. However, GKPS are very clear and precise in their use of X. They state "X will be used to stand for the maximally underspecified category, namely {}, and is never a variable" (p. 27). Clearly it would be a very non-standard convention to stipulate that any occurrences of {} in a rule must be instantiated identically.

It seems that GKPS want to avoid variables because of their position on the underspecification of categories. If a category can be unspecified for *some* feature, then there is no natural way to block its being unspecified for *all* features. Thus, GKPS must accept the maximally underspecified category as a legitimate category. Once this is done, it seems quite strange indeed to have to

. . .

(GKPS p. 85)

But this interpretation would entail that control is a widespread phenomenon, given that 44 of their 65 rules have two daughters. Given their rules, it also entails, for example, that there is a control relationship between a VP and its sister ADVP or between an S and its complementizer. It seems more likely that this passage was intended to say only that the first kind of control occurs in a binary branching tree where one sister is a function from the other sister to the mother, and not that this is always the case in binary branching trees.

acknowledge also the existence of variables, which are identical to the maximally underspecified category, except that the various occurrences of a variable in a rule are linked in some sense. Thus, a theory of grammar should not allow both variables and maximally underspecified categories. GKPS cannot avoid the latter, therefore they shun the former.

Jacobson (1987) suggests that the appropriate stance is ". . . as was assumed in the earlier literature, the extraction rule itself must stipulate that the dislocated constituent matches the category of the gap" (p. 411). Fodor and Crain (in prep.) propose that this is a universal default; see the next chapter for discussion. If so, then the CAP need not apply in (14). That is, this is a job that the CAP cannot and should not do.

4.1.3.4 Connectivity. Jacobson (1984) discusses a relationship she calls "connectivity". That is the relationship whereby an 'extracted' element behaves in some ways as if it were in the extraction site. She distinguishes two types of connectivity: weak connectivity, in which there is evidence that a node of the proper syntactic type at the site of the extraction; and strong connectivity, where the filler has all of the syntactic properties that would be required of an element

in the gap position.¹¹ She uses (15a) and (15b) as examples of weak connectivity and (15c) (for those dialects which distinguish who and whom) and (15d) as examples of strong connectivity.

15) a.	What did Sue fix?	WEAK
b.	What did Sam put the book on?	
c.	Whom/*who did you see?	STRONG
d.	On what are you relying?	

The classic arguments for the TG movement analysis of these structures have been based on connectivity. Arguments based on subcategorization have used data like (15a) and arguments from phrase structure rules have used data like (15b). These arguments require that at some level, an NP node is sentence final, but they do not necessitate movement. However, in (15c) and (15d), the initial WH-phrases act as if they were in the extraction site. Thus, the argument is, that there must be some level of grammatical description at which these WH-phrases *are* in the extraction site.

Jacobson points out that a theory that uses empty categories or traces will be able to account for weak connectivity. She also points out that for a theory which accounts for unbounded dependencies with a mechanism like

¹¹This is effectively, in transformational terms, the difference between binding and movement. A movement analysis entails strong connectivity while binding requires only weak connectivity. However, there are problems here for a transformational theory since, as (15) shows, not all movements entail strong connectivity.

GPSGs SLASH and which takes categories to be feature bundles, strong connectivity reduces to weak connectivity. That is because the X's in a category of the form X/X must be identical. Any requirement placed on the extraction site must be met by X and that information will be passed along the chain of /X's (just like major category information). Then the extracted element, which must also be identical to X, will of necessity fulfill the requirement placed on the extraction site.

Jacobson (1987) points out that GKPS use the CAP to account for the strong connectivity in WH-extraction and topicalization constructions (these are introduced by the top-linking rule in (13)), and also to capture the strong connectivity in subject extraction (that is, STM 2) constructions. But to accommodate both, they have to complicate the statement of the CAP by making inherited SLASH, if it occurs on a category, the control feature of that category (this is to ensure that the VP in subject extraction (STM 2) constructions 'agrees' with the NP which licensed the introduction of the SLASH). (STM 2 will be discussed in detail below.) This then leads to problems with tough-constructions (which exhibit only weak connectivity) and with the trace-introduction principle (STM 1). Briefly stated, taking inherited SLASH to be a control feature forces GKPS to analyze tough-sentences like (16) as containing an ADJ with a sister S (an S-bar in TG).

16) John is easy for Mary to hate.

The widely accepted analysis (having an ADJ with PP and VP sisters) is not available to GKPS due to the interaction of SLASH and AGR (this is discussed in more detail below).

Identifying inherited SLASH as a control feature causes complications with trace-introduction in that STM 1 then cannot mention SLASH. STM 1, given in (17), instead uses the feature NULL.

17) $X \rightarrow W, X^2$

↓

$X \rightarrow W, X^2[+NULL]$

The use of 17) to account for the distribution of gaps requires two additional stipulations. First, $[+NULL]$ and SLASH must be correlated (GKPS accomplish this with the FCR in (18a)) and second, the lexical entry for the empty category must be as in (18b).

18) a. $[+NULL] \supset [SLASH]$
 b. $\alpha[+NULL]/\alpha$

As a lexical entry, (18b) is unique in two distinct ways: it must be specified as $[+NULL]$; and it is of the schematic form α/α . As an alternative, Jacobson proposes the metarule in (19).

19) X --> Y, W

↓

X/Y --> Y[+NULL], W

This analysis is simpler in that it does not require either of the stipulations in (18). But it does require giving up the identification of inherited SLASH as a control feature.

Given these problems, Jacobson concludes, "Surely, then, the CAP is not what is responsible for the connectivity effect." (p. 411). This seems very reasonable when we consider the various constructions in (1) and the connectivity they evidence. As noted above, there is strong connectivity in top-linking (WH-movement and topicalization) and with STM 2, but only weak connectivity in tough-constructions, and apparently no connectivity at all in subject-verb agreement, subject control, object control, or in constructions with expletive or non-nominal subjects. Again, this brings into question the view that there is a generalization to be captured across the phenomena in (1). Even if the matching that is required across these various constructions is the same, it should be clear that different features are required to match in the different constructions (e.g., *all* features in top-linking, but only NFORM and NUM for expletive subjects).

4.2 Data Covered by the CAP

Recall that the CAP is formulated to handle the constructions in (1), repeated here as (20).

- 20) a. Subject-verb agreement (and nominative case)
- b. Object control (and SAI)
- c. Expletive subjects
- d. Non-nominal subjects
- e. Top-linking
- f. Subject control (and Raising)
- g. SLASH Termination Metarule 2 (STM 2)
- h. Tough-constructions

Although there are some similarities that cut across all these constructions, there are also some considerable differences. As Jacobson points out, they differ with respect to connectivity. Connectivity, then, cannot be the generalization that the CAP is to capture. What, then, is this generalization about? The claim in GKPS is that the CAP captures generalizations about three things: the syntactic configuration where the relationship is defined; the features used both to define the relationship and over which 'agreement' must hold; and the path through the tree that this information travels. I will consider the relevant constructions one at a time, pointing out the differences and similarities in these three areas as I proceed. I will argue that verbal agreement and control can be analyzed more simply if they are kept separate and that the other constructions can (and should) be handled either in the lexicon or with ID rules.

4.2.1 Verbal agreement

One of the central facts that the CAP is to deal with is verbal agreement. It seems to be pretty much universally true that VPs agree with their NP sisters;¹² and verbs, if they show object agreement, agree with their NP sisters.¹³ This seems to be the default or unmarked case (see below for a discussion of subject control, which differs from this). The features that are important to verbal agreement are person, number and gender (see Lapointe, 1979 and Pullum, 1985). Case is also sometimes taken to be an agreement feature, usually in discussions that conflate verbal agreement and concord (see Moravcsik, 1978). However, Case seems to be different from verbal agreement in that verbs which take non-nominative subjects are morphologically like those which take nominative subjects. In fact, the distribution of Case behaves as if it is controlled by the SUBCAT feature, so I will assume for now that it is (see Pollard and Sag, 1987, the discussion below of expletive and non-NP subjects, and

¹²I will set aside here questions about subject-verb agreement in languages with no VP (such as Warlpiri, see Hale, 1973, 1983).

¹³As noted in the previous chapter (footnote 15), AGR should probably be split into two distinct features: SUBJAGR and OBJAGR. In the following discussion, I will ignore this distinction unless it is relevant to the point at hand.

Chapter Five).¹⁴ What I am proposing, essentially, is that the value of Case is totally free syntactically: its value is determined by subcategorization, which is essentially lexical.

No harm comes from this proposal, but we can then eliminate from the syntax the two FSDs in (21), which GKPS use to help regulate the value of the feature CASE.

- 21) a. [+N, -V, BAR 2] = [ACC]
 b. ~[NOM]

ACC seems to have been chosen as the default value for NPs largely because it occurs more often and in less restricted environments than NOM (at least in English). Assuming subcategorization for CASE, we could make this a default that occurs in the lexicon, but rather than making it about NPs, we could make it a property of verbs and prepositions that are subcategorized for NP objects. That is, we could take them to be subcategorized for ACCusative NP objects. We would need to change GKPS' version of subcategorization, which Jacobson (1987) has argued is independently needed, so that we can generalize over items that take objects. The version of subcategorization being assumed here (i.e.,

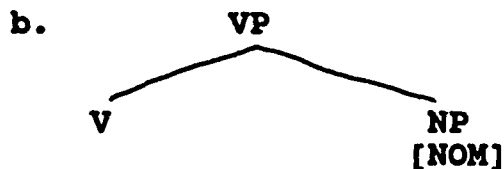
¹⁴This may create a problem for treating concord within NP in languages like Latin, where determiners and adjectives seem also to agree in case. Perhaps case is an agreement feature in some languages and not in others. That is, it is a parameter along which languages may vary. Or, perhaps it is just that some languages lack the morphology to express concord. I will ignore this complication here and simply assume that case is not an agreement feature.

a version like that proposed in Aspects, as opposed to subcategorization by lexical ID rule number) will allow us to generalize over lexical items that require objects. The proposal to eliminate GKPS' theory of subcategorization seems quite reasonable in the face of languages like German, where different prepositions or verbs can 'govern' different Cases. Surely, this is a lexical fact, so let us put it in the lexicon.¹⁵ The GKPS strategy seems to have been to make syntactic defaults and let lexical specification override them. I am proposing that some of the defaults reside in the lexicon and they are then overridden in the lexicon. The syntax is totally unconcerned, it generates trees for all possibilities, but some will not be used if the appropriate lexical items do not exist in the language.

This would mean then that a rule like (22a) could project a tree like that in (22b), abstracting away now from GKPS' system of subcategorization.

¹⁵ Pollard and Sag (1987) also modify the GPSG analysis of subcategorization in several ways. One of their extensions is to allow lexical items to subcategorize for the CASE of their NP complements, as well as for the VFORM of verbal complements and PFORM of prepositional complements. I will follow their lead in this.

22) a. VP --> H, NP



But, this tree could never be terminated in English, so no ungrammatical sentences would be generated. That is to say, no verbs in English are subcategorized for NOMinative objects. The same mechanism that will stop truly intransitive verbs from being inserted into this tree will also stop any transitive verb from being inserted, since we are assuming that they are subcategorized for accusative objects.

GKPS must also include N, V, and BAR (and other HEAD features) in their treatment of 'agreement' because they want the same mechanism to handle top-linking, and also to handle the possibility of S and PP (and AP?) as subjects. However, by restricting the work to be done by the CAP, one can also limit the features that it affects. In order to limit the erroneous predictions made by the theory, it is essential to reduce the features that are taken to be relevant to agreement. A theory which includes N, V, BAR, and other HEAD features among agreement features predicts, among other things, that the agreement morphology on a controllee could be different dependent on the category of the controller. This does not seem to occur in natural languages. In English, and in a number of other

languages, the morphological marking on verbs with non-nominal subjects is always identical to the marking for a (usually third person singular) NP. Agreement (at least as morphologically realized) seems to be based on the PERSON, NUMBER, and GENDER markings in NPs only (where GENDER includes information about noun classes, as in the Bantu languages).

Now let us consider how the AGR feature passes down through the tree beneath the actual agreement configuration that the CAP is concerned with. AGR must follow the path of HEAD features, as a minimal requirement, since it is morphologically marked on the lexical V head of VP. If we suppose that concord within an NP is actually a form of agreement (with N¹ controlling the ADJ or DET that "agrees" with it), as GKPS acknowledge might be the case (pp. 88-91), then most nodes within NP (at least DET and ADJ as well as the projections of N) must have the same AGR specification. We might assume that they all do, even though it is not morphologically marked on every category. This is different from the minimal requirement on AGR noted above: the HFC would only force AGR to be specified on all projections of N, if it were on any of them; it would not involve the DET or ADJ categories at all. What I am suggesting here is that AGR is in fact specified on *all* constituents of NP, not just the head.

Consider now copular constructions such as (23).

- 23) a. John is a doctor.
b. *John is doctors.

This shows that AGR must also be specified on more than just the head V in VP, at least in this case. The simplest assumption to make, then, is that within VP as well as NP, AGR must pass down to all nodes. We will call the principle which passes AGR down in this fashion the "All Daughters Principle (ADP)".

Note that this is a change from the theory proposed by GKPS. It is unclear exactly how GKPS planned to handle copular constructions like (23). It seems as if they had in mind the following analysis. They list the rule in (24) as the ID rule for this kind of construction.

- 24) VP[+AUX] --> H[7], XP[+PRD]

The CAP will ensure that an uncontrolled predicative category (e.g., the object NP in (23)) will have the same AGR feature as its mother. (23b) is ungrammatical because a plural NP cannot agree with a singular NP. However, there are at least two problems with this analysis. First, AGR cannot occur on an NP or PP, according to the FCRs that GKPS postulate. AGR can only occur on [+V] categories (that is, only verbs and adjectives). Here we have an uncontrolled predicative category that the CAP should give an AGR value to, but it cannot.

Even if that problem is resolved (perhaps by claiming that AGR should also occur on all [+PRD] categories), there is another problem that still remains. It is not enough for the NP to have an AGR feature duplicating the mother's AGR feature; that NP must have as its specifications for PERSON, NUMBER and GENDER the same values that the mother has in its AGR. This is similar to the situation that holds between a subject NP and its sister VP. In some sense, the NP contains the 'real' feature specifications; in the VP they are contained in a category-valued feature (AGR), that is, they are embedded one level. Correspondingly, the relevant feature specifications are embedded within the AGR feature on the mother (VP) node, but the 'real' specifications must occur on the daughter (NP) node. Therefore, without some revision, GKPS cannot properly describe the copular construction in English.

It seems as if part of the solution is to take AGR not to be a category-valued feature (as GKPS do) but rather to take it to be a class or set of features, like the HEAD or FOOT features (as was mentioned above in 4.1.3.1 with regard to the problem of nominative case assignment using the CAP). If AGR is a class of features, then there will be no problem stating a principle that will equate this class of features on VP mothers and NP daughters in the copular construction (although stating exactly the right

principle may not be easy). GKPS use AGR as a category-valued feature to facilitate their analysis of the tough and subject 'extraction' (STM 2) constructions in which AGR features interact with SLASH features. However, I will argue below that both of these analyses are independently flawed. I will assume, therefore, that AGR should be taken to be a class of features and not a category-valued feature. In the following discussion, I will continue to use the term "AGR" as GKPS do, and I will use the term "AGR features" to refer to the set of features that nodes can show agreement on.

GKPS essentially propose that there are three kinds of features: HEAD features (which must be on the mother and the head daughter); FOOT features (which must be on the mother and at least one daughter); and non-propagating features (which simply sit on some node and do not 'pass' around the tree). What I am claiming is that there is also another class of features, which GKPS merge with the HEAD features. These are the AGR features and they must be on the mother and on *all* daughters (to the extent that that is possible).

There are 'barriers' to the passing down of agreement features (just as there are 'barriers' to the passing down of SLASH, etc.). Of course, agreement feature specifications may be 're-started' by some node inside the

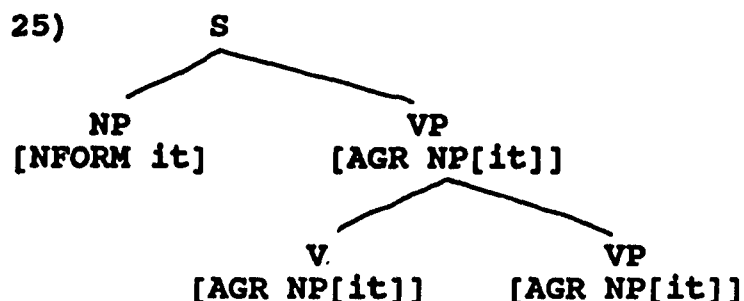
barrier. In particular, every NP node has its own features even if it is inside a larger NP or a VP. As discussed in Chapter 3, S should not bear AGR features and will act as a 'barrier' to agreement. It also seems very likely that PP should be a 'barrier' to this spread of AGR features, given that it is not uncommon in natural languages for a preposition to 'agree' with its sister NP, but uncommon (or even impossible) for it to show agreement with a containing NP (D.T. Langendoen, personal communication).

I will return to a discussion of the ramifications of my analysis of agreement below, and will turn here to a discussion of the other phenomena listed in (20).

4.2.2 Irregular Subjects

The distribution of expletive subjects looks as if it might be a case of subject-verb agreement. That is, only certain verbs license the occurrence of expletive subjects. However, as pointed out in 4.1.3.1, if we want to subsume their distribution into agreement, we must then assume that NFORM is one of the values of AGR. An expletive subject can occur in either of two syntactic configurations: i) as the sister to a VP whose head licenses expletives (this is the normal syntactic configuration for subject-verb agreement); or ii) as the sister to a VP which contains an uncontrolled VP (which can itself contain an uncontrolled VP, etc.) which licenses

expletives (formally, this is like subject control; more traditionally, this is the "Raising-to-Subject" construction). Case (ii) is illustrated in (25).



The first of these cases involves exactly the same syntactic relationship as is relevant to 'normal' person and number agreement in English; so it would be natural for the same mechanism to handle both. However, the second case is syntactically quite different from 'normal' agreement. Also, the relationship between an expletive subject and a verb is unlike normal subject-verb agreement in that it is not morphologically registered on the verb (the verbal inflection is always third person, singular or plural). Rather, it is a matter of verb selection, similar to the subcategorizational relation between a verb and its internal arguments. Thus, instead of subsuming the distribution of expletive subjects under subject-verb agreement, we might assume with Fodor (1983, 1990) and Pollard and Sag (1987) that verbs can be subcategorized for subjects as well as objects. A verb will then take an expletive subject if it is subcategorized for one and not otherwise.

Subcategorizing for subjects, especially expletive subjects, goes against the generally accepted position that expletives only occur in subject position *because* they can only occur in uncategorized positions. However, Postal and Pullum (1988) argue that this claim is wrong. They propose instead that expletives can occur in any position for which lexical items (verbs, prepositions) are subcategorized. Note that some such assumption is unavoidable in GPSG: expletives must be able to occur in subcategorized positions on any theory which does not posit the existence of an element like the PRO of GB theory, which is the syntactically present yet phonologically null subject of infinitivals. Consider (26).

26) John believed there to be unicorns in the garden. Given the lack of syntactic subjects for (bare) infinitival VPs, there is no option but to assume that there in (26) is in object position and thus in a subcategorized position.

Assuming subcategorization for subjects also allows for a straightforward analysis of the "quirky Case" assignment to subjects that some verbs in languages like Icelandic exhibit. By a lexical default, verbs will be subcategorized for NOMinative subjects, but a verb will have a "quirky" subject just in case it is subcategorized for one. We have also now eliminated the need for N and V in the inventory of features relevant to agreement, which

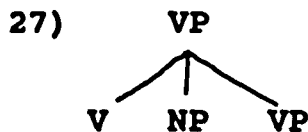
the GKPS analysis of non-nominal subjects requires. A verb will take a non-nominal subject (or an expletive subject) just in case it is subcategorized for one, not because it 'agrees' with one.¹⁶ (Note again that the verbal inflection for PP or S subject is always third person singular NP inflection.) It will also then be possible to eliminate the BAR specification from the set of agreement features (GKPS included BAR in this set to insure that subjects and extracted elements were phrasal), on the assumption that universally, non-head daughters are BAR 2 (or undefined for BAR, like SPEC).¹⁷ Something like this assumption must be made in all theories since it seems to express a true generalization. (We will return to this assumption again in section X.Y.Z of this chapter and the following chapter.) So the set of features relevant for agreement can remain PERSON, NUMBER, and GENDER.

¹⁶Pollard and Sag (1987), like GKPS, conflate Case assignment with subject-verb agreement. However, unlike GKPS, they account for both with their notion of subcategorization. Thus, for them, a verb like walks is subcategorized for a NP subject that is third person, singular and nominative. Although this is an interesting approach, I do not follow them in this, as I want to maintain a more standard view of subcategorization.

¹⁷Notice that this assumption makes the verb topicalization proposal of Fodor and Crain a marked construction. That is, it would introduce a non-head daughter that is not BAR 2. This is not an unwelcome result, since verb fronting does in fact seem somewhat uncommon across the world's languages.

4.2.3 Object control

While object control requires all of the features that subject-verb agreement does (i.e., PERSON, NUMBER, and GENDER), it also requires information about the referential index of the controller¹⁸ and all semantically potent features (like CONJ), but not location-relevant features (like PFORM), for example. Object control is lexically governed in that it can only occur when a verb is subcategorized for both an NP object and an (infinitival) VP complement. The object control construction has a structure like that in (27).



Object control, which, for GKPS, is indicated by having identical AGR specifications, holds between an NP and its sister VP. This is very similar to the normal syntactic configuration for subject-verb agreement, except the NP and VP are arguments to V rather than the only daughters of S. However, the features that are relevant are different. In particular, the description of object control requires information about the referential index (which GKPS do not

¹⁸It seems that in either object or subject control, it is not enough to say that the 'understood subject' of the controlled VP is a third person plural NP. The grammar must also indicate that the explicit NP and the understood subject referentially identical. This is the purpose of referential indices.

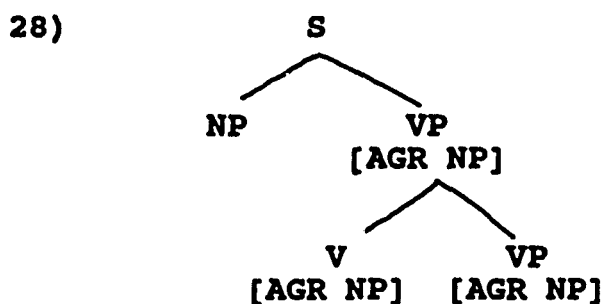
discuss) and NFORM of the controlling NP, as well as PERSON, NUMBER and GENDER. Without the referential index, the appropriate anaphoric binding relations into the embedded VP cannot be enforced. NFORM information is needed because an expletive need not be sister to the VP that licenses it, but the expletive must control that VP. It seems that since object control is a lexically governed phenomenon (which standard subject-verb agreement is not) and since a different set of features are used, it would be reasonable to treat object control separately from agreement.

Jacobson (1987) points out that the essential property of control constructions, (both object and subject control) is that there is *syntactic* agreement with the *semantic* controller. This generalization suggests that there should be a unifying principle (possibly lexically sensitive) that covers both cases, though it distinguishes them both from subject-verb agreement. But it is unclear what that principle should be. I assume that such a principle can be found, but will not discuss it further here.

4.2.4 Subject control

Subject control and tough-constructions (that is, (20f) and (20h), respectively) are both like object control in that they are lexically governed. Consider subject

control verbs. Subject control verbs come in two types: the try type, which is strictly intransitive, and the promise type which occurs both transitively and intransitively. The path that the control information takes with try type verbs would follow immediately from the ADP proposed above, if that information were encoded in an AGR feature, as shown in (28).



This is because there is no object for these verbs and thus there is only one NP that the embedded VP can get the control information from.

Verbs like promise, when they occur intransitively, will behave in exactly the same way as the try type verbs. If no object NP is present, then the path of subject control would follow from the ADP proposed above; the embedded VP will then 'agree' with the matrix subject. When an object NP is present, something else must be invoked to insure that the matrix subject will nevertheless control the embedded VP. The fact that the pattern of control here is different from other transitive verbs like persuade, force, etc., seems to just be a fact about this lexical item (i.e., it should be encoded directly in

subcategorization information of the lexical entry for the item, possibly linked by a redundancy rule to its semantic type). This lexical specification will then prevent the ADP from applying in this situation. This seems to be the right kind of solution since this is a lexically determined phenomenon. Also, the promise type subject control verbs seem to be rare across the world's languages, and are generally held to be the 'marked' case.¹⁹

Like object control, subject control requires more featural information than does subject-verb agreement. Minimally, subject control, like object control, requires information about the referential index and the NFORM of the controller, which verbal agreement does not.

To summarize this section, what we have seen is that although the path that subject control information takes through the tree is predictable based on the ADP, it requires different information than subject-verb agreement and should therefore be kept distinct from agreement.

4.2.5 Tough movement

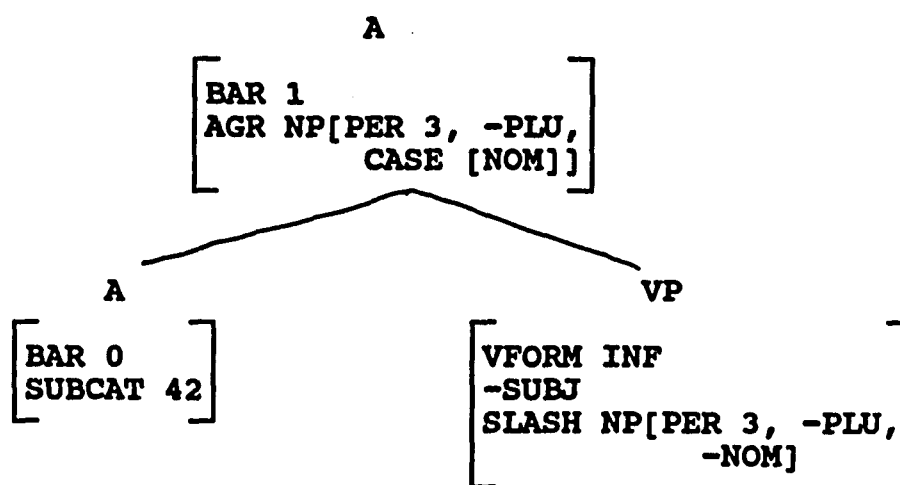
The tough-construction is like subject and object control in that it is lexically governed and it requires 'agreement' in values for PERSON, NUMBER and GENDER. Also, unlike subject-verb agreement but like control, SLASH, and

¹⁹However, this analysis still misses Jacobson's generalization that it is the semantic controller which the VP always shows syntactic agreement with.

other binding relations, tough-constructions also require information about the referential index of the NP. The GKPS analysis is essentially that the AGR feature being 'passed down' from the mother VP to an AP daughter must be translated into a SLASH feature on the V^2 daughter of an A^1 dominating a tough predicate. Consider (29b), one of the trees projected from the rule in (29a). This tree will have a SLASH on V^2 whose value is exactly the same as the value of AGR on the A^1 .

29) a. $A^1 \rightarrow H[42], V^2[INF]/NP[-NOM]$

b.

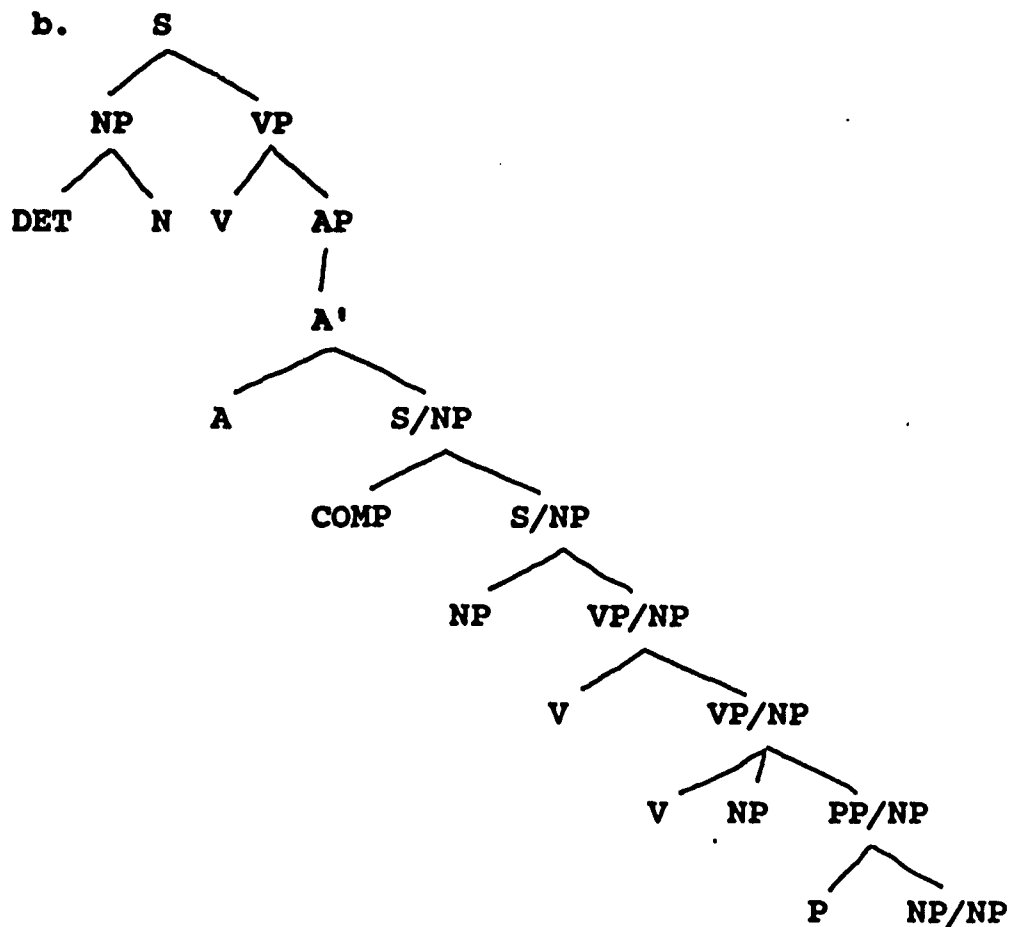


This follows from the CAP because the SLASH is inherited from the rule, and is therefore the control feature on the V^2 ; V^2 has no controller in this local tree, hence the value of its control feature (SLASH) must equal the value of the control feature of its mother (which is AGR), by clause (ii) of the CAP. Intuitively, the tough adjective takes the AGR specification of its VP mother (irrespective of whether that AGR is from normal subject-verb agreement,

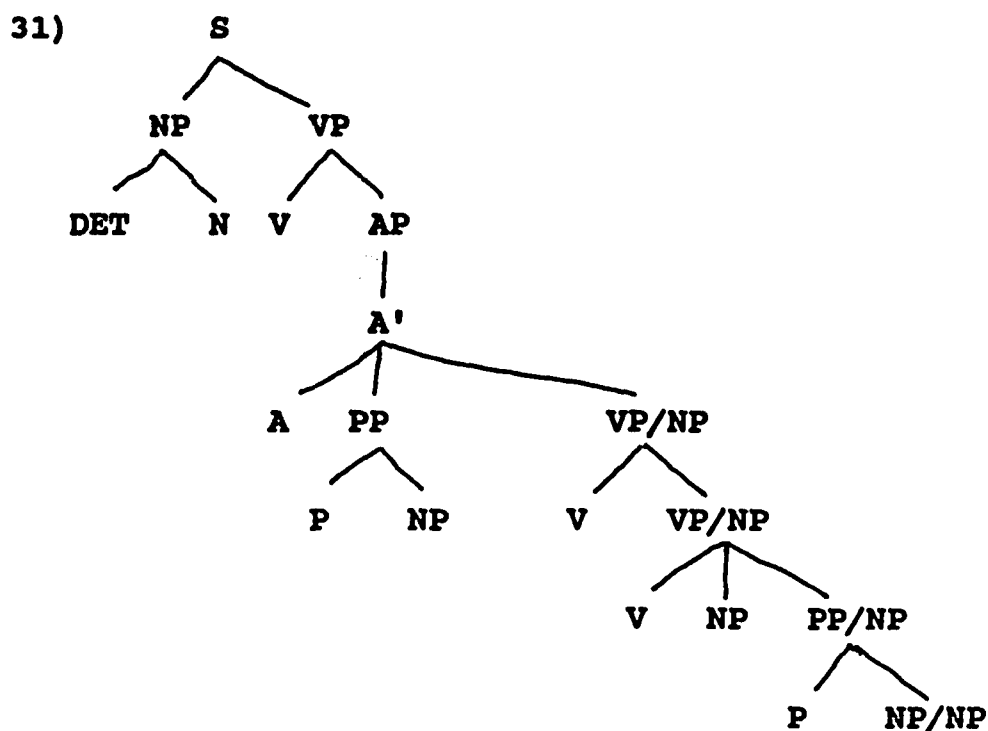
object control, or subject control) and 'turns it into' a non-nominative SLASH on its infinitival V^2 sister. In some sense, the AGR specifications are reinterpreted as SLASH specifications.

As noted above (in 4.1.3.4), Jacobson (1987) argues that SLASH should not be treated as a control feature, and that this analysis of tough-constructions helps to show why. She points out that GKPS are forced to analyze (30a) as containing an infinitival sentence, illustrated in (30b), in order to accommodate both the binding of the trace and the binding of the anaphor.

- 30) a. That fact is hard [_i for Mary to convince herself of e.



However, Bresnan (1971) and Lasnik and Fiengo (1974) have argued that this must be a sequence of PP and VP sisters to the tough-adjective, and this proposal is widely agreed to be correct. The appropriate tree is given in (31).



If (31) is the correct tree, then the SLASH would be the control feature on the VP (since it must be inherited). However, GKPS assume an analysis (essentially that of Pollard and Sag (1983)) of the distribution of reflexives and reciprocals wherein the anaphor has the same PERSON, NUMBER and GENDER specifications as its mother's control feature. Therefore, AGR must be the control feature of this VP/NP. A category cannot have two control features. Thus, GKPS have no way to get the reflexive to 'agree' with Mary, given their assumption that inherited SLASH is a control feature. That is: the VP needs both a SLASH feature (appropriately instantiated) to eventually cash out as a 'trace', and AGR (again, appropriately instantiated) to govern the occurrence of anaphors. The former must pick

up its value from the AGR specification in the higher clause. The latter must be assigned its value by a sort of object control process with Mary as controller. (We will overlook the problem that Mary is inside a PP and so does not satisfy the CAP criterion for controller; this is true of other cases of object control also.)

The problem Jacobson is pointing out is that SLASH cannot be assigned its value on the grounds that it is a control feature without a controller: the VP must have an AGR specification to govern anaphor binding and therefore AGR must be the control feature. This is clearly a serious flaw in the analysis. Also, Jacobson claims that the connection between an embedded infinitival VP, in a tough-construction, and its understood subject is like that between a head noun and a relative pronoun in that they must agree in terms of PERSON, NUMBER and GENDER, but not other features. Since GKPS apparently handle the connection in relative clauses with the semantics, it is not clear why they do not also do so in the tough-constructions.

However, there is another syntactic analysis available. In the revised GPSG system under consideration here, the tough relationship could be handled as in (32), which explicitly states the relation between the AGR feature specifications and the SLASH specification. (RI stands for referential index.)

32) A¹[PER α , B PLU, RI₁] --> H[42], (PP[PFORM for),
 VP[INF]/NP[PER α , B PLU, RI₁, -NOM]

This rule looks much more complex than the equivalent GKPS rule in (29a), this is because of the number of changes that are being introduced here. (32) not only makes explicit the AGR-SLASH relationship, it also includes the notion of a referential index (again, which GKPS ignore), and it corrects their misanalysis noted above.²⁰ A rule to generate this construction must be included in the grammar and there seems to be no reason not to include in the rule exactly this AGR to SLASH transformation. Since this rule introduces a lexical category, it follows that predicates will be lexically specified for insertion into this construction. The implicit claim here is that this construction is somewhat marked (because it is a top-linking rule for SLASH without an antecedent in the same local tree) and so it is appropriate that the rule is more complex than it would be if the construction were totally unmarked.²¹ Again, note that the features involved in the

²⁰Abstracting away from the changes noted in the text, and only making the relevant changes, (32) would be as in (i).

(i) A¹[PER α , B PLU] --> H[42],
 V²[INF]/NP[PER α , B PLU, -NOM]
 (i) looks less complex and looks more like (29a) than (32).

²¹Note that in this construction, if there is no PP, then the 'understood subject' is PRO-arb. This seems to be characteristic of object control. If there is no controller available, then there is a PRO-arb

tough-constructions are not just the features used in subject-verb agreement, but include also (minimally) CASE and the referential index.

4.2.6 SLASH termination metarule 2

Subject control and tough are each handled by one ID rule. STM2 (20g) is, as its name says, a metarule and therefore it expresses a generalization that holds across various lexical ID rules. This makes STM2 conceptually quite different, as it will apply to any lexical ID rule which has a FINite sentence as one of its daughters. Intuitively what STM2 does is to take a SLASH NP specification that is being 'passed down' a tree and 'turn it into' or reinterpret it as an AGR specification on the FINite VP. In other words, STM2 is in many ways the inverse of the tough-construction.

STM2 is as shown in (33).

33) $X \rightarrow W, V^2[+SUBJ, FIN]$

↓

$X/NP \rightarrow W, V^2[-SUBJ]$

For example, GKPS use the basic ID rule in (34a) to introduce a verb like believe.

interpretation (as in He didn't know what to do or To leave now would be wise). However, GKPS ignore cases of PRO-arb (that is, non-obligatory control), and I will follow their lead in not discussing it further.

- 34) a. VP --> H[40], S[FIN]
 b. VP/NP --> H[40], VP[FIN]
 c. Who does John [believe is a fool]?

Because (34a) is an ID rule in the grammar, and the grammar includes (33), the grammar will also include a rule like (34b) (which underlies the bracketed part of (34c)).

In an ID rule which is the output of (33), SLASH is the control feature on the mother (the VP/NP in (34b)) and AGR is the control feature on the V^2 daughter. The V^2 daughter is uncontrolled (the semantic types of the daughters do not combine appropriately for the definition of control); therefore its AGR specification must equal the SLASH specification on its mother. This transition from a SLASH feature to an AGR feature is a reflection of the fact that the 'extracted' NP (which is introduced higher in the tree, along with the SLASH feature, by a top-linking rule) is interpreted as the subject of this FINite VP. This analysis has merits, in that it handles subject 'extraction' without recourse to a SLASH in the lower S, which explains why this construction does not support multiple gaps (see Chung & McCloskey, 1983; Sag, 1983; Jacobson, 1987), why there is no phonological reflex of an empty category in the subordinate subject position (Schachter, 1984), and why structures with subject extraction cannot conjoin with structures that have other extractions (see Gazdar, 1981). However, it is not without serious problems (see Jacobson, 1987 for a discussion).

The problem that is of concern to us at the moment is that, as Jacobson points out, the filler and the gap in these constructions must be exactly identical (i.e., this is strong connectivity). For AGR to mediate the connection here between the 'missing subject' and its antecedent, AGR must be specified with all of the features that SLASH is specified for. That is, as opposed to the other cases of 'agreement' that we have considered so far, this situation requires marking not only for major category, PERSON, NUMBER, GENDER, but also for CASE, NFORM, and any other features relevant to NPs (including referential index, for binding purposes). The antecedent of the SLASH must be exactly the same category as could fill the subject position if there were one. This is unlike the situation for a construction like tough, where there need not be such total identity; e.g., the specification for CASE can be different. Therefore, it is not unreasonable to look for a non-CAP analysis of this phenomenon.

It is possible to pattern an analysis of STM2 on the analysis of the tough-constructions given above. The fact that STM2 is a metarule, as opposed to a regular ID rule like (32) above, is no reason to assume that the SLASH to AGR change cannot be specified as part of the rule, and that it must be a consequence of the CAP and the notion of 'control feature'. Essentially, metarules capture generalizations across lexical ID rules. Thus, if an ID

rule can express the SLASH to AGR transition, then a metarule could also.

In fact, assuming that Jacobson's arguments that SLASH is not a control feature are correct, then the SLASH to AGR transition cannot be a consequence of the CAP.²² Instead we need an explicit metarule, like (35).²³

35) $X \rightarrow W, V^2[+SUBJ, FIN]$

4

$X/NP[PER \alpha, \beta PLU, RI, CASE NOM] \rightarrow$
 $W, V^2[-SUBJ, PER \alpha, \beta PLU, RI,]]$

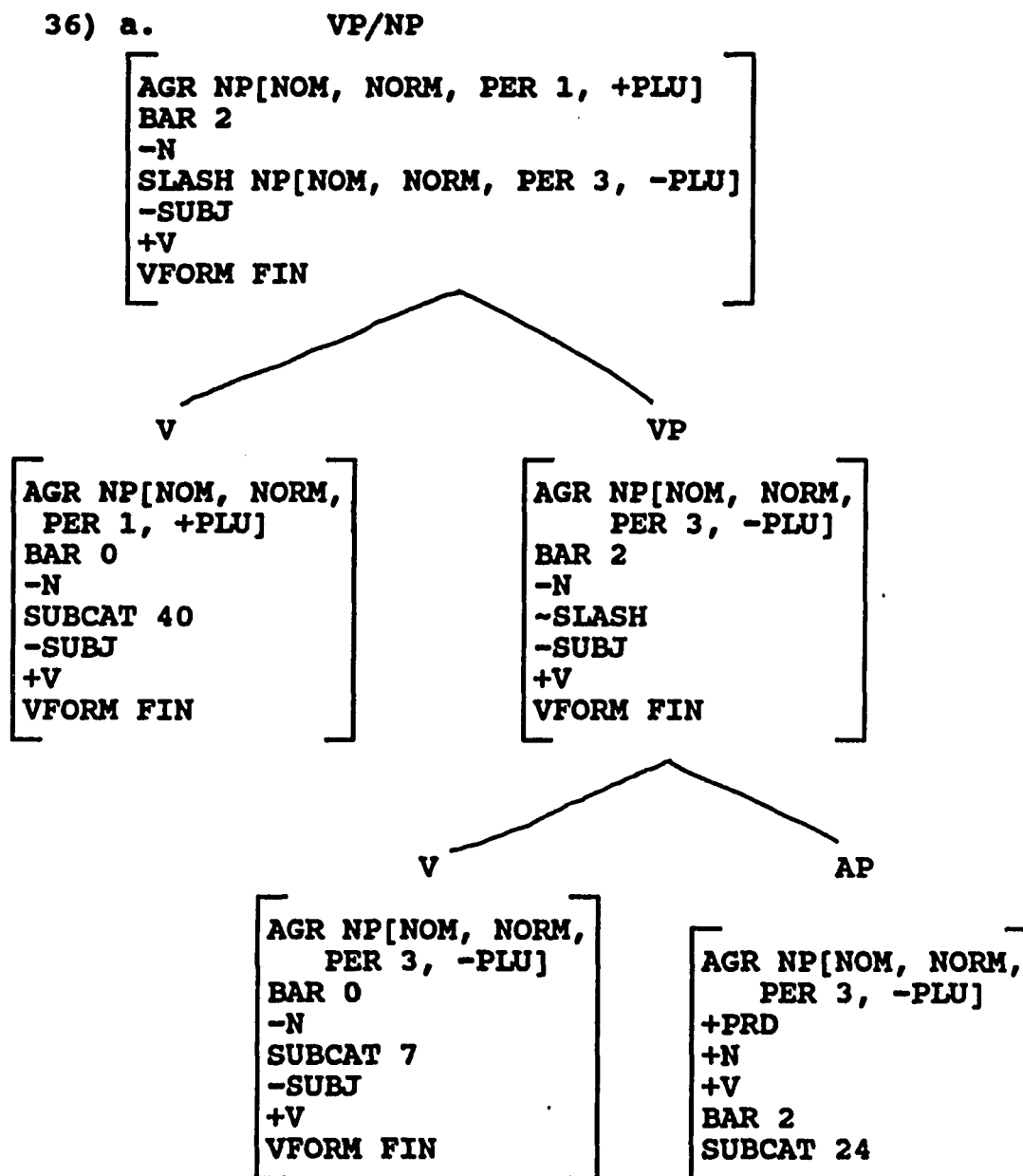
The difference from (33) are that we have to stipulate on the mother in (35) the features that the SLASH specification must include and on the VP in (35) that its AGR features are equal to the argument of SLASH on its mother.²⁴ This must be done so that the VP does not take

²²Jacobson's arguments are that this makes it necessary to distinguish inherited from instantiated SLASH, which then makes SLASH termination more complex than it needs to be. Independent motivation for for taking SLASH to be a control feature comes from top-linking and tough-constructions. Sections 4.2.5, above, and 4.2.7, below, discuss the problems with GKPS' analyses of these constructions.

²³This formulation still does not explain the exact identity required between the two positions. This identity is normal for SLASH (this was one of the motivations for a movement analysis in the first place), but except for this case, AGR does not need to be able to bear all of these feature specifications. It is unclear how to resolve this problem.

²⁴In the next chapter I will have to modify STM2 again, but that modification does not affect the discussion here.

on the same AGR features as its mother, which would give the wrong results. This can be seen in the tree in (36a), which underlies the bracketed part of the sentence in (36b).



b. The doctor who we [think is angry] left.

The daughter verb ('is') must agree with the singular 'displaced' NP who and not with the plural subject we. Since the CAP cannot be relied on to enforce this relationship, it must be expressed in the rule. The ADP also cannot account for this agreement pattern with its passing down AGR features from mother to all daughters, since this would put the wrong AGR features on is.²⁵ Once more, we see the appropriate results according to a standard sort of complexity metric. In order to insure the right agreement facts we must write them into the rule. This added complexity in the grammar seems right and proper, given that subject extraction is a marked construction in terms of its agreement as well as its SLASH not terminating a trace.

4.2.7 Top-Linking

We have now discussed all of the various phenomena that the CAP is designed to handle, except top-linking. There are two reasons not to subsume top-linking under agreement, as GKPS do. First, the paths that the AGR features and SLASH follow are arguably different. If agreement and SLASH passing are part of the same phenomenon, then they should, in an optimal system, traverse the same path through a tree, and that they should

²⁵See Pullum (1985) for an analysis of Achenese passives which also encodes marked agreement facts directly in a metarule.

be introduced in the same manner. However, I have proposed here that AGR features are transmitted to all daughters (except certain daughters at which AGR features are either universally blocked or blocked by a rule, as in (36)), while SLASH must pass to the head daughter if possible and *can* optionally pass to other daughters, but it must pass to at least one daughter (cf. GKPS; Fodor and Crain, in prep.; Jacobson 1987; and Chung & McCloskey 1983 for discussions of multiple gap constructions). Also, AGR features are (universally) blocked at an S node, and have to be re-started below it, but SLASH is not (universally) blocked at S (cf. 'Who [_s does John believe [_s that Mary saw e?']).²⁶

A second reason for not treating top-linking as agreement is that we would have to reintroduce N, V, and many other HEAD features back into the AGR features to account for the various extraction possibilities. To start the SLASH off with the right features, its argument must be an exact copy of its sister XP (but, see footnote 27 below and the discussion above of tough-constructions regarding marked exceptions). Any feature that the syntax can require to be on an unextracted constituent must be

²⁶In languages that do not allow extraction out of embedded clauses (e.g., Polish), it does appear that SLASH is in some sense blocked at the S node. In Chapter Five, however, it is proposed that this is not blocking, but rather that there are no 'bridge' verbs in these languages. That is, there are no verbs that are subcategorized for an S/XP complement.

represented in the SLASH, in order to determine the appropriateness of an extracted constituent (in cases of strong connectivity). For example, consider (37).

37) On which table did John put the book PP/PP?
The verb put requires a locative prepositional argument, therefore, the PP/PP must be marked [+LOC]. As far as I can tell, the same thing can be said for (almost) all features (see Jacobson (1984)). By contrast, Vs only show morphological agreement with NPs and only in terms of PERSON, NUMBER, and GENDER. When there is a non-NP argument, the AGR value is third person singular (the unmarked case).

As noted in an earlier section (4.1.3.3), top-linking is unique in that we must assume that the two occurrences of X in the rule are identical in order for the CAP to apply. But it is the CAP's job to insure that those two occurrences of X are instantiated identically. So, our assumption of identity makes the CAP redundant here. We should abandon either the assumption that identity is necessary for the CAP to apply, or the use of the CAP in this instance. Jacobson (1987) argues that identity is necessary, and hence that top-linking is not an instance of the CAP. If we don't assume identity, then the CAP need not apply and ungrammatical sentences could be generated (as noted above). In order to avoid this overgeneration,

we would have to expand the schematic top-linking rule into its various instances, possibly as in (38).

- 38) a. $S \rightarrow N^2, H[SLASH N^2]$
 b. $S \rightarrow P^2, H[SLASH P^2]$
 c. $S \rightarrow A^2, H[SLASH A^2]$

Having done this, there would necessarily always be a control relationship in top-linking trees and the CAP will determine that there is agreement between the two sisters in the rules, in that the SLASH takes on its sister's values. That is, if the CAP is involved in top-linking (i.e., top-linking is a form of agreement), then we lose the generalization that (virtually) any BAR 2 category can occur in this position (that is, it can be 'extracted'). The grammar must contain the individual rules and not the rule schema in order to subsume top-linking under agreement, but that means losing the categorial generalization. If we want to keep the generalization, then the assumption that the two X^2 s are identical is required and the CAP is not. Thus, the CAP must not be involved in top-linking; top-linking is not a form of agreement.

Fodor and Crain propose this kind of analysis. They argue that it is redundant to specify both values in a top-linking rule and that the identity will follow from a markedness convention or default specification. I will follow them in assuming that it is universally unmarked in the top-linking configuration for the value of a highest (=

orphan) SLASH to be that of its sister XP, so it does not have to be specified explicitly in top-linking rules. The principle that F&C propose is obviously less general than GKPS' CAP but it is a quite plausible candidate as an innate principle, and it is better than the CAP at handling exceptions to top-linking.²⁷

Given the markedness assumption of Fodor and Crain, all that we have to explicitly write into a rule is that a SLASH occurs: its value will follow from the universal principle. This will start SLASH out with the right value and then the HFC and/or the FFP will force it to continue down the tree appropriately. This would mean that the GKPS rule in (13), repeated here as (39a) would become the rule in (39b).

²⁷Note that although it is the unmarked case for the XP and the SLASH to have the same feature specifications in this construction, this proposal allows for marked exceptions, while the GKPS analysis does not. In fact, Kaplan and Bresnan (1982) present data like (i) through (iv) that suggest that there are in fact exceptional cases of top-linking.

(i) That he might be wrong he didn't think of.
 (ii)*He didn't think of that he might be wrong.
 (iii) He didn't think that he might be wrong.
 (iv)*That he might be wrong he didn't think.
 As these examples show, the topicalized S-bar in (i) must terminate in an NP gap; an S-bar cannot replace the NP and an NP cannot replace the S-bar.

- 39) a. $S \rightarrow X^2, H/X^2$
 b. $S \rightarrow X^2, H/$

The rule in (39b) is formally simpler and will cover the same cases as the GKPS rule does and does not rely on the CAP.

4.2.8 Features needed

Consider Table 1. GKPS' implicit claim is that all the phenomena in (20) are sensitive to all of the features listed in Table 1. However, we have seen that this is just not the case. Only some of the features are relevant to only some of the phenomena. One could possibly make a case for N and V being important to all of these phenomena, and maybe stretch this to include PERSON, NUMBER and GENDER, but that is all. Some features (like ADV, INV, PRD, and AUX) are probably never relevant. Also, see the discussion in section 4.1.3.1 above about FOOT features never being relevant to the CAP, except for triggering where control occurs.

	A	B	C	D	E	F	G	H
AGR	Y	Y	Y	Y	?	Y	Y	Y
ADV								
AUX					?			
BAR	Y	Y	Y	Y	Y	Y	Y	Y
INV					?			
LOC					X			
N	X	X	X	X	X	X	X	X
PAST					?			
PER	X	X	Y		X	X	X	X
PFORM				?	X			
PLU	X	X	Y		X	X	X	X
PRD					X			
SLASH							X	X
SUBCAT			X	X	X	?	X	X
SUBJ	X			Y	X		X	X
V	X	X	X	X	X	X	X	X
VFORM	Y	X			X	Y	X	X
RE					?X			
WH					?			
NFORM		X	X	?	?X	X		
CASE	?				X		X	X
RI		X			X	X	X	X

A. Subject-verb agreement (and nominative case)

B. Object control (and SAI)

C. Expletive subjects

D. Non-nominal subjects

E. Top-linking

F. Subject control (and Raising)

G. SLASH Termination Metarule 2 (STM 2)

H. Tough-constructions

RI = Referential Index. (not discussed by GKPS)

X = any analysis probably needs feature; Y = GKPS analysis uses feature; ? = it is unclear if it is needed

Table 1

4.2.9 Summary

Summarizing the preceding discussion: I have proposed that "AGR" is a set of features that includes only PERSON, NUMBER, and GENDER (where GENDER includes specification of noun class membership, required for languages like Bantu, see Pullum 1985b). These are the features that lexical items show morphological agreement with. All other features seem to have no morphological reflexes in agreement situations.²⁸ In the lexicon, PERSON, NUMBER and GENDER are specified on Ns (and NPs). They must then also be on all projections of N in any tree into which that lexical N (or NP) is inserted. This is ensured by the ADP.²⁹ There are two processes occurring here. The first is what is left of the CAP, which I shall call "AGREE" in order to avoid confusion with GKPS' CAP. AGREE copies the agreement features onto V or VP from a sister NP. The second process I have called the 'All Daughters Principle' (ADP). The features that follow this

²⁸As noted above, the exact nature of Case concord is unclear. Therefore I am assuming that it is not an example of agreement.

²⁹It may appear that the ADP is applying here to copy feature specifications from the bottom (the lexical N) up (to the N-bar and NP levels). This is a misconstrual. The information flow, as enforced by the grammar, is essentially top-down. Although lexical items may require particular features in the trees in which they occur, that information does not percolate up from the lexical items; it must be there by free instantiation.

path through trees (the AGR features) are PERSON, NUMBER, GENDER. The ADP spreads them from the mother to all daughters that are not already marked for them.

'Agreement' (or "concord") is a widespread phenomenon. Everything within an 'agreement domain' (that is, everything dominated by a node that bears AGR features) must be marked identically for the AGR features (this is the ADP at work), unless they bear their own AGR features (like NP) or can bear no AGR features (like S). These AGR feature specifications will, of course, be vacuous if there is no morphology to express them. Subject-verb agreement is simply equality between AGR feature specifications on a VP and the AGR features on its sister NP, if such an NP exists. This is all that AGREE (= what was the 'CAP') has to say. These two principles will handle subject-verb agreement. The ADP will account for concord within NP as well. An NP must be marked for PERSON, NUMBER and GENDER if it is to dominate a lexical N (which is marked for PERSON, NUMBER and GENDER). The ADP then passes that information down to all of the daughter nodes which do not have independent AGR features. Thus, concord within NP is the expected case, but will only be overt if the language has morphology to express it.

Control (both subject and object) is a different phenomenon and must be handled separately. Expletive,

prepositional and sentential subjects are now handled by subcategorization, as argued in section 4.2.2; they occur only if the head verb of their sister VP subcategorizes for them. Following Fodor and Crain (in prep.), the tough-construction and promise-type subject control (see sections 4.2.5 and 4.2.4, respectively) are generated by marked rules that must be explicit in the grammar. In section 4.2.7 it was argued that top-linking follows from a universally specified default requiring that the SLASH value be identical to that of the sister XP. And STM2, as discussed in section 4.2.6, is a marked metarule that will capture a generalization about (the lexicon of) some grammars and not others. This mixed solution may not sound elegant, but we have seen that GKPS' CAP is not so elegant, either, and that it is empirically inadequate. Though it would be interesting if the same feature passing principles applied in all these constructions, there is no reason for thinking this must be so, especially if the principles that do apply are all universal and do not need to be learned.

4.3 The ADP

The ADP determines that once the VP has its value for AGR features, that value is passed down to all its daughters. Of course, the AGR feature values are not passed from the VP to a (non-predicative, referential) NP daughter, because this will have its own AGR feature

values. I will assume that S also blocks the transmission of AGR from mother to daughter. That is, an embedded sentence creates an 'agreement island' much like an NP. This can be established by a universal FCR like that in (40) which simply says that S is always undefined for the AGR features.

40) [+SUBJ] --> ~[PER ∨ PLU ∨ GEND]

This is independently motivated by the fact that verbs are not able to subcategorize for a sentence according to the AGR features on that S (as noted in the discussion above). Subcategorization can be sensitive to the VFORM of a sentence, and also to +/-SUBJ (and possibly PAST) but not to the PERSON, NUMBER or GENDER of its subject.

This analysis gives us the appropriate distribution of AGR features, which occur on (almost) all categories. All VPs that have an NP sister must agree on the AGR feature values with that NP and then that specification for AGR features spreads down to all the non-NP, non-S daughters of VP. A potential drawback to this analysis is that all nodes (except NP and S) are now potential 'agrees'. That is, almost every node in a tree will be specified for PERSON, NUMBER and GENDER, and hence could show morphological agreement. Thus there will be a great many unrealized AGR features in trees in morphologically poor languages like English. However, looking across languages we find that verbs, adjectives,

preposition/postpositions and even determiners (sometimes if not always) can show morphological agreement with nouns (or, on this analysis, with the NPs that contain them).

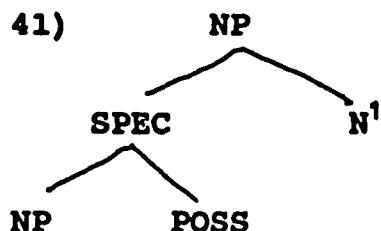
Since SPEC of NP (that is, determiners) can show agreement, one might wonder why SPEC of S (that is, COMP) does not show agreement. There is a simple reason for this, on this analysis. For GKPS, the mother of COMP is not S-bar but S. S is undefined for the AGR feature, therefore it cannot pass an AGR value down to its COMP daughter. Since there is no other place for COMP to get its AGR feature values from, it can have none.³⁰

4.3.1 Genitive Nouns

The analysis of agreement sketched here is probably too generous in its broadcasting of agreement features, just as the analysis in GKPS is too miserly with its agreement features. However, this analysis receives some support from the properties of genitive NPs in various languages. Klavans (1980, 1985) argues that in English, the genitive 's' is a phrasal clitic that attaches

³⁰It is interesting to note that Rizzi (1987) argues for agreement *in* COMP but not *of* COMP. He presents evidence that a complementizer may show morphological agreement, but only in what are essentially 'doubly filled COMP' situations. That is, he shows that a complementizer may show agreement with the trace of an element that has 'passed through' its COMP node (on a movement analysis). But, it is not agreement between COMP and S. This seems to be a marked phenomenon and its analysis remains to be worked out in PSG.

phonologically to the preceding NP. I assume that that analysis is essentially correct, and that it shows that there must be available, universally, a structure such as (41).



English also has a pronoun, which is lexically marked as genitive and does not bear the genitive clitic. I will assume that these are the two options that UG makes available to a language; genitive NPs may either be marked by a morpheme external to that NP, or be lexically specified as a genitive. Given the theory of agreement under consideration here and structure like (41), we would expect that the POSS clitic, if it showed agreement, would have to agree with the containing NP and not the NP which is its phonological host. In English there is no morphological agreement on POSS, it is simply 's'.

However, in a language like German, we find that the agreement facts for genitive pronouns are exactly what the ADP predicts. The internal NP can be any person, number or gender, but the possessive marker shows morphological agreement with the N', as is shown in (42).

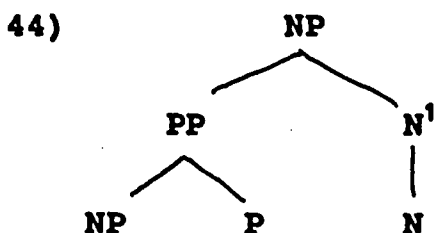
- 42) a. meines (my husband)
 b. meiner (my daughter)
 c. meines (my child)
 d. meiner (my parents)

French genitive pronouns also show the same pattern, where the pronoun shows agreement in number and gender with the head noun. Consider (43).

- 43) a. mon père (my father)
 b. ma mère (my mother)
 c. son ami (his/her friend)
 d. ses enfants (his/her children)

This could easily be explained by assuming that German and French have the same structure as English in this construction, and that as in English, the POSS marker and the internal NP are phonologically merged. Hindi and Romani show a pattern of agreement that is similar to this (Dana McDaniel, personal communication). The difference is that in these two languages, the external possessive marker is apparently a postposition, rather than a clitic.

Therefore we would have a structure like that in (44), where the postposition shows morphological agreement with the containing NP and not the sister NP.



Again, this is predicted by the system of agreement under consideration here, but is not easily explained by the system proposed by GKPS, because they would have to claim

that the N-bar in (44) controls its PP sister while the NP does not control its P sister. This seems like an improbable situation.

4.4 Ordering revisited

Since we have now changed the operation of the CAP, reducing its function a great deal, it is important to reconsider the ordering of the principles of UFI. The relevant questions are: does the proposed modification vitiate the simplification proposed in Chapter Three; and is further simplification possible?

4.4.1 The HFC

Notice that on the version of UFI that we are considering, the HFC and the FFP look very different from each other (as they did for GKPS and for Shieber). The FFP must ignore an inherited FOOT feature, and only instantiate FOOT features two or more at a time, on the mother and daughter(s). The HFC, on the other hand, must *not* ignore inherited HEAD features. Part of its motivation is to ensure that head daughters have the HEAD feature specifications that are inherited on the mother.³¹ Shieber

³¹In principle, this requirement should go both ways. That is, the HFC should require that the head daughter(s) have the inherited HEAD features of the mother *and* that the mother have the inherited HEAD features of the head daughter(s). However, in practice, in GKPS' rules, the head daughter(s) never inherit HEAD features that need to be instantiated on their mother.

deals with this by having the HFC copy HEAD features which are already present from mother to daughter . In particular, N, V, and BAR must always be copied down. The HFC must also 'freely instantiate' HEAD features, which are not yet specified, in pairs (or more) on the mother and head daughter(s). It seems, then, that there might in fact be two different things going on here, which have been conflated as the HFC. Let us call the principle that copies inherited HEAD features from mother to head daughter the 'PREHFC' and continue to call the principle that instantiates pairs of HEAD features the HFC. The important distinction here is that the PREHFC copies existent HEAD feature values while the HFC introduces values for HEAD features that previously had no value. One must question then whether both of these principles are needed or whether there is an unneeded overlap here.

It is the PREHFC that must operate in order to define the semantic types of the nonlexical categories which determine the control relationships that occur in local trees (i.e., it is the PREHFC which must precede the AGREE). The PREHFC is needed because of the use of the H notation in ID rules: the PREHFC fills in the information that H notation allows to be left out of rules. The other principles of UFI are different, however. They fill in

Therefore, in what follows, I will ignore this possibility.

information that we want to omit from rules in order to be able to collapse them together. Abbreviating a category to H never allows several rules to be collapsed, it only allows an individual rule to be written more succinctly.³²

The HFC, as opposed the PREHFC, instantiates default values for HEAD features on both the mother and its head daughter(s). Non-default values, as was seen in Chapter Three, cannot be freely instantiated but must be inherited from the licensing ID rule. However, if the default values were instantiated on the mother *before* the PREHFC applied then the PREHFC could copy the instantiated values from mother to daughter as well the inherited values, thus making the HFC unnecessary. Note that on this analysis, the PREHFC fulfills the intuitive idea originally motivating the HFC: the PREHFC is what ensures the featural identity between mothers and head daughters.

³²It is conceivable that H could be used in a rule which collapses two or more rules. For example, a rule like (i) would collapse the rules in (ii) and (iii), but notice that H is not necessary to the collapsing of the rules.

(i) [-N, BAR 2] --> H¹, NP

(ii) [-N, -V, BAR 2] --> H¹, NP

(iii) [-N, +V, BAR 2] --> H¹, NP

We could do this without the H notation, as in (iv).

(iv) [-N, BAR 2] --> [-N, BAR 1], NP

Note also that cross-category generalization over lexical heads would be incompatible with the GKPS analysis of subcategorization, which requires many specific lexical ID rules in order to provide many different subcategorization features. However, this is independently an unfortunate aspect of the GKPS theory.

Shieber's need to instantiate pairs of HEAD features has been eliminated. I will, henceforth, assume that this analysis is correct.

However, something still must put HEAD features on non-head daughters. As was the case with the free instantiation of HEAD features on mothers and head daughters, this can only be the instantiation of default values; non-default values must be inherited from the ID rule. Therefore I must assume that the principle that instantiates default values for HEAD features says, roughly, "instantiate HEAD features on the mother and the non-head daughters in a local tree". Once this has happened, the ordered system of principles can then apply to pass the appropriate information throughout the tree.

The principles that I am assuming, then, are:

- PREHFC: copy HEAD features from mother to head.
- AGREE: copy AGR features from controller to controllee.
- ADP: copy AGR features from mother to all daughters.
- FFP: instantiate FOOT features, on mother and daughter(s).

All of these principles obey the FCRs and the FSDs. That is, if doing their job would violate an FCR or an FSD, then they do not do it. Notice, that the FFP is freely instantiating features that are not already specified and the PREHFC, AGREE and the ADP are all copying. 'Free

instantiation' means, for the FFP, inserting a pair of identical feature specifications, one on the mother and one on a daughter. If that would violate an FCR or an FSD on either, or if one is already specified for that feature, then it is put on neither. The FFP puts on 'free' features (those not already specified in the tree by inheritance from the rule), while the other principles copy feature specifications that are already present. (In Chapter Five, the FFP will be brought more into line with the operation of the rest of the principles of UFI.)

While this approach might not be striking in its elegance, it does in fact seem to be descriptively adequate. The biggest problem with this analysis has to do with its profusion of AGR feature specifications and control. However, the AGR features do not seem to create a pernicious problem and we have seen that control is an extremely thorny problem for any phrase structure theory. As noted above, control lies at the interface of syntax and semantics, but without a level of deep syntactic structure to motivate empty subjects at surface structure, it is not easy for the syntax to reflect semantic relations. Further investigation into control is beyond the scope of this work, and so I will set aside these concerns. In what follows, I will assume the analysis of UFI sketched in this section.

4.5 Conclusion

From the preceding discussion, it is clear that the theory proposed in GKPS, as impressive as it was, must be modified. We have tried to remodel the GKPS system in various ways to reduce the complexity of it (following the lead of Shieber, 1986). But if we devise a simple and elegant system that is unlearnable for fundamental reasons, it is questionable to claim that we have advanced the field. Fodor and Crain (in prep.) argue, based on learnability considerations, for various changes in standard GPSG. We will see that in several cases, concerns of complexity-reduction and achieving learnability converge on the same revisions. It is the learnability considerations to which we will now turn our attention. Once we have made all the necessary changes for the sake of learnability, we will see that the complexity problems have disappeared.

CHAPTER FIVE

LEARNABILITY

Given the version of GPSG proposed in the previous two chapters, consider now what it would be to learn one of these grammars. Since we are following Shieber in taking FI to be actually filling in feature specifications and not a checking procedure as GKPS propose, the most uncomplicated conception of learning would be to simply reverse the process. That is, given that we are assuming the process of generation to be that of taking highly schematic ID rules and fleshing them out with the principles of FI, then we can take learning to be taking fully specified rules and stripping off the predictable or redundant features using the principles of FI in reverse. This is the proposal of Fodor and Crain (in prep.), but Fodor and Crain do not explore the details of its implementation.

5.1 A model of learning

Fodor and Crain (hereafter, F&C) are interested in the form of innate linguistic knowledge; that is, they want to know how linguistic knowledge is encoded in the brain/mind. As a research strategy, they adopt a very restricted theory (which posits a large amount of innate knowledge that is economically encoded) and propose to enrich that theory only when the data from linguistic description or language acquisition force them to. In particular, they propose that a grammar for a natural language consists of sentences which are encoded in a 'mental metalanguage'. Their idea is that the properties of this metalanguage can restrict the ways in which a linguistic fact can be mentally 'written down' and hence captured in a grammar. The mental resources available for grammar representation consist entirely of this metalanguage; if a proposition cannot be encoded in the metalanguage, it cannot be a part of the grammar for a natural language. The mental metalanguage has both a syntax and a semantics and F&C exploit both in restricting the sorts of statements that can be included in a grammar.

The theory that F&C propose is a formal theory in that it allows only formal principles to be innate and no substantive principles. A formal principle, for them, is one which represents a fact about the sentences of the

metalanguage (which in turn represent the grammars of natural languages), and does not represent a fact about the sentences of natural languages. For example, if the mental metalanguage contains no way to formalize a deletion transformation (a formal constraint), then it follows that natural languages cannot contain deletion transformations. Thus the universal properties of natural languages are seen to be indirect consequences of the mental machinery available to represent the grammar.

Formal principles are very different from substantive principles. F&C offer, as an example of a substantive principle, the Empty Category Principle (ECP) of GB theory (see Chomsky, 1986, Kayne, 1984, and references therein). In GB theory, it is not enough for a learner to know innately that the mental metalanguage contains a sentence expressing the proposition that empty categories must be properly governed, she must also know innately that it is true. Otherwise, the ECP will simply be one of the principles that a child is capable of formulating (i.e., it would be thinkable) and she may or may not include it in her grammar. In order to insure that it is included in all natural language grammars, the ECP must be innately flagged as true. The ECP states a fact about natural language sentence structures: this is a matter of the content (substance) of the

representation, and not a matter of its form. F&C do not allow this kind of principle to be innately specified in the mental metalanguage.

F&C suggest that their Formal Theory (FT) can do the descriptive work required of a linguistic theory, largely based on the achievements of GPSG. However, FT runs into problems when trying to account for language acquisition. F&C make two assumptions regarding the learning process, both of which are fairly standard and seem quite reasonable. They assume first that children do not have (systematic) access to negative data (this is the No Negative Data, NND, assumption), and second, that learning is guided by a formal simplicity metric (that is, a symbol counter). These two assumptions entail, for a formal theory, that grammars cannot contain constraints, overgenerating rules, or rules with a positive simplicity/generalizability correlation. I will consider these points one at a time.

It is easy to see that a child without negative data (that is, without information about what does not occur in her language) could not learn a grammar with constraints. Constraints are inherently negative; they state something that cannot occur in a language. In order to learn a constraint, a child must realize that X (some construction or category) cannot occur in her language. Without negative data this information will

not be available. Hence constraints cannot be learned: all constraints must be universal. The lack of overgenerating rules is a corollary to this.

If a grammar has rules that overgenerate, then it must also include constraints to limit that overgeneration. We have seen that our child cannot learn constraints, therefore these constraints would have to be universal. But, if the constraints are universal and formal, then the child could not posit the overgenerating rule since it would violate the universal constraint (it would be 'unthinkable', in F&C's terms). So, if she posits an overgenerating rule, it must be a rule which, in the target grammar, is constrained by parochial constraints, which, as we have just seen, are impossible. Therefore she must never posit an overgenerating rule. F&C assume that when adding a rule to her grammar, a child is required to add No Unmotivated Generative Power (NUGP). NUGP stops a child from adding arbitrary complexity to her rules/grammar. In this sense, she must be a conservative learner; she must postulate a new rule just in case her data force her to and only as complex a rule as the data demand.

The need for a negative simplicity/generality correlation comes from the formal simplicity metric, in conjunction with conservative learning. The simplicity metric will order potential rules and favor those with

less symbols. If a simple rule is more general (i.e., generates more constructions) than a complex rule, then when she is forced to pick the simpler rule (by the simplicity metric), she risks overgeneration and without negative data she could never recover from that mistake. Therefore, simple rules must be less (or at least no more) general than complex rules. This restricts the rule formats and the abbreviatory conventions that are available to FT. For example, context sensitive rules could not be learned by an FT learner. A context free rule that is identical to some context sensitive rule, except for its lack of context, will generate all of the (positive) data generated by the context sensitive rule, plus (usually) other structures. The simplicity metric will then force the choice of the context free rule (with its potential for overgeneration) and the context sensitive rule would never be learned. Therefore, assuming NND, there can be no context sensitive phrase structure rules in the grammars of natural languages; all phrase structure rules must be context free.

The relationship between complexity and generality in rules that the simplicity metric requires is the opposite of what is usually assumed in linguistics. In most linguistic work, the simpler a rule is, the more general it is. This is how generalizations are expressed. However, for language learning, assuming

NND, this is catastrophic. We want the child to postulate the simplest rule that is consistent with the data. But if that is also a very general rule, then she may be able to generate ungrammatical sentences and without negative data, has no way to recover from this mistake. Consider a child learning about the structure of NPs in English and suppose for the moment that universal principles permit N to be the sister of any phrasal category, under N-bar. Without the NUGP, the child might postulate a rule which allows N to be followed by any phrasal category, in the face of evidence that N can be a sister to PP. This rule is consistent with her data, and there is nothing wrong with it in terms of UG. But it is not a rule of English and a child that learned it would never learn the appropriate rule. The NUGP is a necessary assumption to allow language learning to proceed. The rule which allows any phrasal category to follow an N will generate structures that the data do not motivate, and there is a simpler rule that is compatible with the data, so the more complex rule is disallowed, based on the data.¹

The simplicity metric and NND, F&C argue, conspire to make only GB type (principles and parameters)

¹F&C note that something like NUGP must be assumed by all theories that accept NND. Otherwise, there can be no assurance that the appropriate grammars will be learned.

grammars and context free grammars learnable. The requirements of FT then force the rejection of GB type grammars, since they rely heavily on substantive principles. Therefore F&C are left with only context free grammars to consider as candidates for FT. But not just any of these will do; these context free grammars must be of the appropriate type. Because of the simplicity metric, it must be the case that simpler rules do not generate more constructions than more complex rules; simpler rules must generate the same or less constructions. This negative simplicity/generalizability correlation ensures that larger languages have more complex grammars.

For context free grammars, local trees and rules are more or less equivalent representations of the same information, assuming that the rules are fully specified and not schematic (we will see below how schematic rules change this). Which form one chooses to use at any given time depends on the point at hand and space considerations. Given this equivalence, a large language (that is, a language with many distinct local trees) will necessarily mean more (and more complex) context free rules. This view greatly simplifies the learning process. No hypothesizing is needed to learn the

grammar. Once a structure has been parsed,² the appropriate rule can be read off it. The introduction of feature notation for syntactic categories and the principles of feature instantiation, which give us schematic rules, change this picture considerably. We want to use feature notation in collapsing rules to capture generalizations and to simplify the grammar. However, then there is no longer a simple one-to-one correspondence between rules and trees. Now one rule can generate many trees.

This leads to problems if we accept the standard interpretation of a feature's being unspecified in a rule. Normally, if a feature is not specified in a rule it means that any value for that feature is acceptable. For example, the rule in (1) in standard GPSG is understood to represent the rules in (2).

1) S --> H, NP

²How a new structure is parsed is a question that F&C point out, but do not discuss. This is, however, a problem for all theories. The child is learning about her grammar based on positive evidence, which she must somehow parse. Yet without a grammar, how is the parser to work? The answer probably has to do with the parser/grammar distinction, but a such a discussion is beyond the scope of this work. See Fodor (1989) and Valian (1989) for discussion.

- 2) a. S --> H , NP
 [VFORM FIN]
 b. S --> H , NP
 [VFORM BSE]
 c. S --> H , NP
 [VFORM INF]
 d. S --> H , NP
 [VFORM PRP]
 e. S --> H , NP
 [VFORM PSP]

However, this interpretation entails the wrong simplicity/generalizability relationship. (1) is simpler (and more general) than any rule in (2), therefore none of the more specific rules in (2) would be learnable.

F&C propose a reinterpretation of this feature instantiation convention. They suggest that the lack of specification of a feature in a tree is to mean that only the default value for that feature is acceptable in that tree. They also then propose a principle, which they call 'Specific Defaults', which says that all features in all trees must have a universally assigned default value, unless they are universally free (like the AGR features in Chapter Four) or universally fixed (eg., by an FCR). F&C also point out that even though we cannot now collapse rules by omitting features, we can still collapse rules using parentheses, angle brackets, curly braces, etc. These notational devices share the property that they are all explicitly disjunctive. None of them are as economical as simply omitting the feature in question from the rule and allowing free instantiation of

either value. But free instantiation is compatible with learnability only where the free choice of these values is *universally* acceptable. Where it is not, it must be cheaper to fix a specific value than to provide a choice. Disjunctive notational devices have exactly this property; to include them in a rule makes that rule more complex. Thus, the appropriate simplicity/generalizability profile can be assured in rule collapsing.

It follows, then, from NND, in an optimal system, that there can be no 'free' instantiation of language specific (non-universal default) feature values. That is, all language specific information must be encoded into the rules of the grammar or the lexicon for that language, and all feature instantiation is universal. The reason for this is quite simple. Consider, for example, the difference between languages that have rules passing SLASH into embedded clauses and languages that do not. That is, compare languages like English that have long distance extraction and languages like Polish which do not.³ Because of NND, a child learning Polish could

³I am simplifying here, following F&C, and assuming that Polish does not have extraction out of embedded clauses. This is apparently true for tensed clauses, and for most matrix verbs though not for all (i.e., some verbs are 'bridges' for extraction), but these details do not affect the point being made here. See Fodor (to appear a) for discussion of bridges in this framework.

not learn that her language does not pass SLASHes into embedded clauses ; the lack of SLASH passing might just be an accidental gap in her data base. A child learning English, however, could learn that her language does pass SLASHes, given that there is positive evidence of it in her data sample. Once this is learned, then this fact must be recorded in one of the parochial components of her grammar (that is, either in the lexicon or the ID rules).

Thus there is effectively no 'free instantiation' of features; non-default values must be licenced by an ID rule. Free instantiation can only occur with those features that are universally 'free'. For any feature that can vary from language to language, a child can use the non-default value of that feature in a tree only after she has learned that her language allows it. But then, once she has learned it, it must be recorded in her grammar. As examples of features that seem to be universally 'free', consider the AGR features on nouns. The distribution of nouns does not seem to be restricted at all by their values for PERSON, NUMBER or GENDER. (Of course, there may be restrictions in that other elements in the tree may have to agree with the noun.)

Of course, if we insist that the child postulate two rules, as in (3), to account for the fact that VPs

sometimes have SLASHes and sometimes do not, then we are missing an obvious generalization.

- 3) a. VP --> H[2], NP
- b. VP/NP --> H[2], NP/NP

To the extent that we find correlations like this which do not occur in all languages, we must do rule collapsing in order to capture the generalizations. That is, parochial generalizations cannot follow from universal principles; universal principles follow from innate mental resources available to a grammar and hence are inviolable. Such generalizations cannot follow from constraints; we have seen that parochial constraints are not available to FT. The only source left to encode the relevant generalizations are the rule system and the lexicon, and rule collapsing is the way to capture generalizations about rules. We saw above, with regard to the omission of features from trees, that rule collapsing must be done only with the appropriate conventions, appropriately interpreted, so that greater coverage carries some cost in symbols. And, we will see below, that rule collapsing can still be dangerous as far as the language learner is concerned.

Another consequence of NND is that all relational dependencies must be universal. For example, consider feature percolation, that is, where a feature on the mother and a feature on a daughter systematically co-

vary. This co-variation must be universally specified or it must not occur in any language; a child could not learn it as a language specific (parochial) dependency. This dependency is not learnable because it requires that the child notice that a feature specification [F] never fails to co-occur with another feature specification [G], in a local tree. That is, it requires negative evidence to learn this kind of dependency, and the NND precludes that.⁴ The co-variation between [F] and [G] could, however, be the result of a universal default. As a universal it would be innate (and therefore not learned) but as a default it need not be absolutely universal. Thus, languages might or might not obey this universal, but a child whose language does not obey the universal must complicate her grammar by recording that fact.

5.2 Marking Rules

Given that we must encode in the rules of a grammar those feature values that do not follow from

⁴The 'uniqueness principle' of (Wexler & Culicover, 1980, and Roeper, 1981) is a form of negative evidence that might be allowable. It seems to work best in the lexicon (which is a place for parochial learning) and says effectively, that there is only one way to express each grammatical form. For instance, once a child learns the irregular past tense form for the verb drink, she will not mistakenly use drinked because that 'slot' is filled with drank. However, even in the lexicon, this attempt to allow the use of indirect negative evidence is not clearly well-motivated. Therefore, it seems best to avoid it, if possible.

universal principles (whether UFI, FCRs or FSDs) the question then arises, where in the rule should it be marked? Consider again (3b). As Fodor (to appear b) argues, we must encode in this rule that it is possible for SLASH to be on VP and (at least) one of its daughters. But to encode both of them is redundant; given one, we should be able to predict the occurrence of the other. Consider now the paths that features take through trees, ignoring for now the non-percolating features. It is quite unusual for a feature to be introduced into a tree, and then to disappear part way down, in the middle of the tree. HEAD and FOOT features are almost always lexically realized, in some fashion. Let us call this the 'Lexical Realization Principle' (LRP) and define it as in (4).

- 4) All features on all paths must be realized (or must terminate)⁵ at the lexical level.

What this says, intuitively, is that any feature that occurs at any given point in a tree will continue passing down through that tree until it reaches some lexical item.

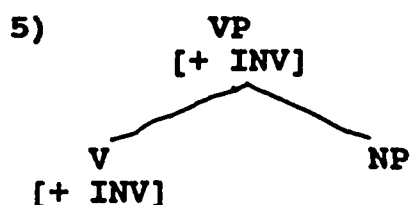
⁵Which version of this we choose seems to depend on our view of morphology. 'Realized' seems, to me, to imply it has a morphology, and then English must have a lot of zero-morphology, since it is not the case that all these features will have overt morphological/lexical reflexes (as was the case with many of the AGR feature specifications discussed in Chapter Four, for example). 'Terminated', on the other hand, seems to only imply that it ends, which it must.

Compare this to the paths that features take when we consider them as running upwards from daughter to mother. It is not the case that features further down in a tree are necessarily represented higher in the tree. This can be seen with features like SLASH (which are explicitly introduced by top-linking rules, not necessarily at the highest level of a tree) and any other (HEAD) feature that is on a non-head daughter (these feature specifications will start on that non-head daughter and will not be represented on its mother). Take a feature specification like [VFORM BSE] on a bare embedded VP. This specification will start in the tree at the level that the embedded VP is introduced, and will not 'pass' further up the tree. Features only go up trees to the point at which they are licensed (either by top-linking rules for SLASH, or until they reach the highest non-head daughter dominating them).

If features were required to pass up the tree from daughters to mothers, then one would expect that the highest S node in a tree would be marked for all of the features that occur below it. This is clearly not the case and is in general impossible since many incompatible features specifications may occur *in* the S but *on* the S. Therefore, given a that a (percolating) feature [F] is on a mother node, we can predict that it will also be on a

daughter node, by the LRP, but given that [F] is on the daughter node, it is not at all certain that it will also be on the mother node. When writing rules, we will make use of the LRP. A feature that is predictable will be encoded in a rule on the mother node and then some principle of UFI will then cause it to also be on a daughter node in any tree projected from that rule.⁶

To see how this would work, consider a child who is confronted with the tree in (5).



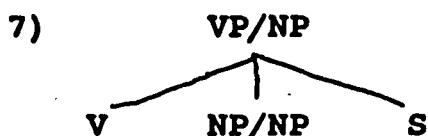
Given the PREHFC, one of the specifications for [INV] is redundant and because of the LRP, we are assuming that it is the mother which will retain information about the feature specifications to which the principles of UFI will be responsive. Therefore, the child will remove the

⁶This also allows some savings in the analysis of parasitic gaps. If we mark the SLASH on the mother, it may then 'license' SLASH on more than one of its daughters. However, if we chose to mark SLASH on the daughter(s), then we would have to mark all of them that are possible and they would go up to only one SLASH on the mother. The empirical issue here is whether it is always predictable *which* daughter a feature can pass to. See Fodor (to appear b) for discussion. It may be that features should be marked on mothers in non-lexical rules and on daughters in lexical rules. This would, however, necessitate some revisions in the version of UFI that is being developed here.

[+ INV] specification from the V node, and will form a rule roughly like that in (6).

6) VP[+ INV] --> V, NP

F&C discuss how a child would learn that her language has SLASH passing rules (see Fodor, to appear b, for additional discussion). Assume that the child is presented with a local tree like that in (7), occurring in the root clause of a question.⁷

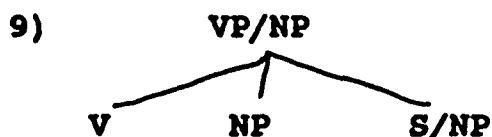


The proposal that F&C put forth is that the child knows, by the FFP, that one of these SLASHes is redundant. By the LRP (but see footnote 6) we can assume that it is the SLASH on the daughter. When she drops the [SLASH NP] off the daughter NP, she will derive the rule in (8).

8) VP/NP --> V, NP, S

But this rule, which was learned on the basis of direct object extraction in a matrix clause, predicts that NP is also extractable from an embedded S, since this rule will project not only the tree in (7), but also the tree in (9).

⁷In this chapter, as before, I will be ignoring many features in trees when they do not bear on the point at hand. This is simply for expository convenience.



Although F&C point out that this kind of predictive generalizing is in general dangerous for an FT learner, it will cause no trouble in this case. The child learning English will make use of both trees if she has a further rule expanding S/NP and knows an appropriate 'bridge verb' (a verb subcategorized for S/NP). However, a child learning Polish will never have reason to postulate a verb that is subcategorized for S/NP (that is, she will not learn any 'bridge' verbs), assuming that lexical learning is also conservative (see Fodor, to appear a). (But she will have rules to expand S/NP, in order to allow extraction in matrix clauses.) Therefore, a tree like that in (9) will never be terminated and will not cause her grammar to overgenerate.⁸ The LRP, then, is useful in guiding the process of removing features from fully specified trees to derive schematic rules.

⁸There is another way out of this potential overgeneration. GPSG does not have an S-bar level distinct from S (S-bar is simply S marked for the feature COMP). If it did, then we could simply assume that the child learning Polish would never learn rules for expanding S-bar/NP even though she will have rules for expanding S/NP. (Of course, this requires the additional assumption that embedded clauses are always dominated by S-bar. This is not an uncontroversial assumption.) Whether there are other cases that suggest a need for an independent S-bar level is still unclear.

F&C also hypothesize that the child uses the principles of FI in acquisition as devices to remove predictable features from trees and thereby determine the appropriate rule. Below we will explore this possibility with regard to the version of GPSG proposed here.

5.3 Back to GKPS

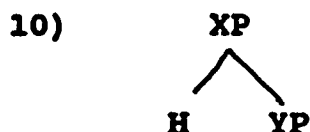
Given the modifications proposed in Chapters Three and Four, the question then is, are GPSGs learnable, under the assumptions of F&C? The answer is a resounding 'No'. In this section, I will examine the changes that must be made in the theory proposed in Chapters Three and Four, in order to account for learnability.

5.3.1 Feature Instantiation

F&C's proposals make very little difference to AGREE (the remnant of the CAP), the ADP, or the PREHFC, but the change that they propose drastically alters the FFP. FT requires that there be no 'free instantiation' and that the HFC and the FFP are effectively copying procedures. That is, they will copy inherited features from the mother to the relevant daughter(s). Chapter Four argued that this was independently useful for the HFC and that, in effect, this reduces the HFC to the PREHFC. Accepting F&C's proposals also makes the FFP very similar in content to the PREHFC.

In fact, we now have three principles of FI that are almost identical: the PREHFC copies HEAD features onto the head daughter(s); the FFP copies FOOT features onto some daughter(s); and the ADP copies the AGR features onto all daughters. It then seems simpler to say that this is just one principle which says to copy features to the relevant daughter(s), where the 'relevant daughter(s)' is (are) determined by the kind of feature being copied. (See Fodor, to appear b, for a similar proposal regarding the HFC and the FFP.) If this were the case, then one would expect there to be no ordering between these principles and that seems to be the case in fact.

It is fairly clear why this should be so. Once the CAP has been reduced in function to AGREE and the ADP is accepted, it is in general the case that the different principles of UFI operate on distinct sets of features. For example, AGREE copies the AGR features across and the PREHFC copies HEAD features down. Logically, there could be an interaction, but because they affect different features, there never is. AGREE and the ADP are the only two principles that act on the same set of features, but it is never the case that they both operate in the same local tree. Consider (10).



If the head (H) is V and YP is NP, then AGREE can operate to appropriately fill in the values for OBJAGR (assuming a language that has morphology to express OBJAGR). If the OBJAGR values had to be expressed on XP (= VP) also, then these two principles would need to be ordered with respect to one another.

However, in Chapter Three, it was pointed out that the distribution of VPs is not determined by the AGR features of its object and that VPs should not bear the OBJAGR feature. That is exactly the result of the proposal here; because the ADP copies *down* there is no way for the OBJAGR feature to be marked on the mother. Thus, the ADP and AGREE do not interact in (10) (when it is instantiated as VP, V and NP). The only other possible instantiation of (10) where AGREE and the ADP might interact is when XP is S, H is VP, and YP is NP (the linear order of the daughters is unimportant to this point). Again, AGREE will cause the VP to take on its sister's values for the AGR features and, as has been noted several times before, the mother (S) should not take on those values. S will not have those features since the ADP copies down. In this local tree also, then, the ADP and AGREE do not interact.

The PREHFC and the FFP overlap slightly in that they both affect the distribution of SLASH because SLASH is a HEAD and a FOOT feature. It is the only feature (to date) that has been advanced as being both. If possible, SLASH must pass to the head daughter, by the PREHFC. If that is not possible, then SLASH must pass to some other daughter, by the FFP. We see here both that the PREHFC is a default in that it can be overridden and that it, in some sense, takes precedence over the FFP. These two facts appear to be contradictory, but that can be resolved. If the PREHFC copies SLASH down onto the head, then the FFP is also satisfied. The FFP requires that SLASH be on at least one daughter and the head is a daughter, so there is no conflict. If the PREHFC cannot copy SLASH onto the head, then the FFP steps in and copies it onto some other daughter. This looks very much like the 'elsewhere condition' (which states that more specific rules take precedence over less specific rules) that has been proposed for phonology. Thus, if we assume that the application of FI is guided by an Elsewhere Condition (as suggested by Fodor, to appear b), then we do not need an explicit ordering statement.

5.3.2 The FCRs

Consider now the FCRs, which are inherently negative; they state restrictions on what features may

co-occur on a category. FCRs define what is a possible category; universal FCRs define possible categories in human languages and parochial FCRs define possible categories in a particular language. In the terms of F&C, both kinds of FCRs are 'constraints'. GKPS postulate both universal and parochial FCRs. (The violation of a universal FCR is 'unthinkable' in the terms of F&C.) If an FCR is universal and may be posited as a part of the innate knowledge that all children have, then all is well and good; it need not be learned and does not require negative data. However, many of the FCRs proposed by GKPS are obviously not universal but are parochial and would require negative evidence to learn since they are constraints. In order to learn an FCR a child must somehow be told that two particular feature specifications either cannot co-occur or must co-occur (cannot not co-occur) on a category. Yet, we are assuming that children do not have negative data like this available to them (at least, not systematically).

Therefore, FCRs are unlearnable, given what are reasonable assumptions about how a grammar may be acquired. For example, consider (11) (= GKPS' FCR 21).

11) $A' \supset \sim[WH]$

This records the fact that in English, WH AP's can only have the licensing WH-word in the determiner position. That is, AP cannot pied-pipe with their WH objects. It

seems that (11) should be parochial. It is unclear whether some languages do allow a WH-word as the object of an A, or have WH-adjectives (see Ultan (1978)). I will assume therefore that (11) is parochial because parochial constraints will in general be harder to describe than universal ones (see discussion in F&C). The FCR in (11) blocks sentences like (12a) while allowing sentences like (12b).

- 12) a. *Fond of whom are you?
b. How fond of them are you?

(12a) is ruled out because for the WH feature (a FOOT feature) to occur on the AP it must 'pass' up from the PP daughter up to its A-bar mother and on up to the AP. But it cannot because of (11). Thus, this AP cannot be 'extracted'.⁹ (12b) is allowed because the WH-word is in the determiner slot and then the WH feature can be on the daughter and its AP mother without going via A-bar.

The child's grammar would be formally simpler without the FCR (according to the simplicity metric) and

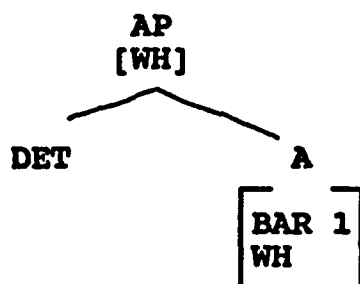
⁹It is unclear exactly how to deal with WH *in situ*, as in echo questions or multiple WH-questions. Lee (1985) argues against the FCR in (11) using data from multiple questions in English and Korean. His analysis, however, does not explain the difference between (12a) and (i).

(i) Mary is fond of whom?
Syntactically, WH-words *in situ* seem to behave as if they do not in fact have the feature WH (see Fodor to appear b; Maxfield 1990, for some discussion). We will see below another case where we want to say there is an occurrence of WH, but which does not act like regular WH.

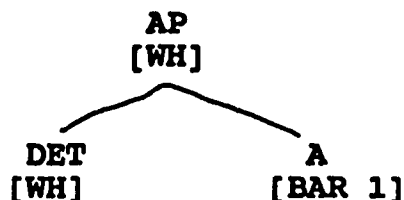
would cover all the data the child has been exposed to. That is, a grammar without the FCR in (11) would generate all of the grammatical English sentences with APs, as well as the ungrammatical ones as in (12a). Since we are assuming that the child does not know that (12a) is ungrammatical, there would be no motivation for her to postulate such an FCR as it would complicate her grammar unnecessarily. Thus we need another analysis of this restriction, one which does not posit a parochial FCR. The general problem is that GKPS use FCRs to clean up the overgeneration caused by highly schematic rules and liberal feature instantiation. We will have to restrict the rules in order to remove these constraints from the grammar, and this unfortunately is almost certain to mean complicating the rules. In this section, I will look at the FCRs that GKPS propose, and see what will be required to eliminate the FCRs that are (as far as we can tell) parochial.

Given that (11) is language particular, a child learning English will have to learn that APs do not pied-pipe, but she could not learn this FCR. The situation is similar to that of SLASH above. Consider first a child who is learning a language just like English, except that it also allows a WH words to be the object of an adjective. This child will be exposed to the trees in (13) which she will then reduce to the rule in (14).

13) a.



b.



14) AP[WH] --> DET, A[BAR 1]

(14) is the appropriate rule as the FFP will then force WH down onto either daughter (or both), which is fine in this language.

Now, consider a child learning English. She will be exposed only to a tree like that in (13b) and never a tree like (13a). She will be encouraged by the simplicity metric to reduce this tree to the rule in (14) (since the FFP makes the [WH] on the daughter redundant). But, this rule predicts that the WH can be on either the DET or the A-bar, which is wrong for English. However, even though the child's grammar will allow the WH to pass to the A-bar daughter, she will never be exposed to a tree with WH on the A-bar, so she will have no rule(s) expanding that node and the tree can never terminate. Again, this predictive generalization has no bad

effect.¹⁰ A child learning a language that allows WH-words as objects of adjectives, or has WH-adjectives, will simply learn a rule like (14) and other rules to expand A[BAR 1, WH].

GKPS' FCR 22, given in (15), is very similar to (11).

15) VP $\supset$ ~[WH]

As with (11) above, this FCR records the fact that the feature WH cannot pass through a VP node (therefore VP cannot pied-pipe if one of its complements contains a WH-word).¹¹ The fact that VPs do not pied-pipe is probably related to the fact the S also does not pied-pipe, as shown in (16).

- 16) a. Fred believed the claim that Sally saw Ted.
b. *The claim that who saw Ted did Fred believe?

In (16b), the WH feature should be able to pass up from the embedded subject to the S node and on to the containing NP, but it cannot. We can tell, however, that S must sometimes bear the WH feature because verbs like

¹⁰Note that this generalization could overgenerate, depending on the other constructions in the language. This situation is distinct from that with Polish, above, because English never has WH passing through A-bar, where Polish does have at least one instance of SLASH passing through S (that is, in matrix clause extraction).

¹¹It is interesting to note that it seems to be the case that those languages that have WH adjectives and WH verbs are also languages that do not allow 'extraction' (cf. Chinese). Is there a necessary connection here, or is it just coincidental?

wonder are subcategorized for WH complements. The FCR in (15) obscures the connection between the lack of pied-piping in both VP and S. One possible resolution of this is simply to say that WH on S and VP is like WH *in situ*; that is, not a FOOT feature at all. Then it would not be surprising that these instances of WH cannot be involved in pied-piping. However this is resolved, it seems clear that this FCR is ill-motivated and not only can, but should be dispensed with.

GKPS capture WH-island effects with the FCR in (17).

17) ~([SLASH] & [WH])

They do not say whether they consider this to be universal or parochial. Given the arguments in Rizzi (1982), it seems that it has to be parochial. That is, it appears that Italian, at least, allows extraction out of what should be WH-islands. In (18) (taken from Rizzi, 1982), the WH-phrase 'a cui' has been extracted from a WH-clause.

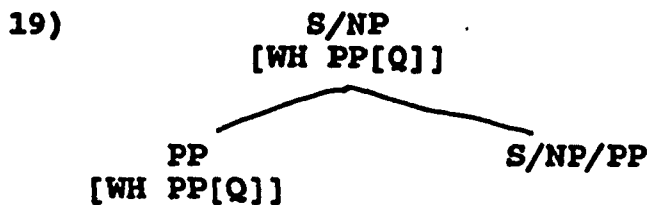
18) Tuo fratello, a cui mi domando [s che storie
abbiano raccontato, era molto preoccupato.
Your brother, to whom I wonder [s which
stories they told __ __, was very troubled.

Rizzi analyzes this in terms of the parameterization of subjacency. What is important to us here is that the

lowest S must be specified as both [WH] and [SLASH]. Since there is variation across languages as to whether they obey (17), a child must learn whether her language obeys this constraint. Because constraints are unlearnable, (17) must be universally specified and the child learning Italian must learn the rules to override it. I will assume, therefore, that Universal Grammar includes a statement identical to (17), only it is an FSD, not an FCR. By changing GKPS' FCR into an FSD, we are also able to change it from a parochial statement into a universal statement. This predicts that languages will tend to observe WH-islands, but that they can extract from them at a cost. (That is, the WH-island constraint is a universal trend, not an absolute universal.¹²)

For the child learning Italian, then, the occurrence of structures like (18) is a fact that she must include in her grammar. I assume, following Rizzi, that the tree for (18) has a subtree with a structure like that in (19).

¹²Note that in general universal trends will be harder for a theory to accommodate than absolute universals. Absolute universals can always be encoded in UG. But universal trends force a theory to include some explicit notion of markedness. See F&C for a fuller discussion.



Thus the child learning Italian will have reason to postulate the rule in (20) while the child learning English will not be motivated to construct a rule like this.

20) $S[\text{SLASH NP, WH PP[Q]}] \rightarrow PP, S[\text{SLASH}]$

The value for SLASH on the mother need not be specified on the head daughter, as SLASH is a HEAD feature and its occurrence on the head daughter is predictable from the PREHFC. Note, in addition, that the value for the second SLASH specification need not be indicated as it will follow from the markedness convention of F&C discussed in Section 4.2.7 (i.e., it will have the same specifications as its XP sister). Also, the FSD in (17) will cause the WH specification on the mother to be realized on the PP daughter, in the unmarked case, which this is.

GKPS postulate four FCRs that all deal with COMP, restricting the distribution of the elements in it. Their FCR 15, given in (21), seems in fact to be a strong universal trend, but it cannot be an absolute universal (even assuming, as GKPS do, that only sentences can have subjects).

21) $[\text{COMP}] = [+SUBJ]$

It does not hold in the 'for-to' dialects of English, which find sentences like (22) grammatical.

22) There wasn't anything for to eat.

The accepted analyses of these structures is that for in (22) is the complementizer for, and here it appears to be in construction with an infinitival VP. GB theory posits phonologically null NP (PRO) to be the subject of these structures, but GPSG admits only one phonologically null NP, essentially WH-trace. Thus, (22) appears to violate (24). (21) expresses a strong universal trend, but not an absolute universal. Therefore, (21) cannot be an FCR, but it can be an FSD.

(21) is, however, an equivalence and must hold in both directions. Considering the equivalence in the other direction, it does seem to be the case that all categories that are [+SUBJ] are specified also for COMP. In non-root environments, it seems generally to be the case that an overt complementizer is obligatory in sentences, while in root clauses it seems much more uncommon for there to be a phonologically realized complementizer (see van Kemenade (1987) for examples of complementizers in Old English root clauses). Following GKPS, I assume that English has NIL as a possible value for COMP, which can usually alternate with that (this makes it appear that in many cases that is optional). I also assume that English is uncommon in this respect, and

that as a general course, languages do not make use of NIL in embedded clauses, except in clauses with initial WH phrases.¹³ I will henceforth assume the FSD in (23a) and the FCR in (23b), and eliminate (21).

- 23) a. [COMP] $\supset$ [+SUBJ]
 b. [+SUBJ] $\supset$ [COMP]

Chomsky and Lasnik (1977) proposed the "doubly-filled comp filter" to explain the fact that clause initial WH words and overt complementizers (e.g. for, that, if, etc.) cannot cooccur. In TG, both complementizers and clause initial WH phrases reside in the COMP node, which is dominated by S-bar, but only one can occur in any given COMP. (24) (GKPS' FCR 16) is effectively duplicates for GPSG the descriptive work done by Chomsky and Lasnik's doubly-filled COMP filter.

- 24) [WH, +SUBJ] $\supset$ [COMP NIL]

While this might be a universal trend, it is obviously not a universal; there are languages that allow multiple WH-words in initial position as well as languages that allow both a WH-word and a complementizer to be clause

¹³It is unclear just how to restrict NIL so that it cannot occur in sentential subjects and extraposed clauses, and force that to occur in them. However, this is a general descriptive problem that affects not only the analysis of GKPS, but all other theories that I am aware of. Stowell (1981) proposed an analysis of the alternation of that and NIL, based on the ECP. However, this analysis does not go through in the 'Barriers' framework and so Chomsky (1986) dispensed with it. To pursue this issue here would take me too far afield, and so I will ignore it.

initial. Again, however, we can reinterpret this parochial FCR as a universal FSD.¹⁴ This strategy of reinterpreting parochial FCRs as universal FSDs, allows us to capture universal trends but permits cross-linguistic variation. Therefore a child learning a language that allows multiply filled COMPs will have to mark that fact in her rule, as in (25), but a child learning a language like English will have no motivation to include the COMP specification on the daughter S.

25) S --> XP, S[COMP, SLASH]

GKPS' FCRs 17 and 18, given in (26) and (27), respectively, include English morphology, as noted in Section 2.4., and for that reason are clearly parochial.

26) [COMP that] > ([FIN] V [BSE])

27) [COMP for] > [INF]

However, if we eliminate the English morphology, then it is possible to maintain them as universal FCRs. The appropriate FCRs are given in (28) and (29).

28) [COMP FINT] > ([FIN] V [BSE])

¹⁴The exact form of this FSD may be debatable. The form it takes in the text is based on GKPS' analysis of COMP as a feature on S (as noted above in footnote 8), as opposed to the GB-type analysis where COMP is a separate node, dominated by S-bar. If this kind of analysis is determined to be superior, then (24) could be changed, as in (i) which comes closer to capturing the notion that the WH-word is *in* COMP, or rather replacing COMP.

(i) [WH] > ~[COMP]

29) [COMP INFI] $\supset$ [INF]

We may note that it seems to be less common for languages to have an infinitival complementizer. English does, so it has infinitival sentences (that is, what GPSG regards as infinitival sentences, viz. sentences with a phonetically realized NP preceding an infinitival VP), as in (30).

30) For John to leave would surprise Bill.

Many languages (eg., the Romance languages, except Portuguese), however, have no infinitival complementizer and sentences comparable to (30) are ungrammatical.

As far as I can tell, FCR 14, given in (31), is motivated totally by GKPS' analysis of conjunction.

31) [+PRD, VFORM] $\supset$ ([PAS] V [PRP])

Following Sag, et al. (1985), GKPS allow (almost) any category that is [+PRD] to be conjoined after a form of be. The ID rule which introduces this predicative construction is given in (32).

32) VP[+AUX] $\rightarrow$ H[7], XP[+PRD]

However, it appears that present participles and passive participles are the only verb forms that can enter into this construction freely, although all verbs are predicative. That is what (32) enforces. Jacobson (1987) argues against this analysis for several reasons. Firstly, it gives up the early GPSG account of coordination where only categories which are alike in

major category features can conjoin (see Gazdar, 1981). This account was both simple and elegant. Secondly, this account relies on the assumption that SLASH is a HEAD feature, and we have already seen (in Chapter Four) that Jacobson argues against that, even though taking SLASH to be a HEAD feature simulates for GPSG much of GB's subjacency (which is a persuasive argument to keep SLASH as a HEAD feature). Finally, she points out that the GKPS analysis of coordination along with their formulation of the HFC predicts that (33) should be grammatical.

33) *A man and four women writes and sell books. This analysis allows conjuncts to have distinct values for any HEAD features (specifically, AGR on VP and PER and PLU on NP) as long as their mother can be unspecified for those features and yet remain a legitimate category. However, (33) is glaringly bad. Jacobson suggests that perhaps instead there is an actual category, "Pred", which is what be subcategorizes for, and is conjoined in examples like (34), and that Pred can be rewritten as certain other categories.

- 34) a. Mary is a linguist and proud of it.
 b. They walked slowly and with great care.

I will assume that an analysis like this can be worked out, and that (31) is therefore unnecessary to the grammar.

GKPS's FCRs 12 and 13 are both eliminated here. FCR 12, given in (35), is simply not true on the analysis of agreement proposed in Chapter Four.

35) [AGR] $\supset$ [+V]

(36), their FCR 13, is also not needed in the syntax, assuming that nominative Case assignment is handled by subcategorization because subjects are subcategorized for (see section 4.2.2).

36) [FIN, AGR NP] $\supset$ [AGR NP[NOM]]

This FCR could well state a universal lexical redundancy rule, but it is of no use to the syntax proper.

GKPS motivate their FCR 10, given in (37), saying that the FCR will stop [+INV] (a HEAD feature) "from ever dripping through VP (which, by definition, is [- SUBJ])." p. 61.

37) [+INV, BAR 2] $\supset$ [+SUBJ]

However, this 'dripping' of INV is not possible. Their FSD 1, given in (38) would cause [+INV] to occur only where it is inherited, due to the SAI metarule, and on the head of that construction, which must be a lexical verb, given the restriction on the operation of metarules.

38) [-INV]

Even though VP is usually the head of S, in rules that are the output of the SAI metarule, V is the head of S

and *not* VP. Thus, there is no way for a (non-head) VP to be instantiated with [+INV], a HEAD feature and (38) is not needed. On the analysis of SAI and the operation of FSDs proposed here, (37) is also unnecessary. Therefore it will be eliminated from the grammar.

GKPS also propose the parochial FCR in (39) to help deal with the output of the SAI metarule.

39) [+INV] $\supset$ [+AUX, FIN]

This FCR is required because GKPS want the metarule to be very simple and therefore usable by the grammars of several languages (cf. GKPS, p. 61). The metarule is defined for verbs in general. Inversion is limited to Auxiliary verbs in (modern) English by the FCR in (39). However, a language learner could not acquire this constraint on English, as we have already seen. The problem here is GKPS' assumption that metarules preserve subcategorization (i.e., that every lexical item subcategorized for the ID rule input to a metarule is also subcategorized for the corresponding ID rule output). However, we are assuming that items must be independently subcategorized for the derived rule. And it is safe to assume that subcategorization must be acquired by learners independently for each item in the lexicon. (See Fodor, 1983 for discussion.) The invertible verbs in English thus do not need to bear the

feature [+AUX]; they will be just those verbs subcategorized for the relevant ID rule (eg. for a nominative NP sister and a nonfinite VP sister). In other languages, a larger class of verbs will be so subcategorized. On these assumptions, positive data will suffice for learning and (39) is unnecessary.

Assuming that the analysis of SAI as 'verb topicalization' proposed above is tenable, the FCRs in (37) and (39) are as unnecessary as they are on the GKPS analysis. The specification [+INV] will occur only where it is licensed by an ID rule, given the FSD in (38). (39), which serves GKPS by restricting the overgeneration of their SAI metarule, is not needed on this analysis because there is no SAI metarule to overgenerate. Rather, there is simply another lexical ID rule that certain verbs will be subcategorized for, but most will not be.

FCRs 7, 8, and 9 (given in (40)) restrict the distribution of the SUBCAT feature.

- 40) a. [BAR 0] = [N] & [V] & [SUBCAT]
 b. [BAR 1] $\supset$ ~[SUBCAT]
 c. [BAR 2] $\supset$ ~[SUBCAT]

However, since I am using a very different (and more standard) notion of subcategorization, I do not need these restrictions. Assuming that subcategorization information is just a part of lexical representations, and not a feature at all, there is no more reason to

assume that that information will percolate through a tree than there is to assume that phonological information, for example, will percolate through a tree. Therefore, the FCRs in (40) are unnecessary and we can eliminate them. (41), GKPS' FCR 6, stops SLASH from passing from a mother to a lexical daughter.

41) [SUBCAT] $\supset$ $\sim$ [SLASH]

Jacobson (1987) says that this is based on GKPS' misanalysis of SLASH as a HEAD feature (which, as we saw in Chapter Four, does a lot of descriptive work for GPSG). In fact, she claims that it could only be independently motivated based on the assumptions one makes about the lexicon. GKPS assume a schematic lexical entry for the empty string that allows it to be inserted under the node α [+NULL]/ α .

Jacobson points out that this lexical item will differ from other lexical items in two different ways: it must be specified as NULL; and it must have the schematic form α / α . Jacobson argues that it is possible to do without at least one, if not both, of these stipulations. However, it seems to me, that the lack of SLASH on a lexical category is a logical necessity, assuming, as GKPS do, that SLASH takes only phrasal values. That is, as long as a lexical item cannot contain a phrase, it also could not contain a phrasal gap. Assuming that SAI is in fact verb topicalization,

then SLASH must be allowed to take non-phrasal values and it no longer follows that lexical categories cannot bear SLASH (in fact, they must in order to permit the category V/V). We must then reformulate (41) as (42), which says explicitly that a BAR 0 category cannot bear a SLASH for a category of BAR 1 or greater.

42) [BAR 0] $\supset$ $\sim$ [SLASH [BAR > 0]]

The FCR in (43) (= GKPS' FCR 19) also helps restrict the distribution of SLASH.

43) [+NULL] $\supset$ [SLASH]

This FCR must be worded this way because their SLASH Termination Metarule 1 (STM1) introduces the feature NULL and not the feature SLASH. That is necessary in turn because, for GKPS, the FFP must distinguish between inherited and instantiated SLASH. If STM1 were to produce ID rules that introduce SLASH on a daughter node, then that SLASH would be inherited into the trees projected from the rules resulting from STM1 and could not then be instantiated on the mothers. That is, the trees would have the shape of top-linking trees and not bottom-linking trees. But, STM1 puts NULL into the tree and then the FCR in (43) requires that SLASH be instantiated on the daughter and the FFP requires that SLASH be instantiated identically on the mother.

Once we alter the theory, following F&C, such that parochial information is explicitly encoded in the

ID rules, the FFP becomes a copying procedure which applies to both inherited and instantiated FOOT features (see section 5.3.1, F&C, and Fodor to appear b for discussion). Thus we no longer need STM1 in order to force SLASH to terminate: SLASH can terminate just in case it is on a node labelled α/α . And therefore we also no longer need (43). This move also eliminates several of Jacobson's criticisms of GKPS: we have uncomplicated SLASH termination; we need not specify [+NULL] on any lexical item; and we do not need a lexical item with a schematic form.

F&C propose the FSD in (44) which is very similar in form to GKPS' FCR in (43) and which captures the connection between +NULL and SLASH, but it is quite different from (43).

44) XP[SLASH XP] $\supset$ [+NULL]

F&C stress two points about this FSD as opposed to the FCR in (43).¹⁵ First, it captures a generalization that is not an absolute universal; namely the A-over-A constraint. (44) says that in the default case, XP[SLASH XP] will be realized as a trace, but it allows for marked

¹⁵F&C actually posit the FCR in (i).

(i) [+NULL] / XP[SLASH XP]
This format makes the context-sensitivity of this FCR explicit. Janet Fodor (p.c.) has pointed out, however, that (44) has the same effect and does not require the introduction of new notation. That is, GKPS also used context-sensitive FCRs and FSDs.

exceptions to that. Second, it is a context sensitive FSD. F&C argue that [+] is surely the default value for NULL in this context just as much as [-] is the default elsewhere. The use of context-sensitive FSDs allows a richer system of defaults that can then do more descriptive work.

GKPS propose two FCRs whose universality is not easy to judge. They reflect particular decisions that GKPS make about the proper description of certain natural language facts. The first is their FCR 11, given in (45), which simply says that V^2 is the only category that can have a subject.

45) [+SUBJ] $\supset$ [+V, -N, BAR 2]

It is possible that one would also want to analyze NPs as having subjects (as in John's criticism). I will assume that GKPS are correct, but if it turns out that they are wrong, the new analysis could easily be integrated into this theory. The other FCR whose status is questionable is given in (46), their FCR 5.

46) [PAST] $\supset$ [FIN, -SUBJ]

This FCR expresses the definitional point that any verb marked as + or - PAST is finite, and simultaneously stops the tense feature from passing up to the S node. The latter point is justified by the fact that verbs do not subcategorize for sentential complements of a particular tense (though choice of COMP will depend on whether they

are tensed or not). Again, the FCR is acceptable (that is, compatible with learnability) only if it is universal. I will accept it for now, but if further descriptive work reveals that this is a mistake, it can be corrected easily.

The FCRs that I am assuming are taken to reflect universal facts about the co-occurrence restrictions on feature values in categories. There are, however, three facts universal co-occurrence restrictions which seem to be ignored in most discussions. They are: root clauses (in the sense of Emonds, 1976) must always bear tense; the form of a verb does not affect its complement structure; and the choice of + or - PAST is always free.¹⁶ The FCR in (47) reflects the first of these facts.

47) a. [VFORM, +ROOT] $\supset$ [VFORM FIN]

(47) requires the introduction of the Boolean feature ROOT, which will be required by any theory, assuming that the insights of Emonds (1976) are correct. To account for the fact that the choices of VFORM and PAST are universally free, Universal Grammar need not say anything. That is, if we assume that the appropriate interpretation of a feature not having a value specified by an FCR or FSD is that any value for that feature is

¹⁶The "sequence of tense" phenomena is apparently a counterexample to this claim. I will ignore it here.

acceptable, then UG will contain no FCRs or FSDs that assign a value to VFORM or PAST.¹⁷ This is not to deny that certain verbs (such as require) are subcategorized for only certain complements (eg., an S with the verb in the BaSE or uninflected form). But that is a matter of subcategorization (which I am taking to be lexical), and not in the domain of the syntax proper.

All that is left now of GKPS' FCRs are those that they call 'uninterestingly universal'. That is, they are definitional; they (partially) define what it is to be a possible category in a natural language. In addition to the FCRs listed in their book, I will also be assuming several others, that are 'uninteresting' in the same way. Both sets are listed in (48); those prefixed with a '#' are new ones that I am proposing.¹⁸

¹⁷This interpretation was pointed out to me by Janet Fodor.

¹⁸Note that I have modified GKPS' FCRs 2 and 3 such that they are equivalences rather than just implications. I have also changed GKPS' FCR 11, replacing [+V, -N] with [VFORM]. Given their equivalence (or implicational relationship on GKPS' view) this does not change the content of the FCR at all.

- 48) 2. [VFORM] = [+V, -N]
3. [NFORM] = [-V, +N]
4. [PFORM] ⊃ [-V, -N]
5. [PAST] ⊃ [FIN, -SUBJ]
11. [+SUBJ] ⊃ [VFORM, BAR 2]
- #1. [COMP] ⊃ [+V, -N]
- #2. [POSS] ⊃ [-V, +N]
- #3. [LOC] ⊃ [-V, -N]
- #4. [ADV] ⊃ [+V, +N]
- #5. [+INV] ⊃ [FIN]
- #6. [NFORM] ⊃ [CASE]
- #7. [NFORM] ⊃ ([PER] & [PLU] & [GEND])
- #8. [+SUBJ] ⊃ ~[PER] & ~[PLU] & ~[GEND]
- #9. [VFORM, +ROOT] ⊃ [FIN]
- #10. [COMP FINT] ⊃ ([FIN] V [BSE])
- #11. [COMP INFI] ⊃ [INF]
- #12. [+SUBJ] ⊃ [COMP]

Of the FCRs that I am assuming, #1 through #4 are 'uninteresting' in that simply restrict the distribution of minor, special purpose features. If these features were allowed to proliferate in the tree, it would only make the trees messier and harder to read; it would do no real harm. #5 expresses my assumption that only FINite verbs can be inverted. If this turns out to be only a universal trend and not an absolute universal, this FCR can be converted into an FSD, but at this point I do not know of any evidence that suggests that this change is necessary. FCRs #6 and #7 reflect the assumption that all projections of N are marked for PERSON, NUMBER, GENDER, and CASE. If this assumption is shown to be false, then the rules that I postulate below will be slightly more complex, but the analysis will still go through. The fact that sentences should not bear the AGR features, as discussed above, is expressed in FCR #8.

Since all of the FCRs in (48) are universal, they can be innate and are thus no threat to learnability.

5.3.3 FSDs

Before we look at the specific FSDs that GKPS have proposed and the ones that I will be assuming in the rest of this work, there are some more general questions about FSDs that must be considered first. We will turn to these now, and return to the individual FSDs in section 5.3.3.2.

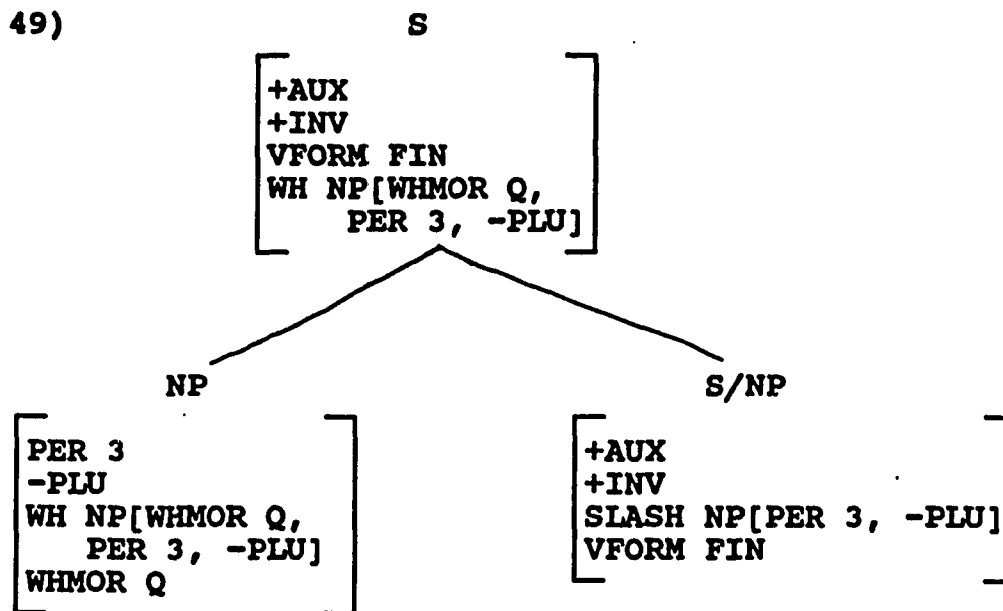
5.3.3.1 The function of FSDs. Consider again what it is to learn a GPSG: it is the inverse of the feature instantiation process. That is, we can picture a child as taking a fully specified tree and stripping off the features that are predicted by UFI, the FCR's and FSD's. (I will ignore LP statements in what follows.) Whatever is left is a schematic rule which should be added to the child's grammar if it is not already present. However, if this conception of learning is combined with the linguistic theory proposed by GKPS, it runs into several problems. One problem is that a learner can only know and apply the FCRs (and FSDs) if they are innate and hence universal. We have dealt with this by eliminating all aspects of feature instantiation that are not universal. All parochial facts must be

encoded in the rules and the lexicon of the grammar for the language in question - i.e., not stripped off.

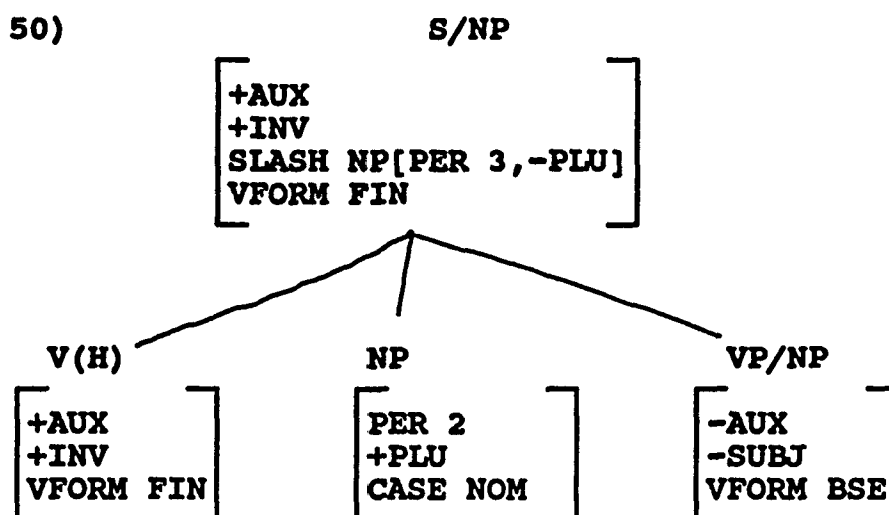
But there are other problems too. This is not surprising, since GKPS were not concerned to develop a system that is learnable, or even psychologically real in any sense. They were concerned only with devising a system that is descriptively powerful enough for natural language but also not overly powerful, and hence explanatorily adequate: this is no small task. Nevertheless it is important to find out whether there is some variant of their theory which does meet the demands of learnability. Regardless of other any changes made to the theory, unless GKPS' implementation of the FSDs is changed, there is a problem which threatens the conception of rule learning as feature stripping from local trees. On the GKPS model, in order to determine if a particular default is to apply to a category in a tree (so that feature can be stripped from that tree), we must inspect the set of candidate projections for the ID rule which licenses that tree but this means that we must know the rule, which is what we were trying to induce. This circularity is unavoidable, given the GKPS implementation of FSDs.

GKPS need to implement FSDs in the way they do in order to allow the non-default value to occur (by free instantiation) in local trees in certain cases but not in

other cases. Normally this would occur where something in some other local tree requires a particular HEAD or FOOT feature to have its non-default value, and the HFC or FFP requires that value to be 'passed up' into this local tree, so the two local trees can be joined. Consider, for example, the tree in (49).



As noted above, the default for [INV] is minus, but the S and S/NP in (49) both have the non-default value for this feature. This is necessary to allow this tree to be the topmost local tree in a structural description for a direct WH-question, which necessarily has 'Subject-AUX Inversion'. The SAI metarule introduces the feature [+INV], but not in this tree. It must be present in (49) only so that this tree can hook up with a tree like (50), which is the result of the SAI metarule, with an S/NP mother and an auxiliary verb daughter.



The GKPS definition of (non-local) admissibility was complicated for just these cases, where a phrasal daughter and mother co-vary on a HEAD feature so that information can effectively be passed through non-local trees (see the discussion in Chapter Two and GKPS, pp. 103-104). This feature (INV) then does not need to assume its default value and appears to 'violate' the requirement of the FSD's.

On GKPS' view, the FSDs act basically as filters; they must look at all of the possible projections for a schematic rule, and then exclude some of them, if certain conditions do not hold. Shieber departed from that view, but left the FSDs effectively nothing to do, except in the case of the non-propagating features. I proposed that the FSDs should apply like the FCRs; that is, to be constraints on FI. The result of this was that the rules which generate constructions with non-default values would

have to explicitly include those values and hence be more marked, in the formal sense.

This is exactly the position that F&C take, for different reasons. F&C argue that FSDs (once their parochial elements have been removed) are useful, in fact possibly necessary, for capturing universal trends (as we saw with the parochial FCRs above). For example, as discussed above, the A-over-A condition is a strong universal trend; extraction rules generally do not take an XP out of an XP. F&C capture this fact with the FSD in (44), repeated here for convenience (see footnote 15).

51) XP[SLASH XP] > [+NULL]

As a default, this will force rules that have the non-default value for NULL to include more information; they will be marked. That is, the rules necessary for extraction from 'picture noun phrases' in English must be as in (52).

52) a. VP/NP --> V, NP[-NULL]
 b. NP/NP[-NULL] --> DET, N'

Thus the grammar is more complex for having to include these two marked rules than it would be if English did not have this marked exception to the A-over-A condition.

Thus, F&C depart from GKPS in a different way than Shieber does. They argue, from learnability considerations, that the highly schematic rules of GKPS must be replaced by more specific rules. Once this is

done, the HFC and the FFP become very much like the PREHFC of Chapter Four. That is, for F&C, features can be freely instantiated on a particular category only if they are universally acceptable on that category in that position in a local tree. Therefore, all non-universal features in a language must be encoded in the rules of the grammar. The PREHFC and the FFP cannot be used to instantiate these non-universal features, they can only copy features from the mother down onto certain daughters. Consider, for example, a tree with SLASH specified on both the mother and a daughter. Neither the PREHFC nor the FFP can be used in this case to freely instantiate the SLASH, they can only be used to copy down an inherited feature from the mother to the daughter so that the rule does not have to specify it on both nodes. Thus they are being used to reduce the redundancy in the grammar.

This difference comes from the very different approach that F&C have, as opposed to GKPS or Shieber. GKPS and Shieber are both looking only to make the grammar as simple and elegant as possible. Therefore, it is perfectly acceptable to use their highly schematic rules. F&C, on the other hand, are concerned about the possibility of learning a grammar, and that requires grammars to be very different. As we saw above, learnability considerations force us to eschew constraints, therefore we must also avoid overgenerating rules. One of the results

of schematic rules tends to be overgeneration. Within GPSG, one can use principles like the HFC and the FFP to constrain that overgeneration. Once the HFC and the FFP can no longer be used in that fashion, something else is needed. Universal FSDs are the only device compatible with learnability that will serve this purpose. They will allow the child to eliminate feature specifications from her rules just in case it is the default value that's specified. Therefore, they can make predictions about universal trends as well as allow the rules that a child learns to be simpler and more elegant.

What we see then is that once rules are not as schematic as GKPS assume, the FSDs serve a useful function in the grammar. But they can only allow for universal default features to be stripped off by the learner (equivalently, they allow default values to be supplied in sentence generation); they cannot act as filters on overgenerating rules. In order to appear in a tree, non-default values must be explicitly encoded in the rules. This gives the same results as the GKPS analysis while doing without their notion of 'privilege'.

The circularity that arises if one tries to reverse GKPS' implementation of the FSDs in order to accommodate learning as feature stripping does not arise here. The problem is that on GKPS' view, one has to examine the entire set of projections from some rule in order to

determine if the defaults are to apply to some feature in a tree projected from that rule. But that entails knowing the rule, which by hypothesis, the child does not. On the view being presented here, however, there is no such problem. The FSDs are universal and always in effect, unless they are explicitly overridden; there is no need for the notion 'candidate projection'. Thus, if a child is confronted with a tree which contains a non-default value for some category, she knows that that must be recorded in her rule. A rule with no specification for some feature will only license trees with the default value and a rule which specifies the non-default value will license trees with that value.

5.3.3.2 The FSDs. For GKPS, FSDs, like FCRs, can be either universal or parochial. Also like FCRs, FSDs will pose no threat to learnability if they are universal. However, if FSDs are parochial, then they must be learned. But FSDs would require negative evidence to learn since they are constraints. Therefore, assuming NND, FSDs cannot be parochial, they must be universal.

The one FSD which GKPS list that is clearly parochial is given in (53), their FSD 9.

53) [INF, +SUBJ] $\supset$ [COMP for]

This FSD says that in (standard) English, infinitival sentences are preceded by the complementizer for. This is

specific to English partly because the phonetic form of the lexical item is more or less specified. Note that it is also almost the inverse of (27) (repeated as (54)).

54) [COMP for] $\supset$ [INF]

Once the parochial FCRs in (27) and (28) are made universal, (repeated as (55) and (56)) then there is no need for (53) (given that sentences, by definition, are specified for COMP, and for is the only infinitival COMP in English).

55) [COMP FINT] $\supset$ ([FIN] V [BSE])

56) [COMP INFI] $\supset$ [INF]

The fact that infinitival VPs seem to be more common (or less marked) than infinitival sentences can be recorded in the following FSD, which we can assume is universal.

57) V^2 [INF] $\supset$ [-SUBJ]

GKPS propose several FSDs to constrain the occurrence of minor category or special purpose features. These FSDs allow these features to occur only when they are inherited from an ID rule. Consider, for example, GKPS' FSDs 5, 6 and 7, shown in (58).

- 58) a. [PFORM] $\supset$ [BAR 0]
 b. [+ADV] $\supset$ [BAR 0]
 c. [BAR 0] $\supset$ ~[PAS]

The FSDs in (58) are just complicated ways of saying that feature specifications for [PFORM], [ADV], and [VFORM PAS] are default absent (i.e., that they only occur where licensed by an ID rule), as represented in (59).

- 59) a. ~[PFORM]
b. ~[ADV]
c. ~[VFORM PAS]

Take (58a), for example (similar comments could be made about (58b) and (58c) as well). GKPS say

. . . because of the FSD in (109) [= (58a)], a prepositional phrase will in general be unspecified for PFORM if it is licenced by a PP in an ID-rule that is itself unspecified for PFORM. This places what appears to be the proper constraint on the distribution of subcategorized-for prepositions.
(GKPS pp. 132-133)

Clearly then, GKPS intend for (58a) to express what (59a) expresses; a PP daughter in a tree will be specified for PFORM only if that specification is inherited from the ID rule, otherwise PFORM will be unspecified. GKPS cannot use the more concise and straightforward formulations in (59) because for them, lexical categories cannot have non-default values just because they co-vary with their mothers. Thus an FSD like (59a) would entail that no preposition could be marked for PFORM in any tree. The FSDs in (59) apparently achieve the appropriate results, their intended meaning is clearer than the comparable FSDs in (58), and they are of the general form of many other FSDs dealing with minor features. Therefore, I will remove (58) in favor of (59).

There are also two other FSDs that GKPS propose that I shall eliminate. One says that projections of N (categories that are specified NFORM) are default specified as [NFORM NORM] and the other says that verbs, in the

default case, agree with NPs that are [NFORM NORM]. GKPS use these two FSDs to help regulate the occurrence of expletive NPs. However, since I am using subcategorization to handle the distribution of expletives, these FSDs are not needed. That is not to say that I doubt the generalization captured by each of these FSDs. It seems more plausible, though, to regard them as generalizations about the lexicon. While this information could well be used to reduce the redundancy of lexical entries, it does not seem to be necessary to the syntax.

Before we consider how a child would learn the type of grammar being proposed, let us look at the FSD's that I will be assuming. These are listed in (60) (the # indicates an FSD not taken from GKPS).

- 60) 1. [-INV]
 2. ~[CONJ]
 3. ~[NULL]
 4. ~[NOM]
 #1. ~[ADV]
 #2. ~[PFORM]
 #3. ~[LOC]
 #4. ~[PRD]
 #5. ~[RE]
 #6. ~[WH]
 #7. ~[COMP]
 #8. ~[GER]
 #9. ~[NEG]
 #10. ~[POSS]
 #11. ~[REMOR]
 #12. ~[WHMOR]
 #13. ~[SLASH]
 #14. XP[SLASH [XP]] $\supset$ [+NULL]
 #15. ~([WH] & [SLASH])
 #16. [WH] $\supset$ ~[COMP]
 #17. [COMP] $\supset$ [+SUBJ]
 #18. V²[INF] $\supset$ [-SUBJ]
 #19. ([N] & [V]) $\supset$ [BAR 2]
 #20. [COMP, +ROOT] $\supset$ [COMP [NIL]]
 #21. [SLASH V] $\supset$ [SLASH V[FIN]]
 #22. [SLASH V] $\supset$ [+INV]

(The FSDs that GKPS propose that are not listed in (60) are either not needed on this analysis, or have been rewritten (eg., #1 and #2)). (60) contains four FSDs that have not been discussed yet. #19 simply encodes the fact that non-head daughters are usually phrasal. This is often assumed to be a part of X-bar theory, but not always (see Pullum, 1985, for discussion). The fact that root sentences tend not to have overt complementizers is recorded in #20.¹⁹ Again, this is a descriptive fact that is often ignored.

¹⁹The feature ROOT was proposed above (section 5.3.2) in regards to FSD #9, which expresses the fact that root clauses are tensed. All theories must employ something equivalent to ROOT, given the fact that languages often distinguish root from non-root clauses.

#21 and #22 simply reflect two assumptions I am making about inverted verb: categories bearing [SLASH V] are also marked as INVerted; and only FINite verbs invert in the default case (#21). No significant changes will be entailed if either or both of these assumptions prove to be false.

As should be clear from the discussion, all of these FSDs (and the FCRs in (48)) must be assumed to be universal, for an account of learning to work. It is not surprising that quite a few FSDs and FCRs have had to be added, given that GKPS state: ". . . we shall not attempt to state all the requisite FCRs, though it would be straightforward to do so." (p.29). The FCRs that I have added are simply definitional and the FSDs function to restrict the occurrence of various features except where they are either lexically specified or inherited from a rule. It does not seem unreasonable to assume that (60) and (48) are universal lists of feature dependencies.

5.4 Learning Rules

The program now is to take the fully specified trees that we are assuming are the data for children, strip off all features that are predictable, either by some principle of UFI, the FCRs or the FSDs, and see what rule is left. I am assuming that the child is confronted with a non-local tree and that her first job will be to break this

tree in to its local subtrees.²⁰ However, due to the size of the trees and the limitations of the page size, I will assume that this first step has already taken place and will only show here the local trees that go together to form the non-local tree. Consider first the trees in (61) (which underlie the sentence, They saw Mary).

²⁰Note that I am assuming that the has parsed and assigned a structure to a sentence for which her grammar has no rule. How children determine the tree structure and features for a novel sentence raises many interesting questions that will not be addressed here. This is a problem that every theory must confront, but it is rarely discussed (see, however, Valian, in press and Fodor, in press).

61) a.

BAR 2
-N
+V
-INV
-PAST
-PRD
-SLASH
+SUBJ
VFORM FIN
-RE
-WH
COMP NIL
-CONJ
-GER
-NEG
-NULL
-REMOR
-WHMOR
+ROOT

BAR 2
+N
-V
PER 3
+PLU
-PRD
-SLASH
-SUBJ
-RE
-WH
CASE NOM
-CONJ
-GER
-NEG
NFORM NORM
-NULL
-POSS
-REMOR
-WHMOR

BAR 2
-N
+V
PER 3
+PLU
-INV
+PAST
-PRD
-SLASH
-SUBJ
VFORM FIN
-RE
-WH
-CONJ
-GER
-NEG
-NULL
-REMOR
-WHMOR

b)

BAR 2
-N
+V
PER 3
+PLU
-INV
+PAST
~PRD
~SLASH
-SUBJ
VFORM FIN
~RE
~WH
~CONJ
~GER
~NEG
~NULL
~REMOR
~WHMOR

BAR 0
-N
+V
PER 3
+PLU
-INV
+PAST
~PRD
VFORM FIN
~RE
~WH
~CONJ
~GER
~NEG
~NULL
~REMOR
~WHMOR

BAR 2
+N
-V
PER 3
-PLU
~PRD
~SLASH
-SUBJ
~RE
~WH
CASE ACC
~CONJ
~GER
~NEG
NFORM NORM
~NULL
~POSS
~REMOR
~WHMOR

Let us first consider the local tree in (61a). The child can tell which of the two daughters is the head by looking at the HEAD features on each of the nodes. The right hand daughter must be the head, as it has the same (major class) HEAD features as the mother. Therefore, all of the HEAD feature specifications that the daughter shares with the mother (i.e., BAR, N, V, INV, PRD, SLASH, and VFORM) are predictable, and may be deleted and the metavariable H is inserted to show that this node is the head daughter.²¹ Its value for the AGR features is predictable from AGREE and may be deleted (cf. Chapter Four). Most of the rest of its feature specifications follow from universal FSDs, and may therefore be deleted. For example, the head daughter is undefined for CONJ (CONJ does not occur on this node). This is predictable from the FSDs and thus the specification ~[CONJ] may be removed. The node label in (62) is what remains.

62)

H
-SUBJ

²¹I am following GKPS and Pullum (1985) here in assuming that the notion of 'succession' is not necessary to X-bar theory. That is, I assume that BAR is a HEAD feature and needs to be included in the rule only if its value on the head daughter is different from the value on the mother. See discussion in Chapter Two.

The unpredictable features of the VP in (61a) are that it is the head daughter, and that, though the head daughter of an S it is not an S but a VP (i.e., [-SUBJ]).

Turning now to the other daughter, its BAR level is predictable from the fact that non-head daughters are default BAR 2. It is +N and -V because it is marked for NFORM. The value for NFORM is like PLU and PER in that its value never needs to be specified by a rule; any value can be freely instantiated. The distribution of expletive NPs as opposed to normal NPs will be handled by subcategorization as argued above (cf. Section 4.2.2.) The fact that the non-head daughter is undefined for SLASH, CONJ, GER, NEG, NULL, POSS, PRD, RE, WH, REMOR and WHMOR means that these features do not have to be mentioned in the rule, since these non-specifications follow from the FSDs. If the category were marked for any of these features, it would represent a fact that a child would have to learn and therefore include in the rule. That this NP is marked for [CASE NOM] is also irrelevant to the rule. We saw in Chapter Four that it did not matter which CASE was instantiated in a rule since only a CASE that is subcategorized for will ever be in a terminated tree. Therefore, CASE is like the AGR features in that it never needs to be specified in rules. So the CASE feature may be deleted along with the AGR features. This node is then left with the specification in (63).

63) [NFORM]

Let us now consider the mother node. From the fact that the mother node is specified as [VFORM] (whose value is universally free) it follows that it must be specified for [-N, +V]. Because the mother is [+SUBJ], the child can tell that it must also be <BAR 2, -N, -PAST, +V, COMP>. The lack of an overt complementizer follows from an FSD because this is a root clause. The features AUX, INV, PRD, SLASH, RE, WH, CONJ, GER, NEG, NULL, REMOR, AND WHMOR all have their default values, which allows them to be deleted. This leaves the child with the specifications in (64). Putting all this together, the rule that she will deduce from this local tree is as given in (65).

$$64) \quad \begin{bmatrix} +SUBJ \\ +ROOT \end{bmatrix}$$

$$65) \quad \begin{bmatrix} +SUBJ \\ +ROOT \end{bmatrix} \quad \rightarrow \quad [NFORM], \quad \begin{bmatrix} H \\ -SUBJ \end{bmatrix}$$

This is a fairly simple and reasonable rule.²²

Let us now look at the second local subtree, (61b), which is for the VP in the sentence They saw Mary. Again it will be clear to the child which daughter is the head. This time it is the rightmost daughter. Therefore, she can

²²As noted above, I am not considering LP statements here. See F&C and Fodor (to appear b) for discussion of the problems that LP statements cause.

strip off all the HEAD features (except BAR) on that node and introduce the metavariable H. Since the feature BAR is different, that information must be recorded at this node. The FSDs will take care of the specifications for RE, WH, COMP, CONJ, GER, NEG, NULL, REMOR, and WHMOR. Again, I am ignoring PERSON, NUMBER and GENDER on the assumption that they never need to be explicitly marked in a rule. This node is specified as in (66).

66)
$$\begin{bmatrix} H \\ \text{BAR } 0 \end{bmatrix}$$

Like the NP discussed above, the NP node in this tree need only be specified for NFORM. Moving on now to the mother node, the specifications for INV, PRD, SLASH, RE, WH, CONJ, GER, NEG, NULL, REMOR, and WHMOR all follow from the FSDs. The major category feature specifications (N and V) are predictable from the specification VFORM. The AGR features are universally free and hence will be deleted, as will be the feature PAST and the value for VFORM, which are also universally free. We are then left with the specifications listed in (67).

67)
$$\begin{bmatrix} \text{BAR } 2 \\ \text{-SUBJ} \\ \text{VFORM} \end{bmatrix}$$

The complete rule that the child has induced for this subtree is given in (68).

$$68) \quad \begin{bmatrix} \text{BAR 2} \\ -\text{SUBJ} \\ \text{VFORM} \end{bmatrix} \quad \rightarrow \quad [\text{NFORM}], \quad \begin{bmatrix} \text{H} \\ \text{BAR 0} \end{bmatrix}$$

While the rule in (68) may appear more complex and cumbersome than the equivalent GKPS rule, given in (69), that is largely because GKPS do not spell their rules out in features.

$$69) \quad \text{VP} \rightarrow \text{H}[2], \text{NP}$$

If we convert their rule into feature notation, as in (70), then we see that the only differences are minor.

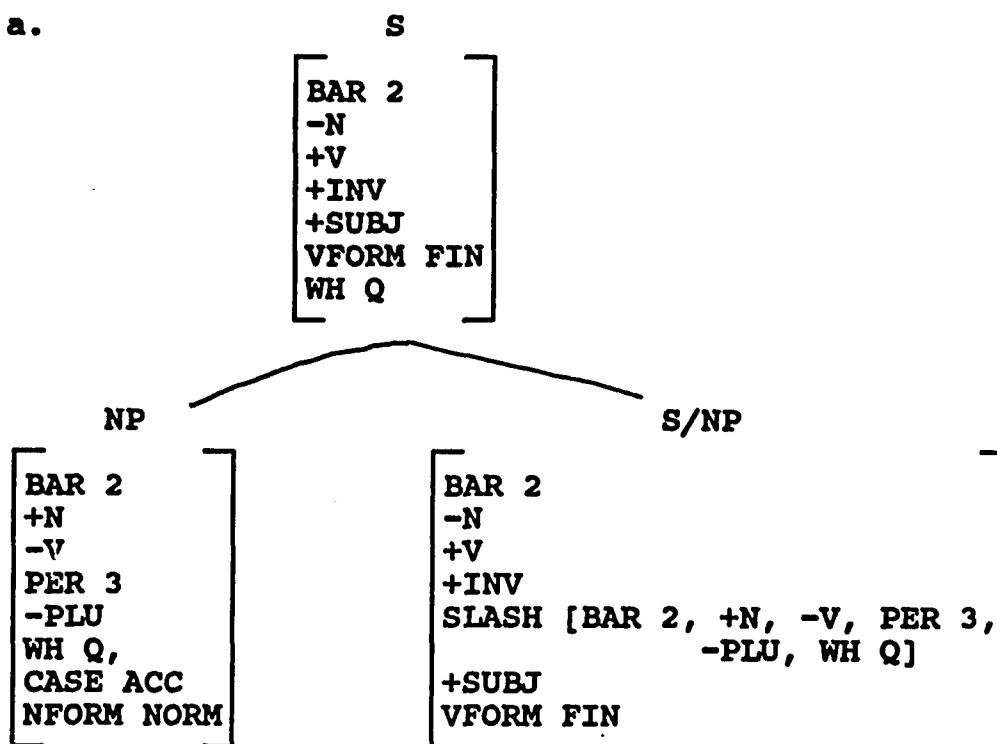
$$70) \quad \begin{bmatrix} +\text{V} \\ -\text{N} \\ \text{BAR 2} \\ -\text{SUBJ} \end{bmatrix} \quad \rightarrow \quad \begin{bmatrix} \text{H} \\ \text{SUBCAT 2} \end{bmatrix} \quad \begin{bmatrix} -\text{V} \\ +\text{N} \\ \text{BAR 2} \end{bmatrix}$$

In (68), I have replaced the major category features in the NP and VP with NFORM and VFORM, respectively. This option is available to GKPS also, given their implicational FCR 2 and FCR 3. I am assuming that subcategorization information is lexical; thus (68) contains no SUBCAT feature. Because of this, (68) must specify the BAR level of the head daughter while in (70), the BAR specification is implied by the SUBCAT feature. GKPS must specify the BAR level on the NP daughter since they do not assume (as I do) that non-head daughters are default BAR 2. Thus although there is some trade off among the feature specifications, these two rules are very similar.

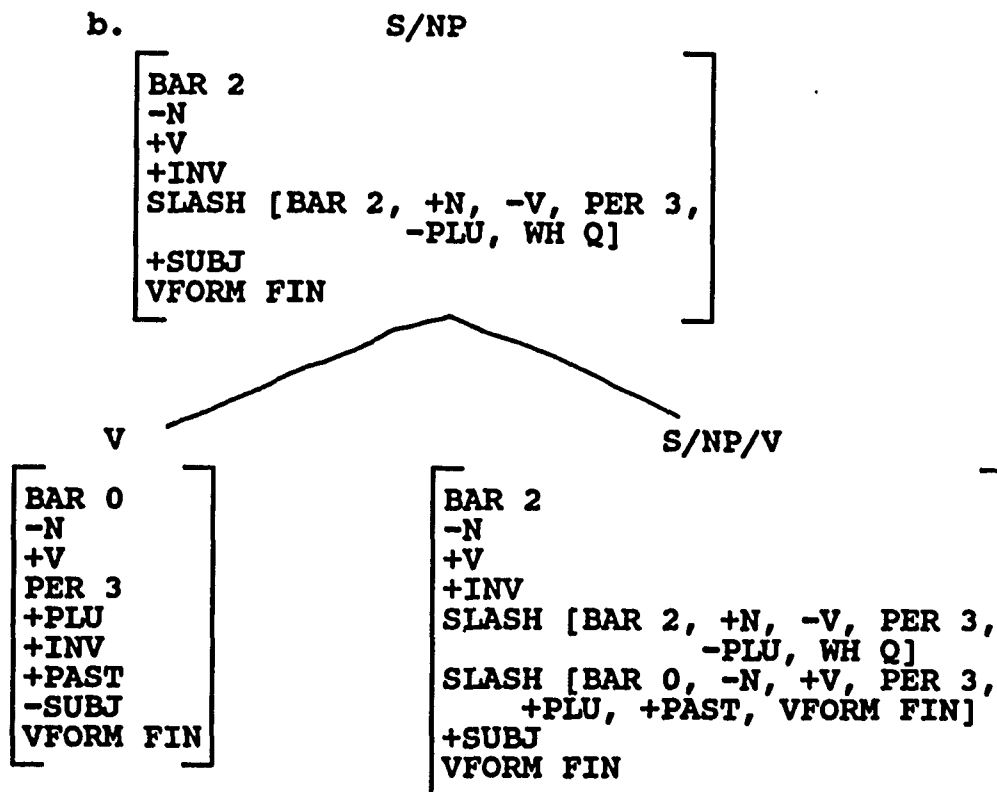
F&C point out that inducing rules in this way may make the absolute size of the grammar larger than GKPS' (though not in this case), but its relative size (in terms of the simplicity metric) is more in line with a theory of markedness. That is, the more constructions that a language has, the more rules it must include. This is opposed to a theory like GB, or the version of GPSG proposed by GKPS, where the lack of a parochial constraint will make the grammar simpler and the language larger. In this instance, however, we have found that our rule is either equivalent or even slightly less complex than the comparable GKPS rule.

Consider now what it would be for a child to learn the rules necessary for a sentence like Who did they see?, assuming the SAI as verb extraction analysis and not GKPS' metarule analysis. The requisite local trees are given in (71). (In all of the trees to follow, I will eliminate all features whose values are undefined. These follow trivially from the FSDs.)

71) a.

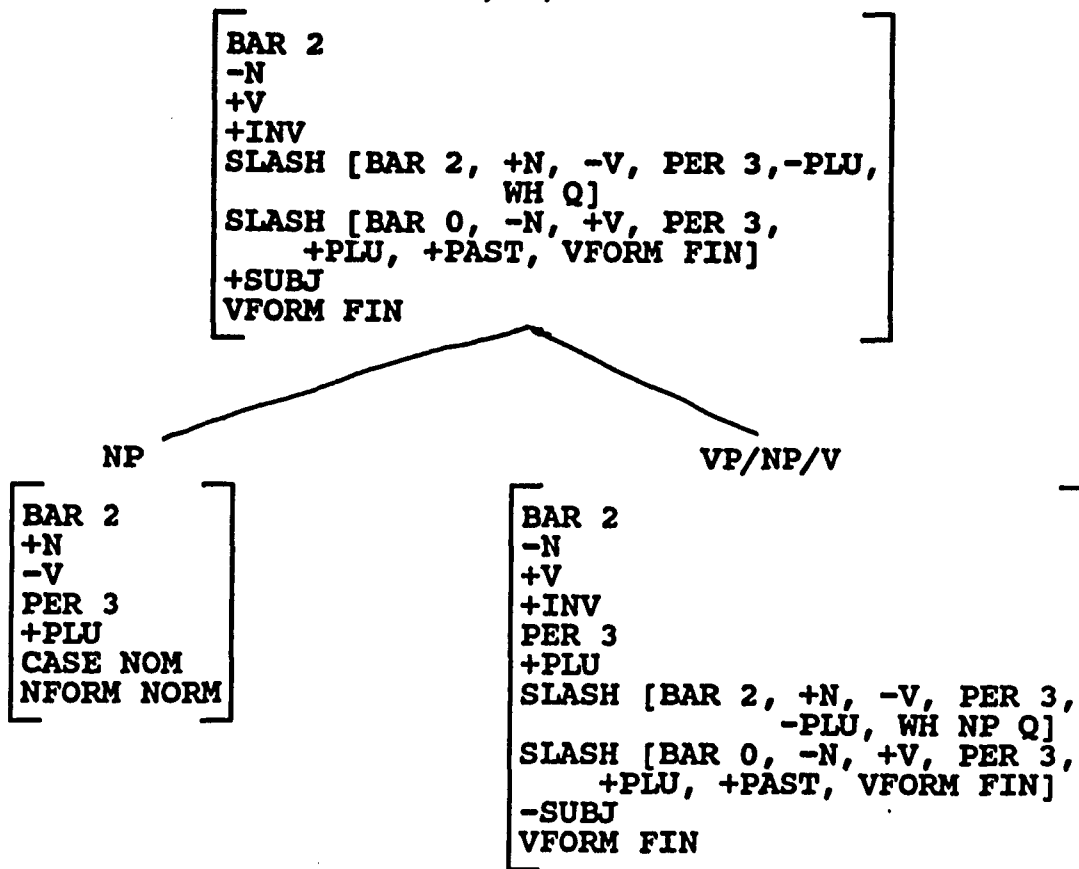


b.



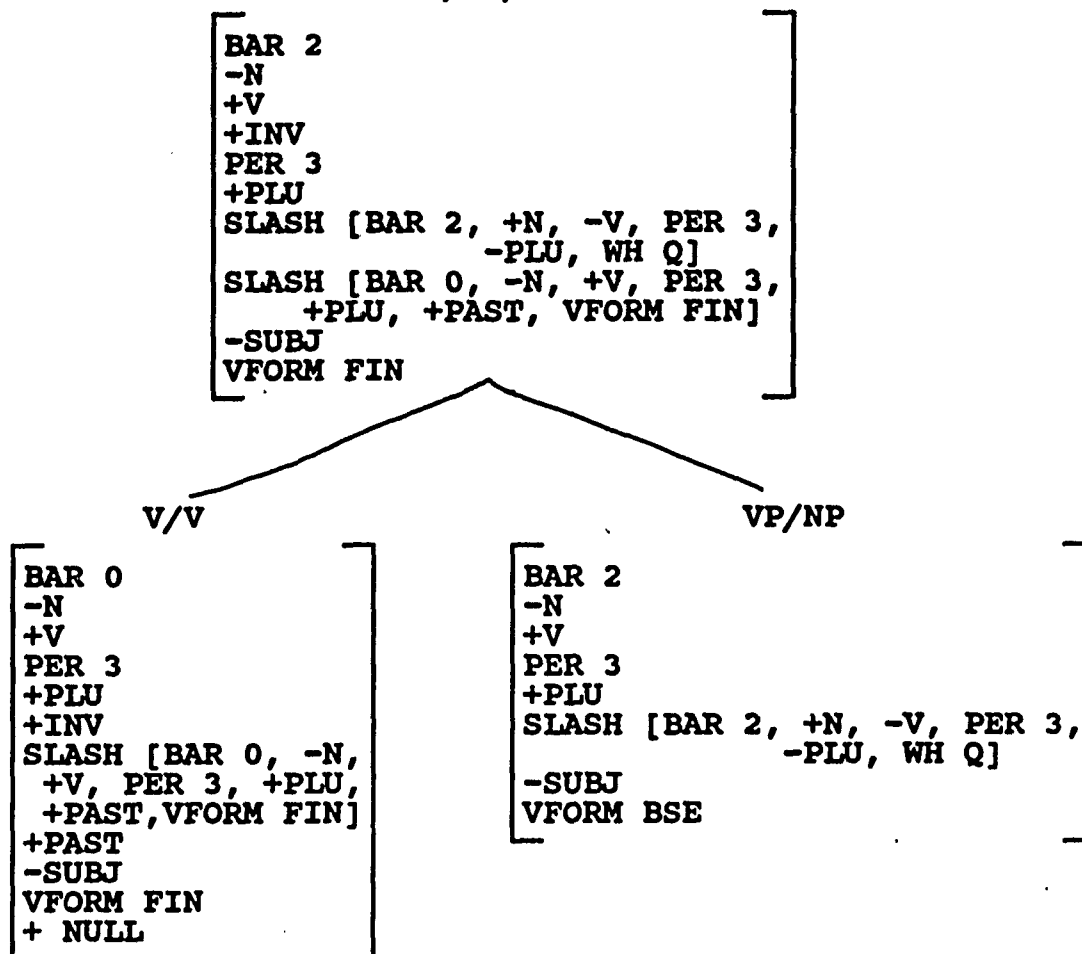
c.

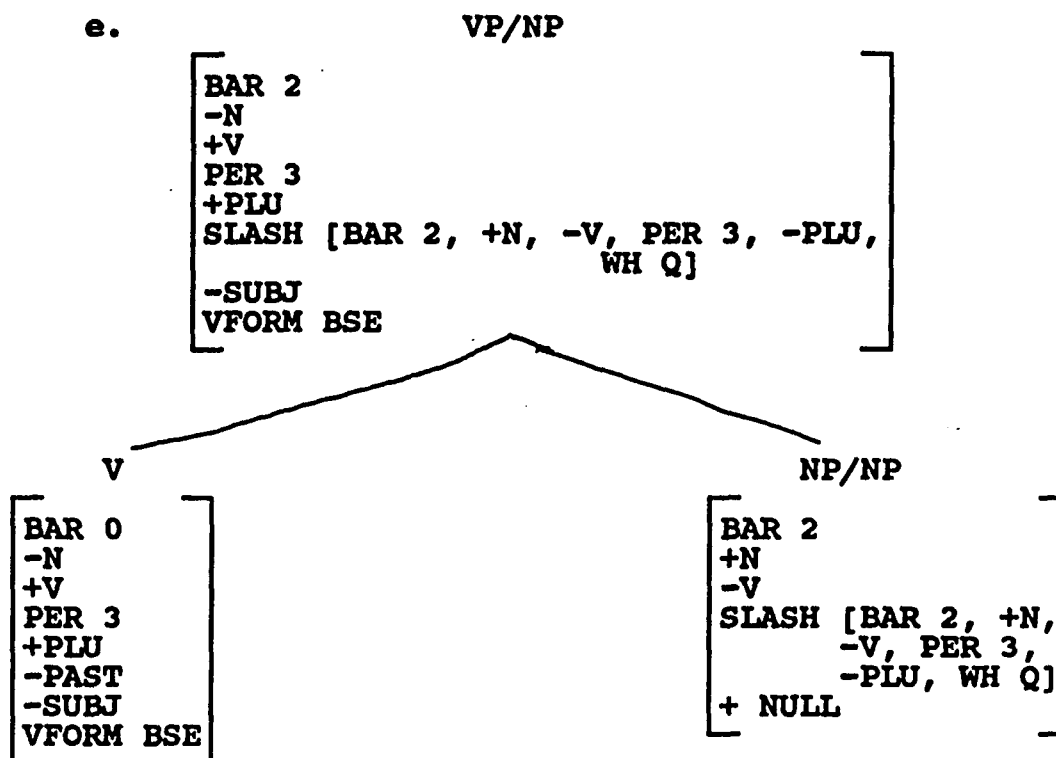
S/NP/V



d.

VP/NP/V





As was the case above, it is clear to the child which daughter in (71a) is the head, therefore all of the HEAD features on that node are removed and H is inserted. This leaves only the SLASH specification. As we saw above, the value for the SLASH is predictable, although the fact that the SLASH is there is not. Therefore, the node is as in (72).

72)



The NP node has its features removed just as above, except that this node is also specified as WH. WH is, however, a FOOT feature, therefore it must also be on the mother and that is where we have argued that such facts

must be recorded. The FFP will then force the WH specification down onto the NP daughter, given that English obeys the WH-island FSD. CASE, as we have seen above, is universally free and can be deleted on this node. The NP node is then marked as in (73).

73) [NFORM]

Turning our attention now to the mother node, the fact that it is VFORM (a feature whose value is universally free) follows from its major category feature values, which follow, along with BAR 2 from its having a subject. Thus, BAR, N, V, and VFORM can all be eliminated. WH is not predictable so it must remain. The rule for this local tree is given in (74).

74)
$$\begin{bmatrix} +\text{SUBJ} \\ +\text{INV} \\ \text{WH Q} \end{bmatrix} \rightarrow [\text{NFORM}] , \begin{bmatrix} \text{H} \\ \text{SLASH} \end{bmatrix}$$

Moving on now to the tree in (71b), it is not necessarily clear whether the [BAR 0] verb is the head or the [BAR 2] projection of V. They share (necessarily) several HEAD feature specifications with their mother and each disagree on a couple of HEAD features with their mother. I will assume that the child can determine that this is a topicalization structure and thus the rightmost daughter is the head in this construction.²³ Given that

²³Note that this is also a problem for the GKPS analysis with ternary branching here instead of binary

knowledge, then BAR, INV, N, SUBJ, SLASH [NP], V, VFORM, and the AGR features can all be eliminated as above. The value of the second SLASH feature ([SLASH V]) will be predictable (because this is a topicalization structure), even though the existence of the SLASH is not. Therefore, we will delete the value of that SLASH, but leave the SLASH feature itself. We are left then with the specifications in (75).

75)

H
SLASH

The fact that the 'extracted' verb is lexical (BAR 0) is not predictable from universal principles: it is an exception to FSD #19 in (60) above, and so that must be directly encoded in the rule. Its major category features follow from its specification for VFORM, which in turn follows from the [+INV] specification. Thus, those features will be deleted. This node must be [-SUBJ] since it is BAR 0, so the SUBJ feature will also be removed. The AGR features and PAST can also be removed, given that they are universally free. The non-predictable features are given in (76).

branching. For GKPS, the lexical V is necessarily the head because metarules only apply to lexical ID rules. But the child has no way of knowing that this is the output of a metarule. The child would be confronted with an S mother that has both V and VP daughters, and has to determine that the V is the head of that S.

76)

$$\begin{bmatrix} \text{BAR 0} \\ + \text{INV} \end{bmatrix}$$

Considering the mother node, the [+SUBJ] specification ensures that this is a sentence node: that is, it is [BAR 2, -N, +V, VFORM]. The value of the feature VFORM (which is FIN), again can be ascertained from the [+INV] specification and hence is not needed in the rule. The fact that this node is [+INV] and has the feature [SLASH NP[Q]] cannot be predicted from anything else in the tree, so these features must remain in the rule. From this tree, the child will induce the rule in (77).

77)

$$\begin{bmatrix} +\text{INV} \\ \text{SLASH NP[Q]} \\ +\text{SUBJ} \end{bmatrix} \rightarrow \begin{bmatrix} \text{BAR 0} \\ +\text{INV} \end{bmatrix} , \begin{bmatrix} \text{H} \\ \text{SLASH} \end{bmatrix}$$

In the next local subtree (71c), the VP daughter is obviously the head. Therefore, the only HEAD feature that needs to remain on this node is the one that is specified differently than its mother, namely [-SUBJ]. The AGR features are predictable, given AGREE, and they can be deleted. The NP daughter will reduce simply to the specification [NFORM] in the way that is now familiar. Now consider the mother node. Again, the fact that it bears SLASHes and what values those features have is not predictable and those features cannot be eliminated. The specifications for INV and SUBJ must also remain, as these

do not follow from any of the universal principles. The rule that the child will induce is given in (78).

$$78) \quad \left[\begin{array}{l} +INV \\ SLASH \ NP[Q] \\ SLASH \ V[FIN] \\ +SUBJ \end{array} \right] \rightarrow [NFORM], \quad \left[\begin{array}{l} H \\ -SUBJ \end{array} \right]$$

When considering the tree in (71d), we find a motivation for ordering the removal of the information that is predictable based on the FSDs and the PREHFC. The head daughter is [+NULL] because it is of the form $X[SLASH \ X]$, but the SLASH specification will be stripped off by the PREHFC. If the PREHFC is allowed strip off the features that it can (eg., [SLASH V]) before the FSDs strip off the information that they can, then the specification for NULL will not be the default value.²⁴ Thus it must remain in this rule, and that is surely wrong. Instead, what must happen is that the FSDs must remove the relevant values (eg., [+NULL]) and then the PREHFC can strip off its values. Given this, second, ordering, then the only feature that will not be eliminated is BAR; the rest will be stripped off as we have seen above.

The major category feature values of the non-head daughter follow from its specification for VFORM. The fact that the head verb of this VP must be in its uninflected

²⁴F&C make a similar point with regard to the order in which the FFP and the FSDs remove information from a tree.

form (i.e., [VFORM BSE]) does not follow from any of the universal principles; it is an idiosyncratic subcategorizational property of the verb *do*, and thus it need not be recorded in this node (but it must be recorded in the lexical entry for *do*). The SLASH feature and its value on this node is predictable and will be deleted. As with NP daughters, this VP's BAR level will follow from the general principle that non-head daughters are [BAR 2]. Thus, the only specifications that are left are for the features VFORM and SUBJ.

Turning now to the mother node, its major category features are predictable from the fact that it is specified for VFORM. That it is [VFORM FIN] follows from its being [+INV], so that feature may also be removed. The major category values for the arguments of the SLASHes are not predictable, but once the child learns other rules for passing PPs and APs through a VP, this rule will collapse with them. The FIN value of [SLASH V] is the default, as is [+INV], and both can be eliminated. The rule that is induced from this tree is as in (79).

$$79) \quad \left[\begin{array}{l} \text{BAR 2} \\ \text{SLASH NP[Q]} \\ \text{SLASH V} \\ \text{-SUBJ} \end{array} \right] \rightarrow \left[\begin{array}{l} \text{H} \\ \text{BAR 0} \end{array} \right], \left[\begin{array}{l} \text{-SUBJ} \\ \text{VFORM} \end{array} \right]$$

The tree in (71e) will reduce in more or less exactly the same fashion as the tree in (61b). The

differences will be that the specification for VFORM is different (but it will be treated in a comparable manner), and that a SLASH specification is needed on the mother. The appropriate rule is given in (80).

$$80) \left[\begin{array}{l} \text{BAR 2} \\ \text{VFORM} \\ \text{SLASH NP[Q]} \\ \text{-SUBJ} \end{array} \right] \rightarrow [\text{NFORM}], \left[\begin{array}{l} \text{H} \\ \text{BAR 0} \end{array} \right]$$

Note that SLASH and NULL are not marked on the NP daughter. The specification for NULL follows from an FSD, and the SLASH, which must pass down from the mother by the FFP, can only pass to the NP daughter; it cannot appear on a [BAR 0] category. As pointed out by F&C, the FSD must operate first to remove information before the FFP because when the FFP removes [SLASH NP] from this node it destroys the environment which makes [+NULL] the default.

Given the system of rules developed here, it appears to be possible for extract appropriate rules for her language, assuming that she is confronted with the relevant trees. Thus, the theory which was developed on linguistic grounds seems to be completely compatible with the requirements that F&C place on theories based on learnability reasons. This is an exciting finding.

CHAPTER SIX

CONCLUSIONS

6.1 Summary of Arguments

This study started with an in depth look at GKPS' version of GPSG. GKPS' theory is impressive in the scope of its coverage of English and in the interrelatedness of the principles proposed. It is however also highly complex. In Chapter Three we saw that we could greatly reduce much of that complexity if, following Shieber (1986), we impose an ordering on the principles, rather than allowing them to apply simultaneously as GKPS did. This necessitates a switch from GKPS' view that the various principles only check feature values that have been either inherited or freely instantiated to the view that the principles actually supply the features.

In working out the details of the ordering, it was argued that Shieber had followed GKPS too closely and that other changes were needed, not only to achieve simplicity but also to maintain descriptive adequacy. In

particular, it was argued that the FSDs could not be split into two different groups such those that affect lexical categories apply before the HFC and those that affect non-lexical categories apply after the HFC. Rather, there must be only one set of FSDs and they must apply last in the ordering. Otherwise, they are of little use as defaults. This ordering then led to the conclusion that non-default values, if they are to occur in a tree, must be inherited from the ID rule that licenses that tree. This means that there is almost no 'free instantiation' as GKPS envision it, but rather that the principles of UFI are procedures that copy feature values from one node to another, within a local tree.

Assuming this ordering, Chapter Four then examined the CAP, which is the locus of much of the complexity that carries over from GKPS' unordered system of principles to the system of principles with an explicit ordering statement. GKPS make use of the CAP in many different constructions to explain several different linguistic phenomena (e.g., subject-verb agreement, control, top-linking, tough-constructions, subject-raising, etc.). It was argued that the generalization being expressed by the CAP was spurious, and that the simplification which followed from separating the analyses the CAP was collapsing, more realistic generalizations could be expressed. Two results follow

from breaking up the CAP. First, agreement features are seen to be features which (almost) all daughters of a maximal projection share. Second, the principles of UFI become very similar: the FFP copies FOOT features from mother to some daughter; the PREHFC (what is left of GKPS' HFC) copies HEAD features from the mother to the head daughter; and the ADP (part of GKPS' CAP) copies agreement features from mother to all daughters.

This theory, as proposed so far, is very similar to the theory argued for by F&C, the difference being that they arrive at their conclusions based on learnability considerations and we have only looked so far at the linguistic motivations for modifying GKPS. Chapter Five examines the further changes that must be made to make the theory proposed here compatible with F&C and hence learnable. The only changes required are eliminating the parochial constraints (the parochial FSDs and FCRs). Once this is done (which could probably be linguistically motivated also, if we had a general theory of features and a restrictive theory of FSDs and FCRs; see section 6.2.2 below), it was shown that the grammar proposed here for English could be learned by a child, assuming no negative data and a general, symbol counting simplicity metric.

6.2 Further Development of the Theory

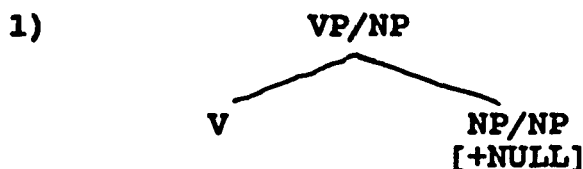
Assuming that the theory proposed so far is well-motivated and heading in the right direction, we can still ask the question, is it possible to simplify the theory? That is, is there still theoretical machinery which we could do without, without losing descriptive coverage? Given the kind of theoretical framework assumed here, there are several places to look for such simplification.

6.2.1 Eliminating extrinsic ordering

GPSSG, as developed by GKPS, contains highly complex principles that are unordered yet interact in complex ways which are determined by the complex definitions of the principles. Following Shieber (1986) we are able to extract an explicit ordering statement from the principles, thereby reducing the complexity of those principles. The ordering statement is, however, a stipulation, not an explanation and is in fact an added complexity in and of itself. It seems to be a worthwhile addition, given the simplification it allows, but the theory would be nicer if it could do without the stipulation altogether. It was briefly mentioned in section 5.4 that feature stripping requires the feature instantiation principles to be ordered in a certain way (namely, with the PREHFC before the FSDs). This is

compatible with the ordering that was argued for on linguistic grounds in Chapter Three. However, it is known that reversing the order of a derivation can sometimes create ambiguities (e.g., it was found in the early days of TG that a well ordered system of transformations for deriving surface structures from deep structures did not simply reverse to derive deep structures from surface structures). Therefore, it is important to inspect more closely the ordering we have been assuming.

The question to be addressed is, are there ordering interactions that occur between FSDs and the FFP, the PREHFC and the ADP, and are there orderings that must hold between the various FSDs? We have seen an ordering effect between the principles of UFI and the defaults in the interaction between the FFP, and the FSD for NULL. In this case, for sentence generation, the principles of UFI must operate before the FSDs, so that the appropriate context is available for the correct FSD to apply. As noted in section 5.4 above, in terms of language learning (i.e., feature stripping), the FSD must first strip off its feature(s) before the FFP removes the feature(s) that are the context for the FSD. Consider the tree in (1).



Given the FSDs in (2), [+NULL] is the default on the NP daughter only as long as that NP is specified as NP/NP.

- 2) a. $X/X \supset [+NULL]$
 b. $\sim[NULL]$

Once the FFP has removed the predictable SLASH[NP] from the daughter, [+NULL] is no longer the default and hence no longer predictable and must remain in the rule.

Therefore, for feature stripping to work as rule learning, the FSDs must be applied (in reverse) before the FFP applies.

Fodor (to appear b) argues that the principles of UFI are all defaults and not absolute requirements. This is not unreasonable since even for GKPS the HFC was explicitly a default principle and even the FFP and the CAP could be overridden in certain circumstances. Given that the principles of UFI are defaults, like the FSD, Fodor then proposes that this interaction between the FFP and the FSDs for NULL is an instance of a general Elsewhere Condition which makes a more specific default operate before a less specific default. Thus, Fodor is collapsing the distinction between the relational principles of UFI (which are inter-category) and the FSDs (which are intra-category). She calls both of these

types of defaults Local Tree Defaults (LTDs); the only difference between them (especially once it is established that for learnability they all must be innate) is the scope of the context that is relevant to determining the default value. Further details of this proposal remain to be worked out, but the results in this case are acceptable. In sentence generation, the FFP will first copy down the SLASH[NP] from the mother to the daughter, and then the FSD in (2a) can apply to provide the [+NULL] specification. In rule learning, first the FSD will operate to remove the [+NULL] and then the FFP will eliminate the SLASH[NP] specification from the NP daughter. If the correct ordering follows from a general condition like the Elsewhere Condition, then the ordering need not be explicitly stated.

In section 5.3.1 above, it was noted that there can be almost no useful ordering statements among the FFP, the PREHFC and the ADP, given that they affect disjoint subsets of features. The only case of overlap is with SLASH, which is both a HEAD feature (and therefore falls under the PREHFC) and a FOOT feature (and therefore falls under the FFP). If there is any conflict here (and it is not clear that there is), it can be resolved by the Elsewhere Condition that Fodor has suggested to fact resolve conflicts between defaults. Recall that the Elsewhere Condition causes the most

specific rule, whose environment is met, to apply before (or instead of) a less specific rule. The PREHFC counts as more specific than the FFP, because it refers to the head daughter and not just any daughter; thus it should take precedence over the FFP. So, SLASH must pass to the head daughter if possible and to any other daughter if not to the head. Thus, the Elsewhere Condition may also be of use in intrinsically ordering the principles of UFI.

This kind of general ordering principle will simplify the grammar, but it will not necessarily ensure no mistakes in the learning process. There are two types of cases where the child might make an error, i.e., strip the wrong information from the tree. The first type (which we will call an "egregious") is where the child mistakenly removes a feature from a tree because it *seems* to fall under an FSD, and the second type (which we will call "non-optimal") is where the child has a choice of which feature to strip and makes a non-optimal choice. Non-optimal problems are points of non-determinism in the learning process, and thus the child is open to making mistakes.

From this point on, most of the FSDs discussed in this chapter will be abstract required rather than

examples of FSDs attested in natural languages.¹ The reason for this is simply that the relevant analyses have not been done yet. Until we have more data regarding what FSDs other languages require, we will not have concrete examples of potential ordering problems. For instance, no FSD of the form (3b) below seems to be required for the description of English.² Perhaps, though, this type of FSD is evidenced in some other language.³

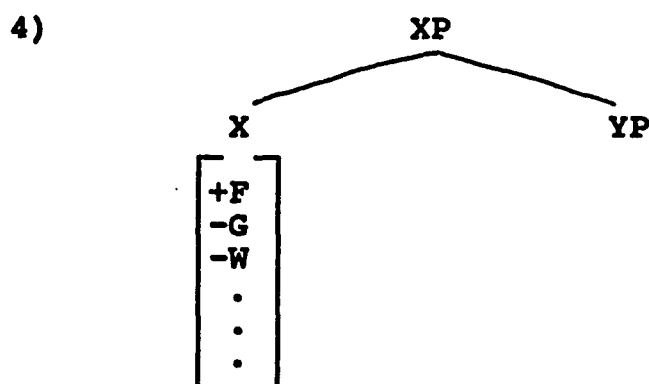
Consider the FSDs in (3) and the category X in (4).

- 3) a. $[+F] \supset [-W]$
 b. $[-G] = [+F]$

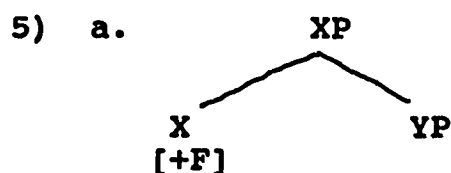
¹For ease of exposition, I will further restrict my attention here to Boolean features.

²In fact, as far as I know, no FSDs of the type in (3b) below have been proposed for natural languages. GKPS postulate FCRs of this form (such as the FCR equating $[+SUBJ]$ with $[COMP]$), but those were all argued against in Chapter Five. Chapter Five does, however, posit other FCRs with this form, but they are the boringly definitional FCRs (e.g., the FCR equating $[VFORM]$ with $[+V, -N]$). Thus, the FSD in (3b) is somewhat questionable.

³This claim does not counter F&C's requirement that *all* FSDs must be universal and therefore innate. It could well be the case that there are universal FSDs which all of the languages that we are familiar with obey completely and so we have never noticed that they in fact exist. Only further research will bring them to light.



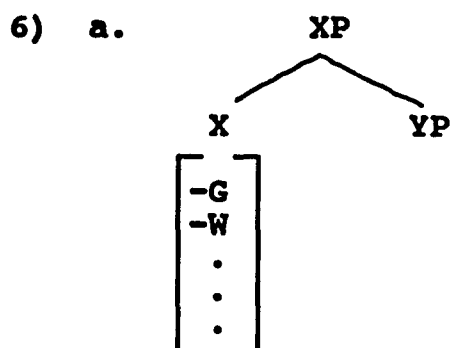
When stripping the features from this category in an input tree, the child is faced with a non-deterministic choice. (3b) allows the child either to remove [+F] in the environment of [-G], or to remove [-G] in the environment of [+F]. Except where a more specific FSD applies, [+F], [-G], and [-W] will have the same distribution because of (3b). But nevertheless it matters which one is stripped off. If the child chooses correctly and [+F] remains in the tree, then all the other features may be stripped. The relevant tree and rule will be as in (5).



b. $XP \rightarrow H[+F], YP$

When confronted with (4), if the child removes [+F] based on (3b), then she will have no way to tell that [-W] is the default value, so she will retain it in the tree and thus it will be explicit in her rule and her grammar will

therefore be more complex than it needs to be (she will have made a non-optimal error). Her stripped tree is given in (6a) and her rule in (6b).



b. $XP \rightarrow H[-G, -W], YP$

For reasons of absolute grammar size, we want the child to extract to (5b) and not (6b) from input (4). An efficient grammar would include (5b) if possible and not (6b) where (5b) is possible. However, it is not necessarily the case the natural language grammars are as efficient as linguists would want.⁴ If grammars are not always maximally simple, what is the outcome? Assuming that this kind of error can make it into the adult grammar, that grammar will be slightly larger than necessary, but it will generate the appropriate trees. So, in this case, it would seem not to matter. There is only one feature difference between the target rule and the wrong rule, and one feature, more or less, should not

⁴This is a logical possibility which raises very interesting issues which are, unfortunately, beyond the scope of this work.

matter much to the simplicity ranking of different grammars. But, there is an potential problem that is not obvious because we are looking at a simple example. A child's wrong choice for any given FSD could cause her to leave several features in a rule, not just one. And, she could make non-optimal choices with regard to several FSDs in extracting a rule from any tree. Thus, many rules could contain many redundant features and over the entire grammar, that could and should matter.

In addition, a grammar that includes redundant features misses generalizations. For example, if the child has the capability of using the metavariable 'H' to stand in for the head of an ID rule, and she explicitly marks in her rules the values for [N] and [V] on all nodes in her rules, then her grammar does not express the generalization that ID rules have heads. Thus, because we want grammars to be maximally simple and to capture relevant generalizations, we need a way to ensure either that the child does not make this kind of mistake, or that if she does, she can recover from it.

One possible way out of this potential source of difficulty is to make the child induce both possible rules, and then examine the absolute size of and the language generated by her current grammar with the addition of one of the rules as compared to the addition of the other. It seems obvious that children are not

doing this kind of computation. If they are faced with this kind of choice, it seems to be made rather simply and mechanically. Another possible solution to this problem would be the discovery of some general criteria to guide the decision of which feature to keep and which feature to remove when given a choice. I do not know of any which will work. For instance, superficial complexity in the FSD will not work. Consider (7).

7) $[-G, +H] \equiv [+F]$

The notion of superficial complexity would lead the child to strip $[-G, +H]$ in favor of $[+F]$ simply because $[+F]$ contains less symbols. But it is easy to imagine a situation where keeping $[-G, +H]$ in a tree would allow the removal of eight more features while having $[+F]$ in the rule allows only the stripping of $[-G, +H]$. Thus, superficial complexity would lead to the wrong result. If any such criteria exists, it is not readily available.

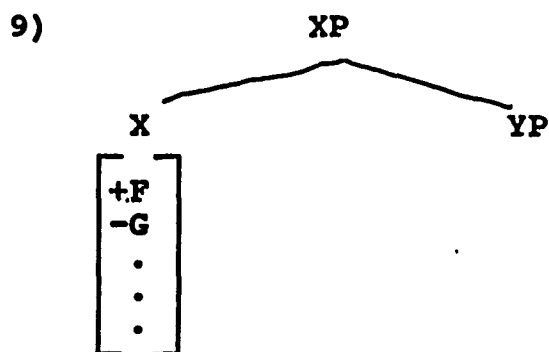
The one solution that is available to this non-determinacy (non-optimal) problem is to assume the child randomly chooses which side of an equivalence to remove and which side to keep in a rule. Then at some point in her experience, the child should have extracted both rule (6b) and rule (5b) from input (4). If we further assume with F&C that, all other things being equal, simpler rules are used more frequently in parsing and generation, and that unused rules wither from the grammar (based on

lack of use), then the unwanted rule in (6b) will be eliminated from the grammar over time.

The other case of potential feature stripping problems (the egregious problems) is where the child thinks that the feature falls under a default when it actually does not. F&C's example using SLASH and NULL is an example of this kind of problem. The general problem is that the child could strip off too much information and thus derive the wrong rule. This problem arises in the case of FSDs of the form in (8).

- 8) a. $[+F] \supset [+G]$
 b. $[-G]$
 c. $[+F]$

What these FSDs say is that in general $[-G]$ is unmarked (not included in rules), but in the context of $[+F]$ (itself unmarked) the unmarked value is $[+G]$. Assume now that the child is confronted with the tree in (9) and must strip features to derive a new rule for her grammar.

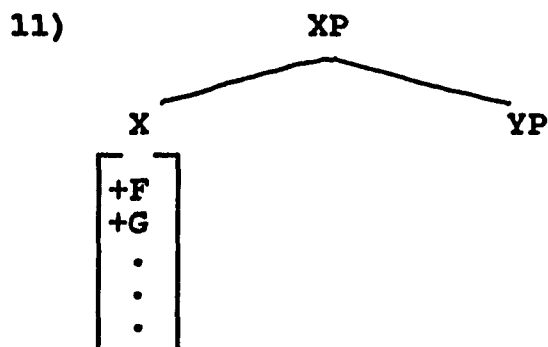


The $[-G]$ in this tree is actually marked, because of the presence of $[+F]$. If the child mistakenly removes $[+F]$ (which is the default), then the $[-G]$ in (9) will appear

to be the default, so the child will then remove it. Then she will have neither feature specified in her rule, which is given in ((10)).

10) $XP \rightarrow H, YP$

This is incorrect. The rule she has will overgenerate. In generation, the default for F is [+F] and (8a) takes precedence over (8b) (by the Elsewhere Condition) so that (10) will generate (11) but not (9).



As was the case with non-optimal problems, the relevant question to ask here is, is there a general criteria that the learner can follow in this case to stop her from making a egregious mistake? In this instance, the answer seems to be "yes". In generation, we use the Elsewhere Condition to cause the default with the most context specified to take precedence over a default with less context. In feature stripping, the Elsewhere Condition requires that the learner try to use the most specified FSD (which will be (8a) in this case) before using a less specified FSD ((8b) in this case). What the child must do then, when contemplating stripping a

feature from a tree, is to check her list of FSDs and see if that feature occurs on the right hand side (RHS) of any other FSD and if so, she must compare the context specified in each so that the most specific FSD will apply. It is interesting to note that what is needed in feature stripping is exactly the Elsewhere Condition and not an inverse of the Elsewhere Condition. If the FSDs were extrinsically ordered, then we would expect the orders for generation and feature stripping to be mirror images of each other. However, assuming that we can do without an extrinsic ordering statement, then the general Elsewhere Condition will be in effect in both feature stripping and generation.

It might be the case that it is impractical for the child to the kind of computation necessary for this proposal to work. If, for example, it turns out that there are only a small, finite number of features possible universally in local trees, and further that no feature occurs as the RHS of more than some small number of FSDs, then the computation required to determine which FSD should apply will be small. However, if there is a large number of possible features in local trees and any given feature may be the RHS of any number of FSDs, then the computations could be quite complex and it seems unreasonable to expect the child to perform them. Even though we are assuming that all FSDs are universally

specified (hence innate) and we can assume that all features are universally specified, we simply do not know enough about the number of features or the number and interrelations among the FSDs to address these issues here (but see section 6.2.2 below for a brief discussion).

Even if there is no linguistically motivated solution to egregious problems, there is a learning theoretic solution just as there was above for non-optimal problems. The result of a egregious error is that the child has induced an overgenerating rule, (10), based on a input (9). However, this is a mistake that she can recover from because she will not be able to generate the data that motivated this rule. The target is (9), but she will only derive (11). Thus, the child will not only overgenerate (11), she will also undergenerate (9). This feedback will tell her that she has made an error and that she must either start again, or ignore this data and just go on.

It appears then to be reasonable to claim that the grammar consists of universal principles (FSDs, FCRs, the FFP, the ADP, etc.) which operate in an unordered fashion, guided by a very general Elsewhere Condition. These grammatical principles are used in sentence generation and in language learning. We have seen that the Elsewhere Condition will resolve some cases of

potential ordering problems, but further investigation is required to determine if this solution will resolve all ordering problems. But we seem to have arrived at a kind of an impasse, which can be resolved only when we know more about the universal set of features and the possible restrictions on their occurrence.

6.2.2 A general theory of features

As noted several times above, we do not know what the possible features are, and what kinds of restrictions hold over the form of FSDs and FCRs. Until we have detailed analyses of more languages, written in feature notation, and until we have more of an idea what the descriptive facts are, there are many interesting questions that we simply cannot address. For instance, as noted above, we cannot tell at this point whether the fact that English gives no evidence for an FSD of the form in (12) is accidental or whether in fact no FSD with that form can exist.

12) $[-G, +H] \Rightarrow [+F]$

Assuming that there are FSDs which we are missing, due to our lack of data, then it would seem that our analyses of English are more complex than they need to be, or are just wrong, but we cannot tell that without knowing about the FSDs we are missing.

Additionally, we do not know what the possible features are and that also could be handicapping our analyses. For example, on the analysis of SAI presented here, the feature AUX is not needed, and in fact INV could almost be eliminated also. Those features, however, play a large role in many (if not most) analyses of SAI. If AUX is not in the inventory of possible features, then this analysis (without that feature) is better than analyses that require that feature. If, however, AUX is a permissible category, then it is less clear how to judge between analyses that uses it and analyses that do not.

Pullum (1985b) points out that it is not necessarily the case that there are a finite number of features or a finite number of feature values. He discusses the agreement systems of the Bantu languages, where the feature GENDER seems to have 20 to 30 values. While this might in fact be true, it is something that we cannot know until we have a theory of what the possible features are. For psychological/acquisitional reasons, we would like there to be a finite (preferably small) set of features universally available, with each language choosing a subset of the features. Pullum could be right that there is in fact an infinite set of features, but at this point in time, we simply do not know what the facts are.

6.2.3 Implementation

The final development of the proposals here that is so blatantly waiting to be done is the machine implementation of this theory. The system that we have developed is quite mechanistic and the interactions of the various components is fairly straightforward. Therefore, the implementation looks simple. This undoubtedly will not be the case, and the problems that the machine runs into in stripping the features out of trees and the inappropriate rules that the machine derives will give us a lot of feedback about what parts of the theory need further revision.

The approach that seems most promising at the moment, is to take seriously the claim that the grammar contains no extrinsic ordering statements. If there is no extrinsic ordering on the principles of feature instantiation, then when stripping features from trees to extract rules, the child should be able to begin the process with any feature. It seems as if this might work, if the child is using something like the following informal procedure.

- i) until all (Boolean) features have been picked, pick a (Boolean) feature at random, to decide whether or not it can be deleted.

- ii) check that feature as RHS in all FSDs, to see if it is unmarked in this context, obeying the Elsewhere Condition. if not, write *m* and return to feature (if any) that led you here and write *u*.
- iii) if unmarked, then check the feature as LHS in FSDs to see what it feeds. if it feeds any FSD, go to (ii) on the fed feature.

Clause (i) represents the fact that it does not matter in general which one starts feature stripping with. The Elsewhere Condition is enforced by clause (ii), which avoids 'egregious' errors. 'Non-optimal' errors are avoided because clause (iii) causes you to search the chain of implicational relationships. This work remains to be formalized, but it seems promising, at least at first blush.

REFERENCES

- Abney, Steven P. 1987. 'The English Noun Phrase in its Sentential Aspect'. Ph.D. dissertation, MIT
- Bach, Emmon and Barbara H. Partee. 1980. 'Anaphora and semantic structure'. In Jody Kreiman and Almerindo E. Ojeda (eds.), Papers from the Parasession on Pronouns and Anaphora, Chicago: Chicago Linguistic Society, 1-28.
- Baker, Mark C. 1985. 'Incorporation: A theory of grammatical function changing.' Ph.D. dissertation, MIT.
- Barwise, J. and J. Perry 1983. Situations and Attitudes. Cambridge, MA: MIT Press.
- Bresnan, Joan. 1971. 'Sentence stress and syntactic transformations'. Language (47), 297-321.
- _____. (ed.), The Mental Representation of Grammatical Relations. MIT Press, Cambridge: MA.
- Chomsky, Noam. 1957. Syntactic Structures. The Hague: Mouton.
- _____. 1965. Aspects of the Theory of Syntax. Cambridge, MA: MIT Press.
- _____. 1970. 'Remarks on nominalization'. In R. Jacobs and P. Rosenbaum (eds.), Readings in English Transformational Grammar. Waltham, MA: Blaisdell. Also reprinted in Noam Chomsky, 1972, Studies on Semantics in Generative Grammar. The Hague: Mouton.
- _____. 1982. Lectures on Government and Binding. Dordrecht: Foris.
- _____. 1986. Barriers. Cambridge, MA: MIT Press.
- _____. and Howard Lasnik. 1977. 'Filters and Control'. Linguistic Inquiry. 8, 425-504.
- _____. and Morris Halle. 1968. The Sound Pattern of English. New York: Harper and Row.
- Chung, Sandra and James McCloskey. 1983. 'On the interpretation of certain island facts in GPSG'. Linguistic Inquiry. 14, 704-713.

- Culy, C. 1985. 'The complexity of the vocabulary of Bambara'. Linguistics and Philosophy, 8, 345-351.
- Earley, J. 1970. 'An efficient context-free parsing algorithm. Communications of the Association for Computing Machinery. 14:453-460.
- Emonds, Joseph E. 1976. A Transformational Approach to English Syntax: Root, Structure-preserving and Local Transformations. New York: Academic Press.
- Engdahl, Elisabet. 1980. 'The syntax and semantics of questions in Swedish'. PhD. dissertation, University of Massachusetts at Amherst.
- _____. 1989. 'Parameters and parameter-setting in a phrase structure grammar. in Language Processing and Acquisition, L. Frazier and J. de Villiers (eds.), Kluwer, Dordrecht: Holland.
- Fodor, Janet Dean. 1983. "Phrase structure parsing and the island constraints." Linguistics and Philosophy, 6, 163-223.
- _____. 1990. 'Cross serial dependencies and subcategorization percolation. CUNYforum 15, City University of New York, 53-75.
- _____. in press. 'Learnability of phrase structure grammars. In R. Levine (ed.), Formal Grammar: Theory and Implementation, Vancouver Studies in Cognitive Science, University of British Columbia Press.
- _____. to appear a. 'Islands, learnability and the lexicon. in H. Goodluck and M. Rochemont (eds.) Island Constraints: Theory, Acquisition and Processing, Kluwer Academic Press, Dordrecht: The Netherlands.
- _____. to appear b. Learnability of phrase structure grammars. in R. Levine (ed.), Formal Grammar: Theory and Implementation. Vancouver Studies in Cognitive Science, University of British Columbia Press.
- _____, & Steven Crain. (in prep.) On the Form of Innate Linguistic Knowledge. to be published by Bradford Books.

- Fodor, Janet Dean and Charles Jones. 1986. Connectedness, C-command, and GPSG. (ms)
- Gazdar, Gerald. 1981. 'Unbounded dependencies and coordinate structure'. Linguistic Inquiry. Vol. 12(2), pp. 155-185.
- _____. 1982. 'Phrase structure grammar'. in Pauline Jacobson and Geoffrey K. Pullum (eds.), The Nature of Syntactic Representation. Dordrecht: Reidel, 131-186.
- _____, and Geoffrey K. Pullum. 1982. 'Generalized phrase structure grammar: a theoretical synopsis. Bloomington, Indiana: Indiana University Linguistics Club.
- _____, Ewan Klein, Geoffrey K. Pullum, Ivan A. Sag. 1985. Generalized Phrase Structure Grammar. Cambridge, MA: Harvard University Press.
- Horrocks, Geoffrey. 1983. 'The order of constituents in Modern Greek. In Gerald Gazdar, Ewan H. Klein, and Geoffrey K. Pullum (eds), Order, Concord and Constituency. Dordrecht: Foris, 95-112.
- Jackendoff, Ray S. 1977. X-bar Syntax: A Study of Phrase Structure. Cambridge, MA: MIT Press.
- Jacobson, Pauline. (1984) Connectivity in Phrase Structure Grammar. Natural Language and Linguistic Theory (1), 535-581.
- _____. 1987. Review of Generalized Phrase Structure Grammar, by Gazdar, Pullum, Klein and Sag. Linguistics and Philosophy 10, pp. 389-426.
- Kaplan, Ronald M. and Joan Bresnan. 1982. 'Lexical-Functional Grammar: A formal system for grammatical representation. In Joan Bresnan (ed.), The Mental Representation of Grammatical Relations. MIT Press, Cambridge: MA. 173-281.
- Kayne, Richard. 1984. Connectedness and Binary Branching. Dordrecht, Holland: Foris Publications.
- Keenan, Edward L. 1974. 'The functional principle: Generalizing the notion of "subject of"'. In Michael W. LaGaly, Robert A. Fox and Anthony Bruck (eds.), Papers from the 10th Regional Meeting of the Chicago Linguistics Society, 298-309.

- Klavans, Judith L. 1980. 'Some problems in a theory of clitics'. Ph.D. dissertation, University College, London. (distributed by the Indiana University Linguistics Club, Bloomington, 1982.)
- _____. 1985. 'The independence of syntax and phonology in cliticization. Language, 61, 95-120.
- Langendoen, D. Terence. 1976. 'On the weak generative capacity of infinite grammars. CUNYForum, 1, City University of New York, 13-24.
- Lapointe, Steven G. 1979. 'A theory of grammatical agreement.' PhD Dissertation, University of Massachusetts at Amherst.
- Lasnik, Howard and Robert Fiengo. 1974. 'Complement object deletion'. Linguistic Inquiry, 5, 535-571.
- Lee, Ik-Hwan. 1985. 'Multiple WH-questions in GPSG'. Papers from the Twenty-First Regional Meeting, Chicago Linguistic Society, Chicago: IL. 236-255.
- Longobardi, Giuseppe. 1985. Connectedness, Scope, and C-command. Linguistic Inquiry vol. 16.2, 163-192.
- Maxfield, Thomas L. 1990. 'Children know about WHAT questions?'. Paper presented at UMASS Workshop on the Acquisition of WH Constructions.
- Moravcsik, Edith A. 1978. 'Agreement.' Universals of Human Language Volume 4 Syntax Greenberg (ed.), 331-374.
- Peters, Stanley and R. Ritchie. 1973. 'On the generative power of transformational grammars.' Information Sciences. 6:49-83.
- Pollard, Carl & Ivan A. Sag. 1987. Information-Based Syntax and Semantics. Vol. 1 Fundamentals. CSLI Lecture Notes, No.13, Stanford, CA: Center for the Study of Language and Information. Distributed by The Chicago University Press.

- Pollard, Carl J. and Ivan A. Sag 1983. Reflexives and Reciprocals in English: an alternative to the binding theory. In Michael Barlow, Daniel Flickinger, and Michael Wescoat (eds.), Proceedings of the Second West Coast Conference on Formal Linguistics. Stanford: Stanford Linguistics Department, 189-203.
- Postal, Paul M. and Geoffrey K. Pullum 1988. 'Expletive noun phrases in subcategorized positions.' Linguistic Inquiry, 19(4), 635-670.
- Pullum, Geoffrey K. 1985a. 'Assuming some version of X-Bar theory'. Papers from the Twenty-First Regional Meeting, Chicago Linguistic Society, Chicago IL. 323-353.
- _____. 1985b. 'How complex could an agreement system be?' Eastern States Conference on Linguistics, 1, 79-103.
- Rizzi, Luigi. 1982. Issues in Italian Syntax. Dordrecht, Holland: Foris Publications.
- _____. 1987. 'Relativized Minimality'. ms.
- Roeper, Thomas. 1981. 'In pursuit of a deductive model of language acquisition. In C.L. Baker and J.J. McCarthy, eds., The logical problem of language acquisition. Cambridge, MA: MIT press.
- Sag, I. A. 1983. 'On parasitic gaps'. Linguistics and Philosophy, 6, 35-45.
- _____, Gerald Gazdar, Thomas Wasow, & Steven Weisler. 1985. 'Coordination and how to distinguish categories'. Natural Language and Linguistic Theory. 3, 117-171.
- Schachter, Paul. 1984. 'Auxiliary reduction: An argument for GPSG'. Linguistic Inquiry. 15, 514-523.
- Sells, Peter. 1985. Lectures on Contemporary Syntactic Theories: An Introduction to Government-Binding Theory, Generalized Phrase Structure Grammar, and Lexical-Functional Grammar. CSLI Lecture Notes, No.3, Stanford, CA: Center for the Study of Language and Information. Distributed by The Chicago University Press.

- Shieber, S. 1985. 'Evidence against the context-freeness of natural language'. Linguistics and Philosophy, 8, 333-343.
- _____. 1986. 'GPSG: A simple reconstruction'. Technical Note 384. SRI International, Menlo Park, CA.
- Stowell, Timothy. 1981. 'Origins of phrase structure'. Doctoral Dissertation, MIT.
- Stucky, Susan. 1981. 'Word order variation in Makua: a phrase structure grammar analysis'. PhD. dissertation, University of Illinois at Urbana-Champaign.
- Travis, Lisa. 1984. 'Parameters and effects of word order variation.' Ph.D. dissertation, MIT.
- Ultan, Russell. 1978. 'Interrogative systems'. In Universals of Human Language, Volume 4, Syntax, Greenberg (ed.),
- Uszkoreit, Hans and Stanley Peters. 1985. 'On some formal properties of metarules.' Report No. CSLI-85-43. Stanford, CA.
- Valian, Virginia. 1989. 'Logical and Psychological constraints on the acquisition of syntax. Paper presented at UMASS workshop on Language Processing and Acquisition.
- van Kemenade, Ans. 1987. Syntactic Case and Morphological Case in the History of English. Foris, Dordrecht, Holland.
- Wexler, Kenneth & Peter Culicover. 1980. Formal principles of language acquisition. Cambridge, MA: MIT Press.