

Automodular Grammar

Jerry Sadock

University of Arizona, January 18, 2008

Lecture 1

1.0 Basic Principles

An automodular grammar consists of several independent generative devices, each capable of characterizing an infinite number of structures and their associated terminal strings. The several dimensions are informationally distinct and each operates with its own categories and its own principles of combination. I will gradually introduce the following components:

- Syntax, which characterizes the structure of an expression in terms of syntactic categories: noun phrases, verb phrases, prepositional phrases, and so on, leading eventually to clauses.
- Function-argument (F/A) structure, which specifies the organization of an expression according to combinatoric semantic categories: functions such as predicates, relations, operators, arguments, and so forth. The maximal expressions in this dimension are propositions.
- Role structure (RS), where the categories are event types, and participant roles such as proto-agents and proto-patients that together form complete events.
- The linear order component (LOC) that characterizes strings of elements defined in terms of categories established in one or more of the other components.
- Morphology, whose categories are stems of various morphological kinds, words, and functions that apply to morphological categories and produce morphological categories. The morphological word is the maximal structure defined by this component.
- The morphophonological (m-phon) component determines the phonological content of the output of a morphological function as a function of the morphophonological content of the stem, a string of segments, and a morphophonological operation on that alters the string.

The analysis of a natural language expression will consist of a sextuple of expressions each of which meets the requirements of one of the self-standing components. These fully independent representations must be matched to each other in order to assure that they are allowable autonomous-modular analyses of one and the same expression. This is the job of the interface mechanism. One function of the interface is to insure that the same lexical content is found in all of the autonomous representations and that is partially the job of the lexicon that provides the value of lexical items in the various modules.

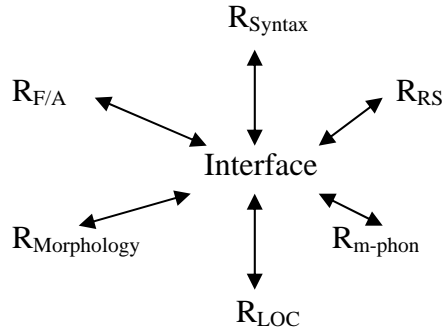


Figure 1. The Automodular Model

1.1 Function-Argument Structure:

I begin with the commonplace type-theoretic idea that there are two basic semantic categories, roughly entity expressions and truth valued expressions, an idea which I crib shamelessly from work in the general framework of Montague Grammar, Partee and Hendriks 1997, for example. For these two basic categories I will employ the terms argument (Arg) and proposition (Prop), since they are more familiar to linguists and are not so closely tied to particular versions of semantic theory as are their designations in various logical traditions. From these two basic categories, an infinite number of additional categories can be recursively defined, each a function from existing category types to new category types:

(1) Categories in F/A Structure:

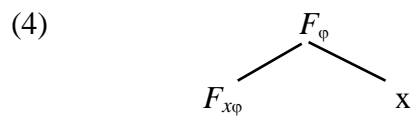
- a. Basic categories: Arg(ument) and Prop(osition)
- b. If A and B are categories, then there is a category C that is a function from categories to categories such that $C(A) = B$.

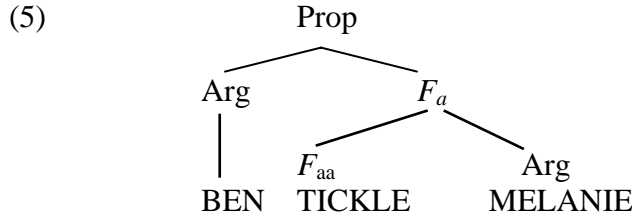
(2) Rule of Functional Application

$$F_{x\varphi}(x) = F_{\varphi}$$

Such a rule can be taken to be an context free phrase structure rule like (3) that describes a tree fragment of the form (4), where order is irrelevant:

(3) $F_{\varphi} \rightarrow F_{x\varphi}, x$

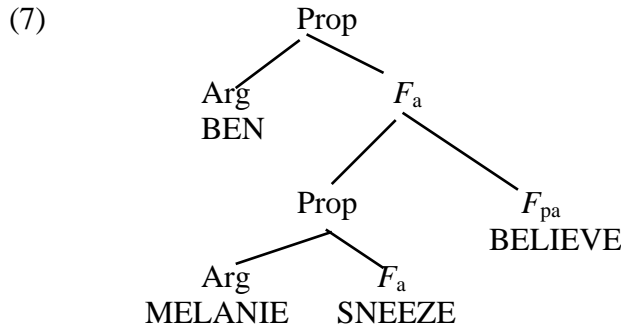




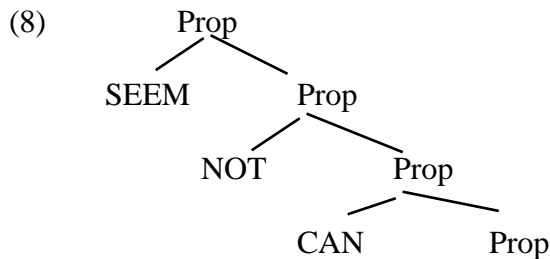
(6) Some functors in English

- a. F_a : intransitive predicates, e.g., *sneeze*, *cat*, *cute*.
- b. F_{aa} : relations, e.g., *steal*, *fiancee*, *fond*.
- c. F_p : one-place operators, e.g., *seem*, *longshot*, *likely*.
- d. F_{pa} : transitive operators, e.g., *believe*, *idea*, *positive*.

Category (6) d. will enter into F/A trees like the following, which represents the structural semantic relations of the clause *Ben believes that Melanie sneezes*:



Syntactic and functional scope usually correspond, but there are cases where scopal facts cannot be conveniently mapped directly from the surface features of an expression. To take just one example that has received some mention in the literature, consider sentences that contain the sequence ... *can't seem to* Langendoen (1970). There is no reason to take the syntactic form of this sentence to be anything other than the well-formed syntactic structure that it appears to be, with *can* and *not* higher in the syntactic tree than *seem*. The F/A structure, however, will have these hierarchical relations reversed; SEEM (the F/A structure counterpart of *seem*) is higher in the functional tree than CAN and NOT:



1.2 Syntax

Because complexity in an automodular grammar arises from the interaction of autonomous components, each of these components can be kept relatively simple. Once semantic, morphological, and other non-syntactic considerations are factored out, it proves adequate to assume a very straightforward, context-free phrase structure grammar as a device for specifying the syntactic structure of natural languages.

Notation and delicacy of feature structure aside, much of which will be made precise in the subsequent chapters, the syntactic component of English might contain the following rules and rule schemata, as well as a few others.

(9) Syntax:

- a. $S \rightarrow NP, VP$
- b. $V' \rightarrow V, (XP, (YP))$
- c. $PP \rightarrow P, NP$
- d. $NP \rightarrow DetP, N'$
- e. $N' \rightarrow N, (XP, (YP))$
- e. $S' \rightarrow Comp, S$
- f. $AP \rightarrow (ADVP), A'$
- g. $A' \rightarrow A, (XP, (YP))$

2.0 The Interface

2.1 Lexical Items

Lexical items in automodular grammar are bundles of instructions to the various component grammars, specifying what the value of a particular lexeme is in each module. With regard to the two components that were just sketched out, a lexical entry will make precise how the lexical item combines syntactically and how it combines in terms of F/A structure. An ordinary intransitive verb, for example, will be able to constitute a syntactic VP all by itself and will be a semantic functor that directly combines with an argument to form a proposition, that is, a member of the category F_a . A standard sort of transitive verb will combine with an NP to form a VP in the syntax and will instantiate the F/A category F_{aa} , combining with an argument to form a predicate.

- (10) *sneeze*:
syntax: V in [VP__]
F/A: F_a
- (11) *take*:
syntax: V in [VP __ NP]
F/A: F_{aa}

2.2 Lexical Correspondence

(12) Intermodular Lexical Correspondence Principle:

If the value of a lexical item occurs n times in dimension D_j , then the dimensionally appropriate values of that lexical item must be present n times in every other dimension of analysis.

2.3 Categorical Correspondence

In general a. and b. hold. Furthermore, if a language has a syntactic category VP, then c. is also generally valid. If a language has no VP, then obviously, c. does not hold.

(13)	Syntax		F/A
a.	S	\Leftrightarrow	Prop
b.	NP	\Leftrightarrow	Arg
c.	VP	\Leftrightarrow	F_a

But these are only *default* correspondences and there are deviations from them of many kinds. For one thing, there are lexical items that are not represented in all dimensions. English has quite a few including functionally empty *be*, pleonastic *it*, and the *do* of *do*-support.

(14) *be*
syntax: V in [__ NP]
F/A: nil

(15) *it* (pleonastic)
syntax: NP
F/A: nil
morphophonology: /It/

(16) *do* (auxiliary):
syntax: V[TNS] in [_{VP} __, VP[INFIN]]
F/A: nil

There are also idiom chunks of various kinds, e.g.,

(17) *look into* “investigate”
syntax: V in [__ PP[into]]
F/A: F_{aa}

2.4 Geometrical correspondence

The second sort of default correspondence that is obvious when one compares syntactic and combinatoric semantic representations has to do with the relative structural positions of the corresponding elements in the two dimensions that have been considered to this point. The general principle is that hierarchical relations among elements in one dimension should be reflected in the hierarchical relations among the corresponding elements in the other dimensions. (This and the correlation of categories discussed above were both included under the Generalized Interface Constraint of Sadock and Schiller 1993.) The structures I which lexical items in (10) and (11) figure are good examples of geometrical correspondence.

(17) Geometrical correspondence conditions

Let A and B be nodes in dimension D1 and A' and B' be corresponding nodes in dimension D2.

Then:

a. Conservation of Dominance:

If A dominates B, A' should dominate B'

b. Conservation of C-command:

If A c-commands B, A' should c-command B'

2.5 “Raising” to Subject

As with categorical correspondence, geometrical correspondence are just defaults. There are many varieties of deviation from it that lead to complications in individual components that are not autonomous, but not here. As a first example, consider those predicators that function as intransitive operators (that is to say, propositional modifiers of the F/A category F_p) and occur with syntactic complements other than subordinate clauses, including VPs and adjective phrases. These include verbs like *seem*, *appear*, and *happen*, adjectives like *likely*, *certain*, and *sure*, nominals like *a certainty* and *a good bet*. The verbs *seem* and *appear* occur with either a VP complement: *appears to be absent*, an adjective complement: *looks awfully sick*, or (chiefly in British English) a nominal: *seems a nice enough bloke*. None of the adjectives or nominals can take an adjective phrase or a nominal as a complement, but most do occur with VPs:

(18) a. is certain to be absent

*is certain very sick

*is certain a nice enough bloke

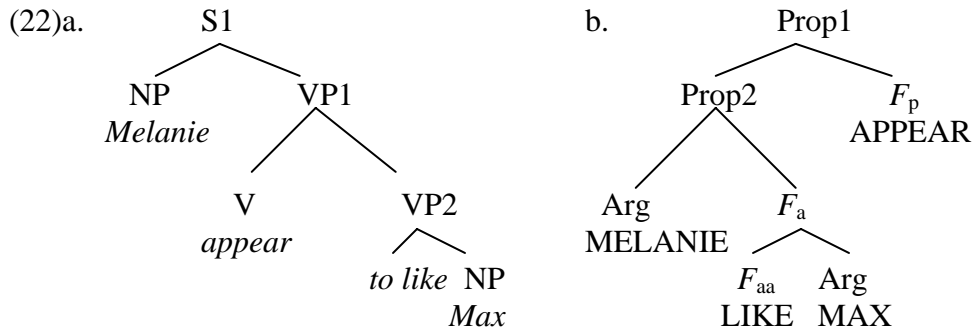
(19) *seem*, *appear*, *look*:

syntax: V in [__ VP[to]] or [__ AP] (%or [__ NP])

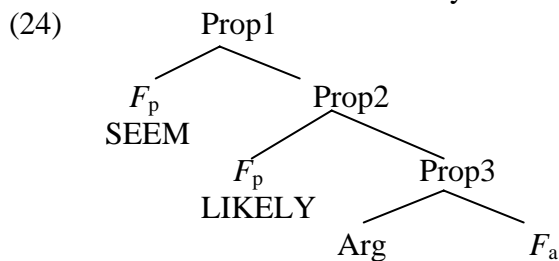
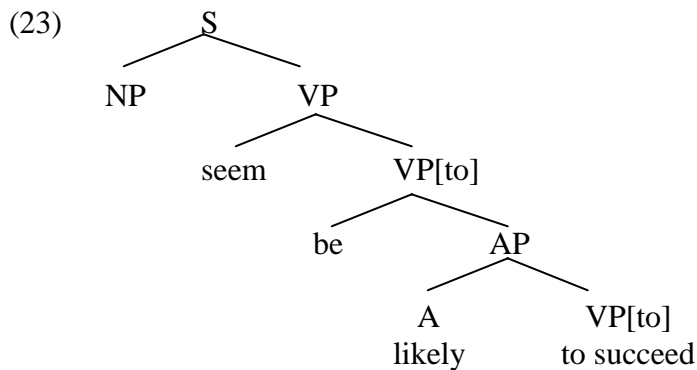
F/A: F_p

- (20) *likely, certain, sure*:
 syntax: A in [__ VP[to]] or [__]
 F/A: F_p
- (21) *certainty, good bet*:
 syntax: N in [__ VP[to]] or [__]
 F/A: F_p

Such lexical entries require geometrical mismatches between syntax and F/A structure. Traditionally, this is called raising. In this framework, nothing more needs to be said to get the desired results.



What has been said so far also correctly predicts some of the behavior of nested complements, such as the fact that while **That Rachel succeeded seems* is ungrammatical, *that Rachel succeeded seems (to be) likely* is not. Note also that the “long distance” raising or cyclic raising is an automatic consequence of what has been said so far.



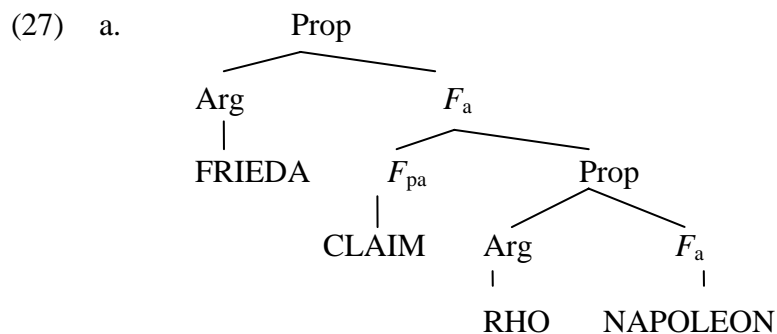
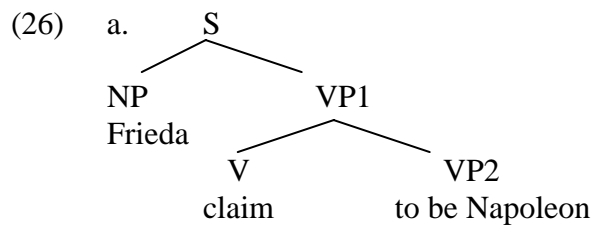
2.6 Control Predicates

Subject control predicates have exactly the same syntax as raising to subject predicates, but different F/A structure, of course. A verb like *claim* can take a VP complement or a clausal complement, but in either case it will be a semantic function from propositions to predicates.

- (25) *claim*:
 syntax: V in [__ S'] or [__ VP[to]]
 F/A: F_{pa}

When such predicates occur with infinitival complements, there is a problem: There are not enough NPs in the syntactic structure whose meaning can fill out all of the argument positions in F/A structure and therefore the F/A structure will be incomplete. We can't stick in another FRIEDA or we would violate the Lexical Correspondence Principle. What is needed is a lexical item with the meaning of an argument but no syntactic value whatsoever, the inverse of pleonastic *it*. I call this element "RHO".

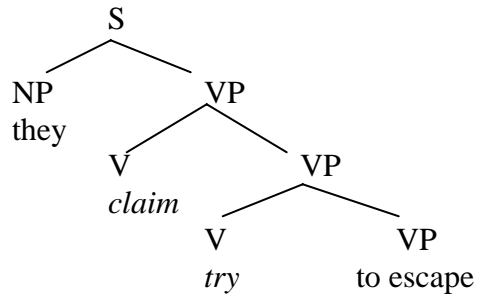
- (25) RHO:
 syntax: nil
 semantics: Arg



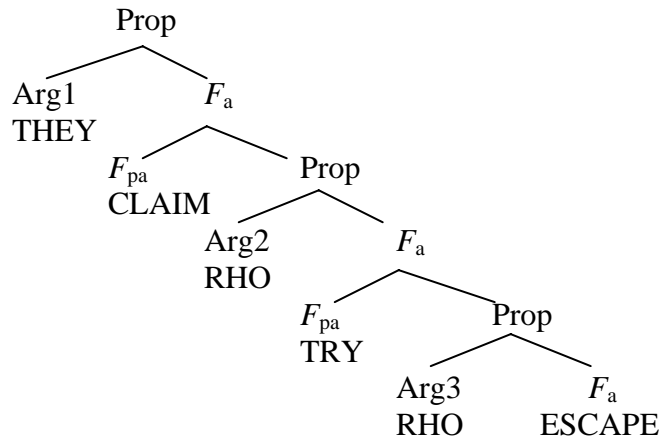
As an argument, RHO must have a meaning, and it does: RHO is a syntactically empty item with the meaning of a pronoun just as pleonastic *it* is a semantically empty item with the value on an NP.

Consider examples with nested subject control verbs like *They claimed to have tried to escape*. The syntactic and F/A structures of this example are (28a) and (28b) with some details suppressed for clarity: The reference of the lower RHO will be the same as that of the higher RHO which will be the same as the reference of “they”.

(28) a.



b.

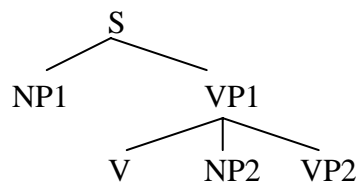


2. Raising to object

A number of verbs that otherwise occur with clausal complements and/or VP complements can also occur with both an NP and a VP as in *Nobody believes those politicians to be trustworthy*. Such predicates are necessarily verbs in English, including those listed in (29), because nouns and adjectives are never subcategorized for an NP object. The items in (29) were treated as triggering the raising of the complement subject to their own object position (Postal 1974) or as assigning case in an exceptional way across a clause boundary (Chomsky 1981).

(29) *believe, expect, find, suppose, think, presume, assume, consider, understand*

(30)



The syntax of sentences like *Sheila found echidna to be delicious* would seem to be identical to the syntax of *Sheila told Sidney to be efficient*. Sentences are understood in much the same way as sentences with clausal complements when the verb is one of those in (29). The NP object is not a semantic argument of the matrix verb but rather is understood only as the semantic subject of the proposition that corresponds to VP2. In other words, the lexical specifications of such verbs in syntax and F/A structure will be of the form (30).

- (30) *believe/expect/find ...*:
syntax: V in [__ NP, VP[to]]
F/A: F_{pa}

3.0 Pop quiz:

What is the lexical representation of verbs of object control?
How does the following contrast between raising and control predicates follow from this?

*Bernanke believed/*persuaded the shit to have hit the fan.*