University of Massachusetts Occasional Papers in Linguistics

Volume 7 *University of Massachusetts Occasional Papers in Linguistics -- Volume 6: Markedness and Learnability*

Article 3

1981

Computational Analogues of Constraints on Grammars: A Model for the Acquisition of Syntactic Knowledge

Robert Berwick

Recommended Citation

Berwick, Robert (1981) "Computational Analogues of Constraints on Grammars: A Model for the Acquisition of Syntactic Knowledge," *University of Massachusetts Occasional Papers in Linguistics*: Vol. 7 , Article 3.
Available at: https://scholarworks.umass.edu/umop/vol7/iss2/3

COMPUTATIONAL ANALOGUES OF CONSTRAINTS ON GRAMMARS: A MODEL FOR THE ACQUISITION OF SYNTACTIC KNOWLEDGE

Robert Berwick

1.0 Introduction: Computational Constraints and Language Acquisition

One of the important goals of modern linguistic theory is to explain how it is that children can acquire their first language. This is a deep and significant research problem: on many accounts the "evidence" that children receive to learn language is quite impoverished, and reinforcement by adults haphazard; yet the process of first language acquisition seems relatively easy and strikingly uniform. Children with enormously disparate sensory environments — normal children, deaf children of normal parents, deaf children of deaf parents, normal children of deaf parents, blind children — all seem, at least initially, to learn the same parts of what linguists call a grammar [Newport, Gleitman, and Gleitman, 1977]. Such robust performance in the midst of raging environmental variation poses a severe challenge for any theory of language acquisition.

Modern linguistics has countered with a research strategy that reconciles this apparent paradox by characterizing so narrowly the class of possible human grammars that the language learner's burden is eased, perhaps trivialized. In Chomsky's metaphor, hypothesizable grammars should be "sufficiently scattered" from one another so that children can easily select the one correctly corresponding to the language of their caretakers (Chosmky [1965]). Such restrictions aid the learner because they rule out countless faulty hypotheses about which possible grammar might cover the language at hand. For example, suppose there was but a single human grammar. Such a situation would be optimal from the standpoint of language acquisition; no matter how complex, the grammar could be built-in, and no learning required. More realistically, current theories of transformational grammar are usually motivated and evaluated with this learnability principle in mind; the possible rules of grammar (and thus the possible human grammars) are restricted as much as possible to just a few actions plus universal constraints on their application. The business of linguistics for the past several years has been to uncover these universal principles, from the "data" of grammaticality judgments, and so advance, indirectly, an explanation for language learnability.

This metaphor of a child searching through a restricted hypothesis space of grammars has proved to be an enormously fruitful one for modern linguistic theory, drawing us closer toward the goal of characterizing what is distinctively human about the complex behavior that is called "language". However, the notion of "generative grammar" as developed over the past few decades deliberately abstracts away from the actual time course of language processing. This idealization has enabled researchers to focus on questions central to the theory of grammar while ignoring theoretically distracting questions of human processing mechanisms.[1] But it has left us with the puzzle of how, exactly, constraints on grammars are to be incorporated into a model that accounts for the apparent use of those constraints during the processing of individual sentences, and more broadly, during the course of acquisition itself. The job remains for those who are interested in developing compurer systems that can acquire new syntactic knowledge "in the manner of a child" to provide an explicit process model that does the double-duty of (1) accounting for language processing via a system closely tailored after the structures provided by generative grammar, and (2) accounting for the unfolding of those processing abilities over time.

The first problem then is to supply a process model of language use that pays careful attention to the structures and rules of a generative theory while at the same time faithfully mirroring "natural" parsing behavior. One such model already exists; Marcus [1977, 1980] has developed an efficient, left-to-right parser for English (Parsifal) that closely mimics the "operating principles" of current transformational theory. In the classical generative account, "syntactic knowledge" is characterized simply as a way to generate a list that pairs surface strings (sentences) with structural descriptions of those strings (labelled bracketings, or parse trees). Hence the term "generative grammar." A parser (such as Parsifal) adds an inherent directionality to this account by providing an effective procedure for passing from any given surface string to its proper labelled bracketing (if the string is syntactically well-formed) or some error state (if it is not).

As with any algorithmic construct, the Parsifal procedure itself can be thought of as a simple two-part machine: an interpreter (a finite-state control, a fixed collection of data structures and a working storage that might or might not be infinite in extent) and the grammar rules (machine instructions) that the interpreter executes. It is the grammar rules that do the actual work of the parse, unwinding the mapping between surface form and structural description; the interpreter serves as bookkeeper and control. The parser's storehouse of rules is thus a repository of syntactic knowledge as it functions in language use, in this case, parsing.

With Parsifal in hand, the obvious next step is to model the acquisition of syntactic knowledge as the acquisition of a series of parsers of increasing sophistication. This is the focus of the research described in this report; the approach is sketched out in Figure 1 below. First, we fix some initial state of knowledge, corresponding to an initial set of parsing abilities and a knowledge of what counts as a valid rule of parsing. This state is labelled as "$P_{initial}$" in Figure 1. Then, an acquisition procedure constructs a sequence of new parsers, P_i, incrementally adding to or modifying the knowledge base of parsing rules in response to a set of input data. At each step we are guaranteed the existence of a machine P_i that can process a subset of the language at hand in accordance with the more abstract rules of grammar uncovered by linguistic theory; yet at the same time, the sequence of parsers mimic an (idealized) acquisition process by utilizing these same constraints to approximate ever more closely a "mature" knowledge of syntax.

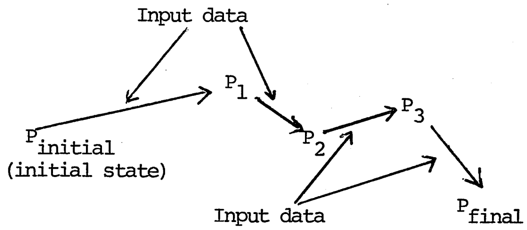


Figure 1. Acquisition as modelled by a sequence of parsers

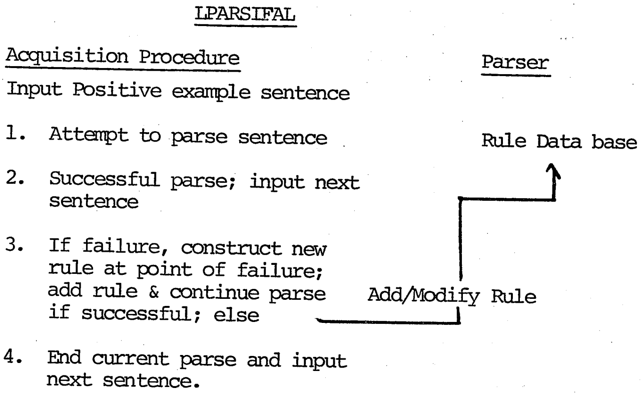
This model owes much to the picture of acquisition drawn in Chomsky's Aspects of a Theory of Syntax. It is easy to see what the current proposal and the Aspects model have in common. Both presume an ability to represent the speech signal as a highly structured set of input tokens -- in the case of the acquisition procedure, as the segmented individual words input to the parser. Both propose to represent those signals as highly abstract, labelled tree structures. Finally, both approaches presume an initial delimitation of the class of possible hypotheses about those structures -- in the case of the Aspects model, initial constraints on grammars, and for the acquisition procedure, a pre-supposed initial parser (perhaps a bare interpreter), and a knowledge of what counts as a valid rule of parsing.

The two models part ways only in the manner of selecting the proper grammar (parser). The Aspects procedure considers all possible candidate grammars and data together, and, using a selection function (the evaluation measure, or evaluation metric) picks out the "best" grammar (the "correct" G_i) at one fell swoop. It is thus instantaneous; developmental sequencing has been abstracted away (or, perhaps, factored into the work that the evaluation metric must do). As Chomsky notes,

> What I am describing is an idealization in which only the moment of the correct grammar is considered ... it might very well be true that a series of successively more detailed and highly structured schemata (corresponding to maturational stages, but perhaps in part themselves determined in form by earlier steps of language acquisition) [1965, footnote 19 to Chapter One, page 202]

The research reported on here might best be thought of as an attempt to relax the idealization of instantaneous acquisition, pursuing the consequences of adopting the developmental approach. There are two possible rewards. First, the developmental approach has (as Chomsky indicates) an obvious psychological appeal; instantaneous acquisition is certainly false to fact; like most other biological capabilities, syntactic knowledge (superficially) obeys a regular maturational course more characteristic of unfolding embryological development. While the microstructure of any single child's language ontogenesis may be expected to differ from any other child's in the same language community, the overall "developmental envelope" for both will be the same. The model described in this report aims at an explicit account of this fact. Second, again as Chomsky points out, by incorporating the actual time course of events into an acquisition model we may hope to gain further insight into what distinguishes the class of natural languages from arbitrary sign-to-symbol mapping systems.

For the developmental approach to succeed we must spell out in detail exactly what is meant by initial parser, input data, and acquisition procedure. These three elements, comprising the complete acquisition model, will be dubbed L parsifal. As its initial state, L parsifal employs a streamlined, rule-less version of Parsifal -- roughly, the bare interpreter and its data structures. As input data, it takes just grammatical sentences (so called positive evidence) and a rudimentary initial ability to characterize words as objects, actions, or unknown. (The ability to label lexical items with category features has a developmental program of its own, interacting with unfolding syntactic abilities; for further discussion, see Berwick [1980] for details.) The developing parser attempts to analyze each example sentence aupplied by the external world;[2] the acquisition of new rules is prompted by failures during the parsing process. Each time that a failure is detected on-the-fly in the left-to-right processing of a sentence and successfully resolved, the system adds a single new parsing rule to its knowledge base or modifies (perhaps reinforces) an old rule. If the system cannot resolve a local parsing failure, it simply gives up on the analysis of the current sentence and tries to parse the next. Three sub-systems of syntactic knowledge are acquired: context-free base rules (corresponding to the basic constituent order of English); (2) lexical insertion grammar rules; and (3) pattern-action grammar rules (corresponding to particular local transformations or long-distance movement rules). This startingly simple procedure has proved to be remarkably successful; the currently-implemented version of the model can acquire from just simple positive example sentences the parsing rules sufficient to handle a large core subset of English sentences. Syntactic phenomena as diverse as Subject-auxiliary verb inversion, simple passives and constituent movements, as well as the "canonical" ("deep structure") ordering of phrases in an English sentence can all be acquired by the procedure.



Input Positive example sentence

1. Attempt to parse sentence
2. Successful parse; input next sentence
3. If failure, construct new rule at point of failure; add rule & continue parse if successful; else
4. End current parse and input next sentence.

Figure 2. Outline of the acquisition procedure

Note that according to this model, new knowledge is acquired by modifying only the rule data base of the parser; the interpreter (working storage and control structure) and knowledge of the valid forms for rules remain fixed. This may seem somewhat asymmetric, but appears to be a legitimate working assumption: the interpreter and the knowledge of constraints on rules are presumed to be universal across all languages, hence in fact fixed. There are, however, several details of the interpreter that might in fact change over time. One can imagine a model where, for example, the working storage of the interpreter is expanded over time (perhaps corresponding to a general increase in memory or attention span) or the inventory of rule features altered (again in response to the concurrent development of other cognitive faculties); an explicit proposal to this effect is made in Berwick [1980]. But no extensive overhauls in the control structure of the interpreter seem to be required, in part because the basic execution loop of the "mature" interpreter is already so simple that there is really very little to learn.[3]

In summary, the proposed model for acquisition is quite simple. Knowledge of syntax (a grammar) is represented as a parser; development of that knowledge by a sequence of parsers. The acquisition process itself is driven by the current parser's attempts to interpret positive example sentences prompting changes to its data base of parsing rules.

2.0 Evaluating the Model

As Pinker [1979] has noted, any model of the acquisition of syntactic knowledge must pass a stiff set of evaluation criteria. Plainly it must actually acquire syntactic knowledge; the model must be powerful and sophisticated enough in either its initial knowledge of syntax, its acquisition procedure, or both, to learn new rules of the sort that people apparently learn. In short the procedure must possess a rich internal structure of its own. For Lparsifal, this amounts to its initial possession of a full Parsifal interpreter, as well as a detailed knowledge of the proper format for grammar rules and context-free base rules. On the other hand, this "given" structure cannot be too rich; providing Lparsifal with the instantiated base rules and grammar rules of English would simply beg the question of acquisition, while at the same time leaving unexplained the fact that children from other language environments learn other rules. Any acquisition procedure must therefore tread a thin line between an approach that is constrained enough to account for acquisition (what Pinker dubs "learnability"), yet not so constrained as to rule out observed variation in human syntactic knowledge (Pinker's "equipotentiality" condition.)

The dial criteria of "learnability" and "equipotentiality" reflect a model's ability to characterize in an abstract, generative sense the class of possible human grammars. In addition, Pinker has isolated a series of criteria that focus upon a model's plausibility as a reflection of human processing demands. If possible, the output of the acquisition procedure should be a characterization of the process of passing from specific strings of words to their analyzed forms (or vice-versa). In addition, the acquisition procedure should assume an initial state, input data, and available computational resources (time and space) that are least broadly compatible with those that can be reasonably assumed available to (developing) children. The acquisition procedure should _not_ take an infinite amount of time; should use only input data (examples) known to be accessible to children; should approximate the ontogenetic course of acquisition (the order and types of rules acquired and corresponding mistakes should correspond to observation); and should not demand "superhuman" perceptual or cognitive abilities. Given these criteria, let us review the success of the Lparsifal model.

2.1 Learnability and Equipotentiality

How well does the procedure fare in actually acquiring parsing rules? Starting with _no_ grammar rules, the currently implemented procedure acquires from _positive_ example sentences many of the grammar rules in a "core grammar" of English originally hand-written by Marcus. The currently acquired rules are sufficient to parse simple declaratives, much of the English auxiliary system including suxiliary verb inversion, simple passives, _simple wh_-questions (e.g., _Who did John kiss?_), imperatives, and negative _adverbial_ preposing. Carrying acquisition one step further, by starting with a relatively restricted set of context-free base rule schemas -- the "X-bar" system of Chomsky [1970] and Jackendoff [1977] --

the procedure can also induce many of the lexical insertion rules and some of the phrase structure rules for English, including recursive phrase structure rules. Acquired base rules include those for Noun Phrases, Verb Phrases, Prepositional Phrases, some Complements to verbs, and a substantial part of the English auxiliary verb system. In short, the procedure can acquire a knowledge sufficient to parse an infinite number of sentences on the basis of very simple triggering evidence alone. (For two quick examples of the procedure in action, see Section 5.)

This short list of acquired rules is not, of course, even close to a "complete" knowledge of English syntax. Many rules lie beyond the current procedure's reach. Some of these gaps in acquisition are the result of corresponding gaps in computational machinery, while others probably signal a deeper deficit in the grammar-as-parser approach. As an example of an easily-remedied deficit of machinery, consider the class of movement rules. At present, Lparsifal has only a single device to handle all constituent movements. Lacking a distinguished facility to keep track of wh-movements, Lparsifal cannot acquire the rules where these movements interact with Noun Phrase movements. (Parsifal employed dual mechanisms to distinguish Noun Phrase and wh-movements.) Current experiments with the system include adding the wh facility back into the domain of acquisition. Rules for the more complex complement structures of verbs simply remain to be tackled; those for conjunction cannot be until some rules for conjunction are developed by hand for the original Parsifal.

More generally however, the present model cannot capture all "knowledge of language" in the sense intended by generative grammarians. For example, since the weakest form of the acquisition procedure does not employ backup, it cannot re-analyze certain sentences as people appear to do, and so deduce that they are grammatically well-formed.[4] In part this incompleteness simply reflects our partial understanding of how our knowledge of language interacts with parsing behavior.

Even though much work remains, the Lparsifal model can clearly in principle strike the proper balance between descriptive and explanatory adequacy. Since its representational armamentarium includes much of the full expressive power of current theories of generative grammar -- recursive phrase structure rules supplemented by the processing counterparts of many transformational operations -- it may suffice to describe a broad range of human parsing abilities. And since it also incorporates the constraints of those theories -- the X-bar system of phrase structure templates and tight restrictions on form and functioning of rules -- it may also be sufficiently narrow as to make acquisition possible.

2.2 Psychological Fidelity

The Lparsifal procedure has also been deliberately designed so as to meet (more or less) the remaining criteria of model adequacy. By definition, the characterization of syntactic knowledge attained by the acquisition procedure is a parser, and can be considered a model of the process of mapping sentence strings to their structural descriptions. By fiat, the acquisition procedure uses only positive (grammatical) examples as evidence on which to base the construction of its new rules. This reliance on positive-only examples can be justified on empirical and methodological grounds. Although the final psycholinguistic evidence is not yet in, children do not appear to receive negative evidence as a basis for the induction of syntactic rules. That is, they do not receive direct reinforcement for what is not a syntactically well-formed sentence — so called negative evidence.[5] If syntactic acquisition can proceed using just positive examples, then it would seem completely unnecessary to move to an enrichment of the input data that is as yet unsupported by psycholinguistic evidence. There is yet another reason for rejecting negative examples as a source of evidence; from formal results first established by Gold [1967], it is known that by pairing positive and negative example strings with the appropriate labels "grammatical" and "ungrammatical" one can learn "almost any" language. Thus, enriching the input to admit negative evidence broadens the class of "possibly learnable languages" too far, admitting systems of rules that cannot possibly be characterized as human.

It is also worth noting that the procedure does provide a theory of acquisition — that is, a way in which examples can drive the acquisition process. This in itself is an important accomplishment. As recently as 1979 Marshall could still write:

> There is, however, a very general problem with practically all studies of language development, whether investigated from the standpoint of rule acquisition, strategy change, or elaboration of mechanism. The problem arises both for accounts that postulate "stages" of development (i.e. a finite number of qualitatively distinct levels of organization through which the organism passes en route from molecule to maturity) and for accounts that view development as a continuous function of simple accumulation. The difficulty is this: No one has seriously attempted to specify a mechanism that drives language acquisition through its stages or along its continuous function. Or more succinctly; there is no known learning theory for language.
> [1979 page 443].

The current model remedies at least part of this problem: the parser's attempts to interpret sentences provides a specific driving mechanism for acquisition, a theory of acquisition.

What about the restrictions on time for acquisition or on developmental and cognitive capacity? Here, although sharp statements of convergence time and fidelity to human language ontogenesis are not yet forthcoming, one can show that Lparsifal meets weak upper bounds on these conditions. For instance, any mature parser consists of a fixed Parsifal interpreter and a varying set of grammar and base rules. But the base rules are finite in number, since they can be expressed by a finite set of strings each of finite length, drawn from the fixed and finite theoretical vocabulary of the X-bar theory. Likewise there are a finite number of grammar rules, for the patterns and actions that form all such rules must by convention be expressible as strings finite in length, using a finite vocabulary of symbols. As a result, there are but a finite number of possible Lparsifal parsers (the crossproduct of the number of base and grammar rules). If this set of parsers is descriptively adequate, then this fact by itself ensures that acquisition "in the limit" can be avoided; a simple-minded acquisition procedure that guesses parsers by enumerating all of them (changing a single rule at a time) could arrive at a "correct" parser without taking unbounded amounts of time.[6]

Further, the procedure is deliberately designed to be able to glean more information from simple example sentences (of limited embedding) than more complex examples. This is accomplished in two ways. First, the acquisition procedure cannot be recursively invoked; if, during the process of acquiring a new grammar or base rule the procedure discovers that it must acquire yet another new rule, the attempt to acquire the first new rule (as well as the second) is abandoned. This ban on recursion has the beneficial effect of keeping the procedure's new rules "close" to those it already knows, imposing a certain incremental and conservative character on its developmental history. Second, the grammar rule patterns themselves are restri-ted under the convention of Marcus' Parsifal to examine only local context in order to decide what to do next. As a result, so-called "garden-path" sentences — examples where disambiguation requires a more global analysis of the sentence, as in, <u>The horse raced past the barn fell</u> — cannot provide the basis for new rules of parsing.[7] Together these restrictions simply mean that if an input datum is too complex for the acquisition program to handle at its current stage of syntactic knowledge, it simply parses what it can, and ignores the rest. The outcome is that the first rules to be acquired handle simple, few-word sentences and expand the basic phrase structure for English; later rules deal with more sophisticated phrase structure, alterations of canonical word order, and embedded sentences. Although this is clearly part of the desired result — current parsing abilities suggest new rules of parsing — it is a picture without much detail. An important area for future research will be to investigate more thoroughly the pattern of over- and under-generalization implicit in the model, comparing them to observed patterns of human language ontogenesis and error production.

3.0 Computational analogues of constraints on grammars

Perhaps most importantly however, the initial success of the Lparsifal model underwrites the methodological approach of the generative grammarians. Their assumption that languages are structured in order that they may be easily <u>acquired</u> receives strong support from the results of this research. Simply put, what makes the rules easy to acquire is that the choices that must be made are few; the acquisition program is limited to constructing rules only of a certain kind, built from but a handful of possible actions. The success of this approach thus also confirms what is fast becoming a truism in artificial intelligence; having the *right* restrictions on a given representation can make learning simple.

What gives further dramatic support to this finding is the isolation of several key processing principles that are apparently responsible for the program's success, specific *locality principles* (compare Koster [1978]) that tightly restrict the operation of the parser and the acquisition procedure. Central here are Marcus' key restrictions on the original Parsifal: *structural "determinism"* (once the parser builds structure, its decisions cannot be undone); *left-to-right operation* (the parse tree is built in a single left-to-right pass through the sentence) and *bounded context actions* (the parser makes its decisions based on the local context of the parse). In other words, the constraints that make acquisition easy can be classified into one of two groups; constraints on rule *application* — the locality principles — and constraints on rule *form* — the pre-specified format for grammar rules.

The way in which Parsifal imposes these constraints is straightforward. Take first the restrictions on rule application. Parsifal grammar rules consist of simple productions of the form *if* pattern *then* action , where a *pattern* is a set of feature predicates that must be true of the current environment of the parse in order for an *action* to be taken. *Actions* are the basic tree-building operations that construct the parse tree. Adopting the operating principles of the original Parsifal, grammar rules can trigger only by successfully matching features of the (finite) local environment of the parse, an environment that includes a small, three-cell look-ahead buffer holding already-built constituents whose grammatical function is as yet undecided (e.g. a Noun Phrase that is not yet known to be the subject of a sentence) or single words. It is Marcus' claim that the addition of the look-ahead buffer enables Parsifal to always *correctly* decide what to do next — at least for English. The parser uses the buffer to make discriminations that would otherwise appear to require backtracking. Marcus dubbed this "no backtracking" stipulation the *Determinism Hypothesis*. The Determinism Hypothesis crucially insists that all structure the parser builds be part of the final output parse tree — that is, that already-executed grammar rules have performed correctly. This fact provides the key to easy acquisition; if parsing runs into trouble, the difficulty can be pinpointed as the *current* locus of parsing, and *not* with any already-built structure (previously executed grammar rules). In brief, any errors are assumed to be *locally* and *immediately* detectable. This constraint on error detectability appears to be a computational analogue of the restrictions on a transformational system advanced by Wexler and his colleagues. (See Wexler and Culicover [1980]). In their independent but related formal mathematical modelling, they have proved that a finite error detectability restriction suffices to ensure the learnability of a transformational grammar, a fact that might be taken as independent support for the basic design of Lparsifal.

Turning now to constraints on rule form, it is easy to see that any such constraints will aid acquisition directly, by cutting down the space of rules that can be hypothesized. To introduce the constraints, we simply restrict the set of possible rule patterns and actions . The trigger patterns for Parsifal rules consist of just the items in the look-ahead buffer and a local (two node) portion of the parse tree under construction — five "cells" in all. Thus, patterns for acquired rules can be assumed to incorporate just five cells as well. As for actions, a major effort of this research has been to demonstrate that just three or so basic operations are sufficient to construct the annotated surface structure parse tree, thus eliminating many of the grammar rule actions in the original Parsifal. (It is these restrictions on rule patterns and actions that ensure that the set of rules available for hypothesis by the acquisition procedures is finite.)

Apparently then, the exigencies of language processing (parsing) interact in intimate ways with the abstract constraints advanced by current theories of grammar; many of the same constraints that ease parsing — forcing decisions to be made locally — also aid the cause of acquisition. In brief, paying deliberate attention to the constrained theories of grammar developed by linguists has paid off handsomely; on the one hand, it has permitted the construction of a working computer program for the acquisition of syntactic knowledge; on the other, it has confirmed the value of theories of grammar as the basis for reasonable processing models of language acquisition and language use.

4.0 The acquisition procedure is simple

As mentioned, Lparsifal proceeds attempting to parse a series of positive (syntactically well-formed) example sentences such as who did Sue kiss, or, I gave the bottle to the baby. Parsing simply means executing a series of tree-building and token-shifting grammar rule actions. Actions, in turn, are triggered by matches of rule patterns against features of tokens in a small three-cell constituent look-ahead buffer and the local part of the annotated surface structure tree currently under construction.

Grammar rule execution is also controlled by reference to base phrase structure rules. To implement this control, each of the parser's grammar rules are linked to one or more of the components of the phrase structure rules. Grammar rules are defined to be eligible for triggering, or active, only if they are associated with that part of the phrase structure which is the current locus of the parser's attentions; otherwise, a grammar rule does not even have the opportunity to trigger against the buffer, and is inactive. This is best illustrated by an example. Suppose there were but a single phrase structure rule for English that gave the canonical order for a sentence as a Noun Phrase followed by a Verb Phrase, e.g.:

Sentence → Noun Phrase Verb Phrase (S → NP VP)

Flow of control during a parse would travel left-to-right in accordance with the NP—VP order of this rule, activating and deactivating potentially-matching bundles of grammar rules along the way. To illustrate, we start up the parse by activating any rules associated with the "S" packet. These are actually grammar rules that are *always active*, rules whose trigger patterns watch for the characteristic opening signs of English Sentences — Noun-Verb clusters such as *Sue kissed*...; *for*-*to* combinations like *For Sue to kiss*..., and so forth. Now suppose the parser actually detects one of these patterns in the input string (a grammar rule fires to this effect), and enters the S NP VP phrase structure rule. First, all grammar rules associated with the "S" remain active, since they must always be ready to trigger (even in the midst of assembling a new Noun Phrase). Second, all grammar rules associated with building the Noun Phrases at the head of a Sentence (in English, the Subject Noun Phrases) would now become *active*: this would include rules whose job it is to watch out for the tell-tale beginnings of Noun Phrases (such as articles or determiners) and rules specifically designed to recognize proper names as Nouns, among others. The parser now has the chance to build a Noun Phrase constituent, eventually advancing in order to construct a Verb Phrase.[8] This step is marked by deactivating the Noun Phrase building grammar rules and activating any grammar rules associated with constructing Verb Phrases.[9] Together with (1) the items in the buffer and (2) the leading edge of the parse tree under construction, the currently pointed-at portion of the phrase structure forms a triple that characterizes the current state, or *instantaneous description* of the parser.

If in the midst of a parse no currently known grammar rules can trigger, acquisition is initiated: Lparsifal attempts to construct a single new executable base phrase structure expansion, lexical insertion grammar rule, or transformational-like grammar rule. New rule assembly is straightforward. Lparsifal first tries to see whether the first item in the buffer can be *attached* to the parse tree under construction in compliance with the conventions of the X-bar theory. If so, it has constructed either a new lexical insertion rule or a new phrase structure expansion. If the X-bar restrictions cannot be upheld, Lparsifal attempts to build a new grammar rule. To do this, Lparsifal simply selects a new *pattern* and *action*, utilizing the current *instantaneous description* of the parser at the point of failure as the new *pattern* and one of three primitive (atomic) operations as the new *action*. The primitive operations are: *switch* (exchange) the items in the first and second buffer cells; *insert* one of a finite number of (specific) lexical items in the first buffer cell; and *insert a trace* (an anaphoric-like element, that is, an item that must be co-indexed with another Noun Phrase) into the first buffer cell. The actions have turned out to be sufficient and mutually exclusive, so that there is little if any combinatorial problem of choosing among many alternative new grammar rule candidates. Finally, if the system arrives at a successful new grammar rule it determines whether it must *merge* its new rule with an existing grammar rule that performs the same action in the same packet (same point in the base phrase structure) as the new one. Merging rules serves to generalize grammar rules that perform common functions. As mentioned, a further constraint is that the acquisition procedure itself cannot be recursively invoked; that is, if in its attempts to build a single new executable grammar rule the procedure finds that it must acquire still other new rules, the current attempt at acquisition is immediately abandoned.[10]

5.0 Simple scenarios show the procedure in action

To illustrate how the acquisition procedure works, the following two subsections present much-simplified scenarios for (1) the acquisition of a context-free base rule to handle certain English Verb Phrases and (2) the acquisition of a grammar rule to deal with inversions in certain English questions. Full details appear in Berwick [1980].

5.1 Acquisition of a Verb Phrase rule

In the X-bar theory, all base phrase structure rules for human grammars are assumed to be expansions of just a few templates of a rather specific form, roughly, XP → ...X... Here, the "X" stands for an obligatory phrase structure category (based on the features of lexical items such as Nouns or Verbs); the ellipses represent slots for other phrases or specified grammatical formatives, such as Articles, Adjectives, or Prepositional Phrases, among others. Actual phrase structure rules are fleshed out by choosing a particular lexical category as the "X" (also called the head of the rule) and settling upon some other way to fill out what may precede or follow it. For example, by setting X = N(oun) and allowing some other "XP" to the left of the Noun (call it the category "Determiner" or "Specifier") we would get one possible Noun Phrase rule, NP → Determiner N. Or consider the English Verb Phrase rule: here, the Object Noun Phrase is attached to the right of the Head of the rule (the Verb). Other choices give other languages: in German, for example, it can be argued that the object Noun Phrase should be attached to the left of the verb. (Thiersch [1978]) Given this template filling approach, the problem for the learner is essentially reduced to figuring out what items are permitted to go in the slots on either side of the "X". Note that the XP schema tightly constrains the set of possible hierarchical tree structures generated by the base phrase structure rules; for instance, no Noun Phrase rule of the form Noun Phrase → Article Adjective Noun Noun would be admi-sible as a core part of phrase structure.[11]

To see in outline how the X-bar constraints can simplify the phrase structure induction task, suppose that the acquisition procedure has already acquired a phrase structure rule for English main sentences.[12]

Sentence → Noun Phrase Verb Phrase

and now requires information to determine the proper expansion of a Verb phrase:

Verb Phrase → ???

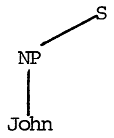
The X-bar theory cuts through the maze of possible expansions for the right-hand side of this rule. Assuming that Noun Phrases are the only other known category type, the X-bar theory tells us that there are only a few possible configurations for a Verb Phrase rule:

Verb Phrase → Verb | Noun Phrase Verb | Verb Noun Phrase   Noun
Phrase Verb Noun Phrase

⋮

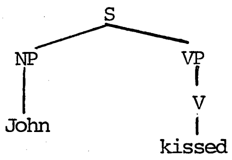
(other combinations of multiple Noun Phrase constituents)

Now suppose that the learner can classify word tokens as Nouns, Verbs or "other" (perhaps by at first linking items to some semantic grounding as "Substantive" and "Predicative").[13] Then, by simply matching an example sentence such as John kissed Mary (or more crudely, even a sentence like baby kiss mommy) against the array of possible phrase structure expansions, the correct Verb Phrase rule can be quickly deduced; only one context-free tree can be fit successfully against the given input string. Omitting much detail, the parse proceeds left-to-right with "John" first built as a Noun Phrase and recognized as a valid expansion of the already-known rule S → NP VP. The resulting tree is:



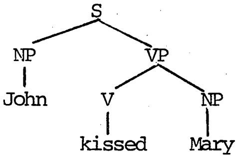
tokens to come: kissed Mary

Now, since "kissed" is supposedly recognizable as a Verb, the VP X-bar schema can be entered. But only one possible Verb Phrase rule expansion can successfully be matched against the sample string, namely, Verb Phrase   Verb(V)Noun Phrase(NP). For, the Head of the X-bar schema just entered must by definition be a verb; only "kissed" meets this criterion. Consequently, "kissed" must be attached as the "V" portion of the growing tree:



tokens to come: Mary

At a stroke, the options attaching Noun Phrases to the left of the Verb Head are ruled out. The only choice left open is whether the given input example is compatible with Noun Phrase attachment to the right of the Head. The next move of the parser shows that it is: Mary is recognizable as a name, hence a Noun and Noun Phrase; it is available for attachment to the Verb Phrase under construction, completing construction of the parse tree:



The VP expansion is as desired. A single simple example has provided positive evidence sufficient to establish a major English phrase structure rule. Although this is but a simple example,[14] it still illustrates how phrase structure rules can be acquired by a process akin to "parameter setting"; given a highly constrained initial state, the desired final state can be obtained upon exposure to very simple triggering data. The additional examples discussed in Berwick [1980] show how the entire phrase structure system can be reduced to the fixing of just six or so basic parameters within the X-bar framework. That report also probes an exploratory use of the X-bar templates as part of a theory of lexical acquisition. Finally, by establishing an order in which parameters may be fixed, the X-bar theory also delimits the possible "developmental envelopes" for acquisition, providing a theoretical framework for interpreting actual ontogenetic data.

4.2 A Subject-Auxiliary Verb grammar rule

Suppose that Lparsifal has all the syntactic knowledge (grammar rules and phrase structure expansions) to parse the sentence John will kiss Mary. Now suppose it is given the sentence Will John kiss Mary? No currently known grammar rule can fire, for all the rules in the phrase structure component activated at the beginning of the sentence will have triggering patterns looking for a Noun Phrase followed by a Verb, but the input stream will hold the pattern [Will + Verb[15]] [John: Noun Phrase], and so thwart all attempts at triggering a grammar rule. A new grammar rule must be written. The acquisition procedure first tries to attach the first item in the buffer, will, to the current active node, S(entence) as the Subject Noun Phrase. The attach fails because of category restrictions from the X-bar theory; since will is not marked as a Noun, it cannot be attached as the core of a Noun Phrase. But switch succeeds, because when the first and second buffer positions are interchanged, the buffer now looks like [John] [will].[16] Since the ability to parse sentences such as John will kiss... was assumed, a Noun Phrase-attaching rule will now match. Recording its success, the procedure saves the switch rule along with the current buffer pattern for remembering the context of auxiliary inversion. The rest of the sentence can now be parsed as if it were a declarative.[17]

Later example sentences of this type can be shown to quickly generalize the required trigger pattern for inversion to something like, [auxiliary verb] [Noun Phrase]. Introguingly, it can also be shown that the grammar rule so acquired also demands that there be no cyclic node immediately above the current active node (in this case, no cyclic node above an "S" node) at the time the inversion is performed. Thus, the Subject-auxiliary inversion rule will not erroneously trigger in an embedded environment — just as desired. (The ungrammatical sentence, I wonder did Sue kiss John should be unparsable after a training sequence consisting solely of positive example sentences of English.) In fact, it turns out that the acquisition procedure is exceedingly conservative, enlarging the class of sentences it can parse only when given positive evidence that it can do so. The computational and logical principles that balance acquisition between fatal over-generalization and timid under-generalization are currently being explored.

6.0 Summary of Key Ideas and Accomplishments

A working computer program has been developed that can acquire substantial syntactic knowledge of English under restrictions faithful to what is known about human acquisition (only simple, positive example sentences are used to trigger the creation of new rules). This knowledge is acquired in the form of an explicit process model of language use (a parser) during the on-going, left-to-right analysis of individual sentences. The syntactic knowledge acquired is of three sorts: context free base rules, lexical insertion grammar rules, and transformational-like grammar rules. Computationally, the system exploits the local and incremental approach of the Marcus parser to ensure that the search space for hypothesizable new rules is finite and small. The system acquires its complement of rules via the step-wise hypothesis of new rules. The ability to incrementally refine a set of grammar rules rests upon the incremental properties of the Marcus parser, which in turn might reflect the characteristics of the English language itself.

The constraints on the parser and acquisition procedure also par-parallel many recent proposals in the linguistic literature, lending considerable support to Lparsifal's design. Both the power and range of rule actions match those of constrained transformational systems. For instance, Wexler and Culicover [1980] Binary Principle (an independently discovered constraint akin to Chomsky's Subjacency Condition [1976]) can be identified with the restriction of rule pattern-matching to a local radius about the current point of parse tree construction (eliminating rules that directly require unbounded complexity for refinement). The remaining Wexler and Culicover sufficiency conditions for learnability, including their Freezing and Raising Principles, are subsumed by Lparsifal's assumption of strict local operation and no backtracking (eliminating rules that permit the unbounded cascading of errors, and hence unbounded complexity for refinement).

These striking parallels should not be taken — at least not immediately — as a functional "processing" explanation for the constraints on grammar uncovered by modern linguistics. An explanation of this sort would take computational issues as the basis for an "evaluation metric" of grammars, and then proceed to tell us why constraints are the way they are and not some other way. But this explanatory result does not necessarily follow from the identity of description between traditional transformational and Lparsifal accounts. Rather, Lparsifal might simply be translating the transformational constraints into a different medium -- a computational one. Even more intriguing would be the finding that the constraints desirable from the standpoint of efficient parsing turn out to be exactly the constraints that ensure efficient acquisition. The current work with Lparsifal at least hints that this might be the case. However, at present the trade-off between the various kinds of "computational issues" as they enter the evaluation metric is unknown ground; we simply do not yet know exactly what "counts" in the computational evaluation of grammars.

Acknowledgements

This article describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the Laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-75-C-0643. The author is also deeply indebted to Mitch Marcus. Only by starting with a highly restricted parser could one even begin to consider the problem of acquiring the knowledge that such a parser embodies. The effort aimed at restricting the operation of Parsifal flows as much from his thoughts in this direction as from the research into acquisition alone.

Footnotes

[1] For extensive justification, see Chomsky [1957]; [1965].

[2] The order of presentation may be considered to be random; that is, the external environment does not impose a structure on the order in which examples are given. Allowing a highly structured input of this kind can be shown formally to trivialize the task of acquisition, because one can encode the "right" hypotheses to pick in the presentation order itself. (See Gold [1967].) A characterization of what, exactly, is the primary data that children receive to learn language has been a topic of vast dispute over the past several years. The ordering of examples actually provided to children is probably neither completely random nor so structured as to provide possible encodings of grammars. Rather, as might be expected, the examples presented depend upon all those myriad things that determine what people say to each other and their children; events in the world, discourse, cognitive states of parent and child, and so forth. Yet

despite the immense possibility for individual variation in data input, the course of acquisition is remarkably uniform. This uniformity must come from somewhere: it derives either from the uniformity of the external world, that of the child (acquisition model), or some interaction of the two. In light of the controversy over what is input to the child, the current research has opted for the weakest assumption possible about external order, namely, *no* order; all structure is imposed by the internal order of the acquisition model itself. Note that even with this assumption there is a device whereby the effects of randomness can be partially "filtered" so as to ensure that (by and large) simple sentences are interpretable before more complex sentences. The competence of the system at any given point in time imposes an *intrinsic* filtering of the input examples, in that examples "too complex" for the parser to either (1) handle directly or (2) easily acquire rules for (in a sense of "easily acquire" that will be made precise later) are simply ignored. The first sentence of *Finnegan's Wake* might be input first, but it will simply not be (completely and successfully) interpreted, and thus not figure prominently in acquisition.

[3] There are *some* organizational modifications that are worth investigating. For instance, the ability to handle the "interrupts" characteristic of the processing of recursive phrases (especially, given Marcus' framework, the parsing of Noun Phrases as separate entities), might be a candidate for developmental change.

[4] For instance, such an acquisition procedure could never determine that the "garden-path" sentence, *The horse raced past the barn fell* is grammatical. Additional machinery must be added in order to accomodate such behavior.

[5] One might in fact distinguish at least *two* types of "negative evidence": (1) *explicit* negative information: the (perhaps methodical) pairing of positive (syntactically well-formed) and negative (syntactically ill-formed) sentences with the appropriate labels *well-formed* and *ill-formed*; (2) *indirect* negative information; correction of ill-formed utterances (alternatively, parses) via (a) explicit negative reinforcement (e.g. *That's wrong*) or (b) tacit negative reinforcement (e.g. responding with the correct pattern, or not responding). In certain restricted settings adult reinforcement patterns have been observed and might indeed aid acquisition — they help to overcome the limitations of attention and memory that might be expected of a primitive initial system. But this does not remove the necessity of a highly specified structure within which the reinforcement operates: only by reducing acquisition to the simple setting of a finite number of "parameters" (a finite domain) does simple reinforcement become plausible.

For additional remarks, see Wexler, Culicover [1980]; for evidence that children do not receive reinforcement for syntactic well-formedness, see Brown and Hanlon [1970]; or Newport, Gleitman, and Gleitman [1977].

However, children might (and seem to) receive negative evidence for what is a semantically well-formed sentence. This sort of input might prove to be of value in acquisition; for further discussion, see Brown and Hanlon [1970].

[6] This is, of course, but a casual demonstration that should properly be made formally. However, the "proof" that a finite search space allows an enumerative acquisition approach to converge in finite time is only of technical interest. It seems most unlikely that acquisition actually proceeds by enumerating entire grammars (or parsers); in any event, the time required to enumerate even modest-sized rule systems would be prohibitive (as discovered by those experimenting with such models; see Horning [1969]).

[7] The obvious extension to handle such cases is to simply add such a global re-analysis facility, invoked only when all straightforward methods have failed.

[8] Thus, once a sequence of constituents is (partially) recognized on the input stream (by definition, a data-driven or bottom-up process), the parse proceeds depth-first by attempting to recognize completely each of the elements of the sequence in turn.

[9] The scheme was first suggested by Marcus [1980, page 60] and implemented by Shipman [1979]. The actual procedure uses the X-bar phrase structure schemas instead of explicitly labelled nodes like "VP" or "S"; see Chapter Three for details.

[10] Obviously, a later attempt at the same sentence could lead to different results, if in the meantime the additional missing rule had been acquired by exposure to some other example sentence.

[11] Several constraints on the form of phrase structure rules have not been discussed here. For extensive discussion of the X-bar restrictions adopted in this report, see Berwick [1980].

[12] No provision has been made so far to account for the symbol "S" appearing in phrase structure rules — it is not a lexical category. There are two obvious alternatives: (1) there are special phrase structure rules (with constraints of their own) standing outside the X-bar system; (2) S is really some "X" category, most often assumed to be some projection of the category Verb. Both positions have been maintained: Position (1) Hornstein, [1977]; Lapointe, [1980]; Position (2) Jackendoff [1977]; Marantz [1980]. In this article, it will be more convenient to consider a Sentence to have a Verb projection as its head. The other alternative could be maintained, but at the price of introducing additional sorts of rules into the Parsifal interpreter.

[13] It is quite easy to extend this rudimentary initial categorization via the usual X-bar distinctive feature approach to a full set of lexical categories for English, including Adjectives, Adverbs, Articles, Quantifiers, Modal verbs, Prepositions, and Particles.

[14] Among other matters, it ignores the so-called "non-configurational" languages, those that do not (apparently) obey the same structural co-occurrence restrictions as English or German (and which may not even have VP nodes).

[15] More accurately, _will_ will be marked [+Verb-Noun] as a member of a more abstract "verb _like_" category that includes the modals such as _could_ or auxiliary verbs such as _have_ or _do_. There is, of course, considerable complexity to the English auxiliary verb system that is being glossed over here; auxiliaries and modals can appear only in certain orders, and in certain combinations.

[16] This procedure clearly presumes that an _entire_ constituent (in this case, a Noun Phrase) has been made available for switching into the first buffer cell. Following Marcus [1980 page 175], the parser does this by temporarily "shifting its attention" to the processing of the Noun Phrase starting with the token _John_. In the first case at hand, the Noun Phrase analysis is simple. But since English Noun Phrases may themselves contain sentential forms (hence other Noun Phrases), this approach leaves open the possibility of an infinite forward chain of attention shifts (hence look-ahead). There are several obvious restrictions that deal with this problem. One (adopted by Marcus) is to note that no "plausible" (or: descriptively sufficient) grammar rule for English ever requires more than five buffer cells total; any shifting beyond this local radius is prohibited (and leads to apparent processing difficulty). This restriction will be called the _total buffer-cell limitation_; it is also adopted in this report. Secondly, one might deliberately disallow recursive invocation of the acquisition procedure; this would push most complex Noun Phrases beyond the reach of the early parser. This restriction is also adopted here.

[17] The fact that a _switch_ was performed is also permanently recorded at the appropriate place in the parse tree, so that a distinction between declarative and inverted sentence forms can be maintained for later "semantic" use.

References

Berwick, R.C. (1980) Learning Structural Descriptions of Grammar Rules from Examples, unpublished ms (AI-TR 578)

Brown, R. and Hanlon, C. (1970) Derivational Complexity and Order of Acquisition in Child Speech, in J.R. Hayes (ed.), _Cognition_ and the Development of Language. New York: John Wiley and Sons.

Chomsky, N. (1957) Syntactic Structures. The Hague: Mouton.

Chomsky, N. (1965) Aspects of the Theory of Syntax. Cambridge, MA: MIT Press.

Chomsky, N. (1970) Remarks on Nominalization, in R.A. Jacobs and P.S. Rosenbaum, (eds.), Readings in English Transformational Grammar, Waltham, MA: Ginn.

Chomsky, N. (1973) Conditions on Transformations, in S.R. Anderson and P. Kiparsky, (eds.), A Festschrift for Morris Halle. New York: Holt, Rinehart and Winston.

Chomsky, N. (1976) Conditions on Rules of Grammar, Linguistic Analysis 2, 303-351.

Gold, E.M. (1967) Language Identification in the Limit, Information and Control 10, pp. 447-474.

Horning, J. (1969) A Study of Grammatical Inference, Technical Report TR CS-139, Computer Science Department, Stanford, CA: Stanford University.

Hornstein, N. (1977) S and the X-bar Convention, Linguistic Analysis 3, pp. 137-176.

Jackendoff, R. (1977) X-bar Syntax: A Study of Phrase Structure. Cambridge, MA: MIT Press.

Koster, J. (1978) Locality Principles in Syntax. Dordrecht, the Netherlands: Foris Publications.

Lapointe, S. (1980) A Theory of Grammatical Agreement, Ph.D. Thesis, Department of Linguistics, University of Massachusetts, Amherst, Published by Graduate Student Linguistics Association of the University of Massachusetts.

Marantz, A. (1980) English Vmax is the Head of S, unpublished manuscript, Department of Linguistics, MIT.

Marcus, M. (1980) A Theory of Syntactic Recognition for Natural Language, Cambridge, MA: MIT Press (publication of 1977 Ph.D. thesis).

Marshall, J. (1979)

Newport, E., Gleitman, H. and Gleitman, L. (1977) Mother, I'd Rather do it Myself: Some Effects and Non-effects of Maternal Speech

Style, in C. Snow and C. Ferguson, Talking to Children: Input and Acquisition. New York: Cambridge University Press.

Pinker, S. (1979) Formal Models of Language Learning, Cognition 7, pp. 217-283.

Shipman, D. (1979) Phrase Structure Rules for Parsifal, Working Paper 182. Cambridge, MA: MIT Artificial Intelligence Laboratory.

Thiersch, C. (1978) Topics in German Syntax. Unpublished doctoral dissertation, MIT Department of Linguistics, Cambridge, MA.

Wexler, K. and Culicover, P.W. (1980) Formal Principles of Language Acquisition. Cambridge, MA: MIT Press.