Non-final draft of a paper presented at the conference "Of Minds and Language: An Encounter with Noam Chomsky", The University of the Basque Country, Donostia - San Sebastian, Spain, June 2006. To appear in a volume edited by M. Piatelli Palmerini and P. Salaburu, Oxford University Press. TO QUOTE, PLEASE REQUEST PERMISSION.

Syntax acquisition: an evaluation measure after all?

Janet Dean Fodor

1. Introduction: Evaluating grammar hypotheses

First I would like to acknowledge the contributions of my collaborators, especially my colleague William Sakas, and our graduate students. We are all part of the CUNY Computational Language Acquisition Group (CUNY-CoLAG), whose mission is the computational simulation of syntax acquisition. We have created a large domain of languages, similar to natural languages though simplified, which we use to test the accuracy and speed of different models of child language acquisition.

I will start today by taking you back to 1965, to Chapter 1 of Noam Chomsky's *Aspects of the Theory of Syntax*, which I recommend to you all. It is, I think, one of the most important fifty pages of all of the important fifty pages that Noam has written, and it is still very germane today. So that will be our beginning point, but it won't be our ending point. We are going to look at Noam's outline of a program for how to set about modeling language acquisition, and then I will tell you why we haven't actually fulfilled it. The past few decades have seen many excellent acquisition studies of real children, studies of what they know and when they know it. But our job is modeling *how* children come to know these things, and that hasn't yet progressed very far at all. I thought that this Conference would be a wonderful occasion to bring a gift to Noam, so that I could say, "Here, in this box wrapped up with ribbons, is the learning model that you called for in 1965." But I don't have anything to give. I'm sorry. I can offer only an apology to Noam and an excuse, which is that the problems turned out to be really difficult, much more difficult than could have been anticipated. Why that is so is what I want to explain to you today.

What Noam asked us to do back then was to consider what must be involved in any acquisition model for language. He said there must be a *representation of the input signal* (the sound waves coming to the child's ears) *in terms of linguistic derivations*. Secondly, there has to be a *specification of the class of possible grammars*, i.e. all the candidate grammar hypotheses that the learner might contemplate. Third, there has to be a *method for selecting one of these grammars on the basis of the child's input*, i.e., an *evaluation measure*. And that turns out to be particularly difficult. The class of possible grammars is what linguists work on, but the evaluation measure (EM) determines the sequence in which learners try out different grammar hypotheses, so it is something that psycholinguists and computational linguists should have contributed to. But we still don't have it under control. EM is important, though, as a means of explaining why all children exposed to the same language make much the same choices and arrive at much the same grammar, and why they don't get confused along the way in the vast maze of alternatives. In addition, *Aspects* Chapter 1 notes that there must be a *strategy for finding hypotheses*. Even in a tightly constrained theory, there are many, many possible grammars. (Estimating how many is

easier to do in terms of parameters: if there were just 30 binary parameters, there would be more than a billion possible grammars, and that is probably an underestimate.) Because it is a huge search space, there has to be a method, as Noam observed, for finding hypotheses that fit the particular input sentences a child hears.

2. From rule creation to triggering

The details of the Chapter 1 blueprint for an acquisition model didn't last very long, because they were based on a notion of grammars as sets of rules and of acquisition as composing rules, and that never worked. There weren't enough constraints on the possible grammars, and there was no plausible EM for fitting grammars to the input. The next step, also from Noam,[1] was to shift from rule-based grammars to grammars composed of principles and parameters, which is what you have been hearing about at this Conference. Languages differ in their lexicons of course, but otherwise it is claimed that they differ only in a small, finite number of parameters. (I will limit discussion to syntax here, disregarding parameters for phonology and morphology.) An example is the Null Subject Parameter, which in languages like Spanish has the value [+ null subject] because Spanish permits phonologically null subjects, whereas in languages like English the setting is [- null subject] because subjects (of finite clauses) cannot be dropped. This is one binary syntactic parameter that a child must set.

The parametric model has properties that lighten the task of modeling language acquisition. Because it admits only a finite number of possible languages, the learning problem becomes formally trivial.[2] From a psychological perspective, input sentences can be seen not as a database for hypothesis creation and testing, but as *triggers* for setting parameters in a more or less 'mechanical' fashion. As Noam discussed earlier in this Conference, syntax acquisition then becomes simply a matter of tripping switches, a persuasive metaphor that he credits to Jim Higginbotham. A sentence comes into the child's ears; inside the child's head there is a bank of syntax switches; the sentence pushes the relevant switches over into the right *on* or *off* positions. Note that it is assumed that the triggers know which parameters to target. This will be important for the discussion that follows: the trigger sentences "tell" the learner which parameters to reset.[3] Finally, the principles-and-parameters model is a memoryless system, so it is economical of resources and it is plausible that a child could be capable of it. The child has to know only what the current parameter settings are, and what the current sentence is; she doesn't have to remember every sentence that she's ever heard and construct a grammar that generates them all.

So the parameter model was gratefully received, a cause for celebration. But then the bad news began to come in. Robin Clark (1989)[4] published some very important work in which he pointed out that many triggers in natural language are ambiguous between different parameter settings. One example of this is a sentence that has a non-finite

---

[1] Chomsky, N. (1981). *Lectures on Government and Binding*. Dordrecht: Foris Publications.

[2] Chomsky (1981), Chapter 1.

[3] Throughout this paper I will simplify discussion by assuming non-noisy input, i.e., that all input sentences are well-formed in the target language.

[4] Clark, R. (1989). On the relationship between the input data and parameter setting. *Proceedings of the 19th Annual Meeting of the North East Linguistic Society* (pp. 48–62).

complement clause with an overt subject, such as "Pat expects Sue to win". The noun phrase "Sue" has to have case, and it gets case either from the verb above it ("expect") or the verb below it ("win"). The former is correct for English ("expect" assigns case across the subordinate clause boundary), but the latter is correct for Irish, where the infinitive verb can assign case to its subject. Thus, there is a parameter that has to be set, but this sentence won't set it. The sentence is ambiguous between the two values of the parameter. There are many other such instances of parametric ambiguity in natural language.

Parameter theory had started with the over-optimistic picture that for every parameter there would be at least one unambiguous trigger, it would be innately specified, and learners would effortlessly recognize it; when that trigger was heard, it would set the parameter once and for all, correctly. What Clark's work made clear was that in many cases there would be no such unambiguous trigger; or if there were, a learner might not be able to recognize it because it would interact with everything else in the grammar and would be difficult to disentangle. This put paid to the notion that learners were just equipped with an innate list specifying that such-and-so sentences are triggers for setting this parameter, and thus-and-such sentences are triggers for this other parameter. Gibson and Wexler's (1994)[5] analysis of parameter setting underscored the conclusion that triggers typically cannot be defined either universally or unambiguously.

You should bear in mind *always* that the null subject parameter is not the typical case. It is too easy. With the null subject parameter, you either hear a sentence with no subject and conclude that the setting is [+ null subject], or you never do, so you stay with the default setting [- null subject]. There are important details here that have been much studied,[6] but even so, setting this parameter is too easy because its effects are clearly visible (audible!) in surface sentences. For other parameters, such as those that determine word order, there are more opportunities for complex interactions. One parameter controls movement of a phrase to a certain position; other parameters control movement of other phrases to other positions. The child perceives the end product of derivations in which multiple movements have occurred, some counteracting the effects of others, some moving parts of phrases that were moved as a whole by others, and so on. This interaction problem exacerbates the ambiguity problem. It means that even for parameters that have unambiguous triggers, they might be unrecognizable because the relation between surface sentences and the parameter values that license them is not transparent.

To sum up: Observations by Clark and others, concerning the ambiguity and surface-opacity of parametric triggers, called for a revision of the spare and elegant switch-setting metaphor. On hearing a sentence, it is often not possible, in reality, for a learner to identify a unique grammar that licenses it. At best, there is a pool of possible candidates. So either the 'mechanical' switch-setting device contains overrides, such that one candidate automatically takes precedence over the others; or else the switches aren't set until after the alternatives have been compared and a choice has been made between them. In either case, this amounts to an evaluation metric within a parameter-setting model. A second important consequence is that triggering cannot be error free. When there is ambiguity in the input, the learner cannot be expected always to guess the right answer. Thus, the original concept

[5] Gibson, E. and Wexler, K. (1994). Triggers, *Linguistic Inquiry* 25.3: 407-454.

[6] Extensive research was initiated by Nina Hyams: Hyams, N. (1986) *Language Acquisition and the Theory of Parameters*, D. Reidel, Dordrecht.

of triggering, though it was an extremely welcome advance in modeling grammar acquisition, proved to be too clean and neat to fit the facts of human language, and it did not free us from having to investigate how the learning mechanism evaluates competing grammar hypotheses. A problem that will loom large below is that evaluation apparently needs access to all the competitors, in order to compare them with respect to whatever the evaluative criteria are (e.g., simplicity; conservatism versus novelty; etc.), but it is unclear how a triggering process could provide the comparison class of grammars.

3. From triggering to decoding

All of this explains why, if you check the recent literature for models of parameter setting, you will find almost nothing that corresponds to the original Chomsky-Higginbotham conception of triggering. There are still parameters to be set in current models, but neither the mechanism nor the output of triggering has been retained. Instead of an 'automatic' deterministic switching mechanism, which has never been computationally implemented, it is assumed that the learner first chooses a grammar and then tests it to see whether it can license (parse) the current input sentence; if not, the learner moves on to a different grammar. This is a very weak system, and limits the ways in which the learner can select its next grammar hypotheses. A triggering learner, when it meets a sentence not licensed by the current grammar, shifts to a grammar that is similar to the current one except that it licenses the new sentence. That seems ideal, but current models do otherwise. For Gibson and Wexler's[7] system the principle is: *if the current grammar fails on an input sentence, shift to a grammar that differs from it by any one parameter*. For Yang's (2002)[8] model it is: *if the current grammar fails on an input sentence, shift to a grammar selected with probability based on how well each of its component parameter values has performed in the past*. Notice that in neither case does the input sentence guide the choice of the next grammar hypothesis. These are trial-and-error systems, quite unlike triggering not only in their mechanics but also in the choices they predict the learner will make.

By contrast, at CUNY we have tried to retain as much of the triggering concept as is possible. Although the 'automatic' aspect has to be toned down, we can preserve another central aspect, which is that the input sentence should tell the learner which parameters could be reset to license it. In a sentence like "What songs can Pat sing?", the *wh*-phrase "what songs" is at the front. How did it get there? In English, it got there by Wh-Movement, but other languages (Japanese, for example) can scramble phrases to the front, including *wh*-phrases. So as a trigger, this sentence is ambiguous between different parameter settings. Nothing can tell the learner which alternative is correct, but ideally the learner would at least know what the options are. We call this *parametric decoding*. The learning mechanism observes the input sentence and determines which combinations of parameter values *could* license it. Then it can choose from among these candidates, and not waste time and effort trying out other grammars that couldn't be right because they're incompatible with this sentence. Parametric decoding thus plays the extremely important role of guiding the learner towards profitable hypotheses. The problem is that nobody knows how decoding can be done within the computational resources typical of an adult human, let alone a two-year old.

---

[7] *Op. cit.*

[8] Yang, C. D. (2002). *Knowledge and Learning in Natural Language*. New York: Oxford University Press.

Our own learning model, called the Structural Triggers Learner, can do *partial* decoding. It uses the sentence parsing routines for this. We suppose that a child tries to parse the sentences he hears, in order to understand them. For a sentence (a word string) that the current grammar does not license, the parsing attempt will break down at some point in the word string. At that point the parsing routines search for ways to patch up the break in the parse tree, and in doing so they can draw on any of the other parameter values which UG makes available but which weren't in the grammar that just failed. The parser/learner uses whichever one or more of these other parameter values are needed to patch the parse. It then adopts those values, so that its current grammar is now compatible with the input. For any given input sentence, this decoding process delivers *one* grammar that can license it. But it does not establish *all* the grammars that could license an ambiguous sentence, because to do so would require a full parallel parse of the sentence to find all of its possible parse trees. That is almost certainly beyond the capacities of the human parsing mechanism. The bulk of the evidence from studies of adult parsing is that the parser is capable only of *serial* processing, in which one parse tree is computed per sentence and any other analyses the sentence may have are ignored.[9]

The limitation to serial parsing entails that the learner's parametric decoding of input sentences is not exhaustive. Partial decoding is the most that a child can be expected to achieve. But partial decoding is not good enough for reliable application of EM, because among the analyses that were ignored by the parser might be the very one that the EM wants the learner to choose. In some other respects, partial decoding is clearly better than none. Our simulation experiments on the CoLAG language domain confirm that decoding learners arrive at the target grammar an order of magnitude faster than trial-and-error models. But for our present concern, which is how learners evaluate competing grammar hypotheses, partial decoding falls short. It is unclear how EM could be accurately applied by a learning device that doesn't know what the set of candidate grammars is. So in a nutshell, the verdict on parametric decoding is that only full decoding is useful to EM but only partial decoding is possible due to capacity limits on language processing. Explaining how learners evaluate grammars is thus a challenge for acquisition theory.

4. The Subset Principle as test case

In what follows I will use the Subset Principle as a test case for evaluation in general. The Subset Principle (SP) is a well-defined and relatively uncontroversial component of the EM. It has long been a pillar of learnability theory and needs little introduction here. It is necessitated by the poverty of the stimulus — yet another major concept that Noam has given us. At CUNY we split the poverty of the stimulus (POS) into POPS and PONS.[10] POPS is *poverty of the positive stimulus*, meaning that learners don't receive examples of all the language phenomena they have to acquire, so they have to project many (most) sentences of the language without being exposed to them. A dramatic example is parasitic

[9] Parallel parsing is severely limited even in parsing models that permit it. See Gibson, E. and N. J. Pearlmutter (2000) Distinguishing serial and parallel parsing, *Journal of Psycholinguistic Research* 29.2, 231-240; also Lewis, R. L. (2000) Falsifying serial and parallel parsing models: Empirical conundrums and an overlooked paradigm. *Journal of Psycholinguistic Research* 29.2, 241-248.

[10] Fodor, J. D. and Crowther, C. J. (2002) Understanding stimulus poverty arguments. In N. A. Ritter (ed.) *A Review of "The Stimulus Poverty Argument"*, Special Issue of *The Linguistic Review*, 19.1-2, 105-145.

gaps, discussed by Noam in *Concepts and Consequences* and *Barriers*.[11] More pertinent for today is the *poverty of the negative stimulus* (PONS), which is extreme. Children typically receive little information about what is not a well-formed sentence of the language, certainly not enough to rule out every incorrect hypothesis that they might be tempted by.[12] Because of this, learning must be conservative, and SP is the guardian of conservative learning. Informally, the idea is that if a learner has to guess between a more inclusive language and a less inclusive language, she should prefer the latter, because if necessary she can be driven by further input sentences to enlarge a too-small language, but without negative evidence she could never discover that the language she has hypothesized contains too many sentences and needs to be shrunk. More precisely, SP says: when there is a choice to be made between grammars that are both (all) compatible with the available input sample, and the language licensed by one is a proper subset of the language licensed by the other, do not adopt the superset language. (From now on, for brevity, I will use "subset" and "superset" to mean "proper subset" and "proper superset" respectively.) SP is essential for learning without negative data. Without it, incurable overgeneration errors could occur. So it is evident that learners have some effective way of applying it. Our job is to find out how they do it – or even how they *might* do it, overcoming the technical snags that evaluation seems to face.

5. Enumeration of grammars

To get started, I must take you on another historical detour back to the 1960's. The work of Gold (1967) provides a straightforward and guaranteed solution to the problem of applying SP. Gold, a mathematical learning theorist, was not concerned with psychological reality, and you may well find his approach hopelessly clunky from a psychological point of view. Certainly it has not been taken seriously in any treatment of SP with psycholinguistic aspirations. But since it works, it is worth considering *why* it works and whether we can benefit from it. I will suggest that we can. Gold's approach needs a certain twist in order to make it psychologically plausible, but then it can solve not only the problem of how to apply SP but also another quite bizarre learnability problem that has never been noticed before: that under some very familiar assumptions, *obeying* SP can cause a learner to fail to arrive at the target grammar.[13]

Gold assumed an *enumeration* of all the possible grammars, in the sense of a total ordering of them, meeting the condition that a grammar that licenses a subset language is earlier in the ordering than all grammars licensing supersets of it. All the other grammars, not involved in subset-superset relations, are interspersed among these in an arbitrary but fixed sequence. (I will assume here that each grammar appears in the ordering just once.) The learner's hypotheses must respect this ordering. The learner proceeds through the list, one grammar at a time, moving on to consider a new grammar only when the preceding one has been disconfirmed by the input. The learner *thereby* obeys SP, without having to

---

[11] Chomsky, N. (1982) *Concepts and Consequences of the Theory of Government and Binding*, Linguistic Inquiry Monographs 6, MIT Press, Cambridge MA. Chomsky, N. (1986) *Barriers*, Linguistic Inquiry Monographs 13, MIT Press, Cambridge MA.

[12] Marcus, G. F. (1993). Negative evidence in language acquisition. *Cognition*, *46*, 53-85.

[13] Fodor, J. D. and Sakas, W. G. (2005) The Subset Principle in syntax: Costs of compliance. *Journal of Linguistics* 41.3, 513-569.

actively apply it or to know what the competing grammars for a given input sentence are. No decoding is required. The learner simply takes the next grammar in the sequence and finds out whether or not it can license (parse) the current sentence. Of course, learning in this fashion is a very slow business in a domain of a billion or more grammars, as the learner plods through them one by one. Steven Pinker wrote a very instructive paper in 1979[14] in which he admonished against trying to create psychology out of enumeration-based learning techniques. He wrote "The enumeration procedure exacts its price: the learner must test astronomically large numbers of grammars before he is likely to hit upon the correct one." After reviewing some possible enhancements to a Gold-style enumeration he concluded "In general, the problem of learning by enumeration within a reasonable time bound is likely to be intractable." From our CoLAG perspective, enumeration-based learning is an especially frustrating approach because it extracts so little goodness from the input. It has no room for parametric decoding at all. It proceeds entirely by trial and error, considering grammars in an invariant and largely arbitrary sequence that has no relation whatsoever to the sentences the learner is hearing. It is also rather mysterious where this ordering of grammars comes from. It must presumably be innate, but why or how humans came to be equipped with this innate list is unclear.

6. From enumeration to lattice

Despite all of these counts against it, I want to reconsider the merits of enumeration. Our CoLAG research has tried to hold onto its central advantage (fully-reliable SP application without explicit grammar comparisons) while improving its efficiency. You may find the question of origin just as implausible for our version as for the classic enumeration, but if I can persuade you to restrain your scepticism for a little while, I will return to this point before we are through. We have taken the traditional enumeration and twisted it around into a lattice (or strictly into a *poset*, a partially ordered set) which represents the subset-superset relations among the grammars, just as Gold's enumeration did, but in a more accessible format. The lattice is huge. The 157 grammars depicted in Figure 1 constitute about one twentieth of our constructed domain of languages. The domain is defined by 13 parameters, it contains 3,072 distinct languages, and in all there are 31,504 subset-superset relations between those languages. (The real-world domain of natural languages is of course much more complex than this, which is why we have to seek an efficient mechanism to deal with it.)

[14] Pinker, S. (1979). Formal Models of Language Learning, *Cognition* 7, 217-283.

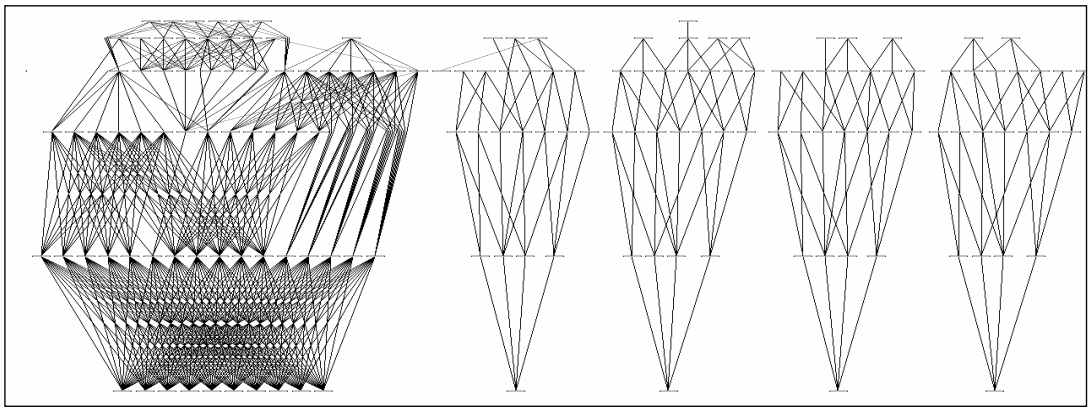

Figure 1. A fragment (approximately 5%) of the subset lattice for the CoLAG language domain. Each node represents one grammar. Each grammar is identified as a vector of 13 parameter values, but the grammar labels are suppressed here because of the scale. Superset grammars are above subset grammars.

This is how a learner could use the lattice. At the top of the lattice, as illustrated here, are nodes that denote the superset languages, with lines running downward connecting each one to all of its subsets, so that at the bottom there are all the languages that have no subsets. We call these *smallest languages*, and by extension the grammars that generate them are *smallest grammars*. These are the only safe (SP-permitted) hypotheses at the beginning of the learning process, and the learner may at first select only from among these. Because they have no subsets, the learner thereby obeys SP. As learning proceeds, these smallest grammars are tried out on input sentences and some of them fail. When this happens, they are erased from the lattice. That is: when a grammar is disconfirmed, it disappears from the learner's mental representation of the language domain, and it will not be considered again. This means the lattice gets smaller over time. More importantly: the pool of legitimate grammars at the bottom of the lattice gradually shifts. Some of the grammars that started out higher up in the lattice because they had subsets will trickle their way down to the bottom and become accessible to the learner, as the grammars beneath them are eliminated. They qualify then as smallest languages compatible with the learner's experience, so they have become legitimate hypotheses that the learner is permitted to consider.

This lattice representation of the domain provides a built-in guarantee of SP-compliance just like a classic enumeration, but it is much more efficient than an enumeration because there is no need for the learning device to work through every language on the way between the initial state and the target language. All it has to work through are all the subsets of the target language (beneath it in the lattice), which is exactly what SP requires. Our reorganization of the domain has cleared away the intervening arbitrarily ordered grammars which merely get in the way of SP in the one-dimensional enumeration. The lattice-based approach has other good features too. The erasure of grammars incompatible with the input makes syntax learning similar to phonological learning, where it is well established that infants start by making a great many phonetic distinctions which they gradually lose with exposure to their target language, retaining only

those relevant to the phonological categories that are significant in the target.[15] Also, the lattice-based model solves the other dire problem that I mentioned earlier: the fact that, although obeying SP is essential to avoid fatal overgeneration errors, it can itself lead to fatal errors of undergeneration.

7. Incremental learning and retrenchment

This disagreeable effect of SP stems from the assumption of *incremental learning*, i.e., that the learner makes a decision about the grammar in response to each sentence it encounters. After each input sentence, an incremental learner chooses either to retain its current grammar hypothesis or to shift to a new one. It does not save up all the sentences in a long-term database, to compare and contrast, looking for general patterns. Only the current grammar (the parameter values set so far) and the current input sentence feed into its choice of the next grammar, so it can forget all about its past learning events; it does not retain either sentences previously encountered or a record of grammars previously tested. Incremental learning thus does not impose a heavy load on memory, making it plausible as a model of children. Incremental learning was clearly implied in the original parameter setting model, and was regarded as one of its many assets. However, SP and incremental learning turn out to be very poor companions. To avoid overgeneration, SP requires the learner to postulate the smallest UG-compatible language consistent with the available data. But when the available data consists of just the current input sentence, the smallest UG-compatible language consistent with it is likely to be very small indeed, lacking all sorts of syntactic phenomena the learner had acquired from prior sentences. Anything that is not universal and is not exemplified in the *current* sentence must be excluded from the learner's new grammar hypothesis. We call this *retrenchment*. SP insists on it, because if old parameter settings weren't given up when new ones are adopted, the learner's language would just keep on growing, becoming the sum of all of its previous wrong hypotheses, with overgeneration as the inevitable result. SP thus makes an incremental learner over-conservative, favoring languages that are smaller than would be warranted by the learner's whole cumulative input sample to date. That can lead to permanent *undershoot errors* in which the learner repeatedly guesses too small a language, and never attains the full extent of the target. This doesn't happen always, but we observe undershoot failures in about 7% of learning trials in our language domain.

An example will illustrate the point. Suppose a child hears "It's bedtime". There is no topicalization in this sentence, so if the child is an incremental SP-compliant learner, there should be no topicalization in the language he hypothesizes in response to it (assuming that topicalization is something that some languages have and some do not). Similarly for extraposition, for passives, for tag questions, long-distance wh-movement, and so on. Even if the child had previously encountered a topicalized sentence and acquired topicalization from it (had set the appropriate parameter, or acquired a suitable rule in a rule-based system), that past learning is now lost. To make matters worse, this is the sort of sentence that the child is going to hear many times. So even if during the day he makes good progress in acquiring topicalization and extraposition and passives, every evening he will lose all that knowledge when he hears "It's bedtime". This is obviously a silly outcome, not what happens in real life, so we must prevent it happening in our model.

[15] See Werker, J. F. (1989) Becoming a native listener. *American Scientist* 77(1), 54-59; and references there.

The guilty party once again is the ambiguity of (many) triggers. If the natural language domain were tidy and transparent, so that there was no ambiguity as to which language a sentence belongs to, a learner would be able to trust her past decisions about parameter settings, and hold on to them even if they aren't exemplified in her current input. Then even a strictly incremental learner could accumulate knowledge. A parameter value once set could stay set, without danger of discovering later that it was an error. But the natural language domain is *not* free of ambiguity, so a learner can't be sure that her past hypotheses weren't erroneous. Hence previously adopted parameter values cannot be maintained without current evidence for them; retrenchment is necessary. But then the puzzle is how learners avoid the undershoot errors that retrenchment can lead to.

8. The lattice limits retrenchment

It seems that the familiar assumption of incremental learning may be too extreme. Incrementality is prized because it does not require memory for past learning events. But even an incremental learner could profit by keeping track of grammars it has already tested and found inadequate. Then it could avoid those grammars in future, even when the evidence that disconfirmed them is no longer accessible to it. Making a mental list of disconfirmed grammars would do the job, though it would be very cumbersome. But an ideal way to achieve the same end is provided by the erasure of disconfirmed grammars from the grammar lattice, which we motivated on independent grounds earlier. Erasing grammars will block repeated retrenchment to languages that are smaller than the target. The smallest language compatible with "It's bedtime" is at first very small. But as time goes on, the smallest of the smallest languages will have been erased from the lattice, and then some larger smallest languages may be erased, and so on. As time goes by, the languages that the learner is allowed to hypothesize, the accessible ones at the bottom of the lattice, will actually include some quite rich languages. Hearing "It's bedtime" won't cause loss of topicalization and extraposition once all the grammars that don't license topicalization and extraposition have disappeared, eliminated by earlier input. Note that keeping track of disconfirmed grammars by erasing them from the innate lattice is a very economical way of providing memory to an incremental learner. The learner doesn't have to keep a mental tally of all the hundreds or thousands of languages he has falsified so far, a tally that consumes more and more memory as time goes on. Instead, memory load actually declines as learning progresses. To summarize: Like a traditional enumeration, the lattice model offers a failsafe way to impose SP on learners' hypotheses; if combined with erasure of disconfirmed grammars it also provides a safeguard to ensure that SP doesn't get out of hand and hold the learner back too severely.

Where a lattice-based learner clearly excels over an enumeration learner is that, although it considers grammars in the right sequence to satisfy SP, it is not otherwise constrained by a rigid pre-determined ordering of all the grammars. For any input sentence, the learner must postulate a smallest language, but it has a free choice of which smallest language to postulate. Its choice could be made by trial and error, if that is all that is available. But a learner with decoding capabilities could do it much more effectively, because the input guides a decoding learner towards a viable hypothesis. And happily, for this purpose *full* decoding is not essential. Once decoding is used just to speed up learning, not for the application of the EM, partial decoding is good enough, because a lattice-based learner doesn't need knowledge of *all* the grammars that could license a sentence in order

to be able to choose one that is free of subsets; instead, the lattice *offers* the learner only grammars that are free of subsets. This is the heart of the lattice solution to the problem of applying EM. The evaluation metric is inherent in the representation of the language domain, so the question of which of a collection of grammars best satisfies EM doesn't need to be resolved by means of on-line computations, as had originally seemed to be the case. The whole cumbersome grammar-comparison process can be dispensed with, because EM's preferred languages are now pre-identified. The Gold-type enumeration, despised though it may have been on grounds of psychological implausibility, has thus taught us a valuable lesson: that evaluation of the relative merits of competing hypotheses does not inevitably require that they be compared.

9. Can the lattice be projected?

We seem to be on the brink of having a learning model that is feasible in all departments: learners' hypotheses are input-guided by parametric decoding but only as much as the parsing mechanism can cope with; SP applies strictly but not over-strictly; neither on-line computation nor memory is overtaxed. But there are two final points that I should flag here as deserving further thought.

First, the appeal of the lattice representation in contrast to a classic enumeration is that it permits constructive grammar selection procedures, like decoding, to step in wherever rigid ordering of grammars is not enforced by EM. But I want to post a warning on this. We are in process of running simulation tests to make sure that this ideal plan doesn't spring nasty leaks when actually put to work. The most important thing to check is that we can integrate the two parts of the idea: using the lattice to identify the smallest languages, and using partial decoding to choose among them. We think this is going to work out, but there's an empirical question mark still hovering over it at the moment.[16]

Finally, there's that nagging question of whether it is plausible to suppose that we are all born with a grammar lattice inside our heads. There's much to be said about this and about the whole issue of what could or couldn't be innate. It would be very exciting to be able to claim that the lattice is just physics and perfectly plausible as such, but I don't think we're there yet. In lieu of that, we would gladly settle for a rationalization that removes this huge unwieldy mental object from our account of the essential underpinnings of human language. If the lattice could be *projected* in a principled way, it would not have to be wired into the infant brain. It might be dispensed with entirely, if the vertical relations in the lattice could be generated as needed rather than stored. To do its job, the learning mechanism needs only (a) access to the set of smallest languages at the active edge of the lattice, and (b) some means of renewing this set when a member of it is erased and languages that were above it take its place. We are examining ways in which the lattice might be projected, holding out our greatest hopes for the system of default parameter values proposed by Manzini & Wexler (1987).[17] But at least in our CoLAG language domain, which is artificial and limited but as much like the natural language domain as we

[16] Performance data for several variants of the lattice model are given in Fodor, J. D., Sakas, W. G. and Hoskey, A. "Implementing the Subset Principle in syntax acquisition: Lattice-based models", to appear in *Proceedings of the European Cognitive Science Society*, 2007.

[17] Manzini, R. & Wexler, K. (1987). Parameters, binding theory, and learnability. *Linguistic Inquiry*, *18*, 413-444.

could achieve despite necessary simplifications, we have found exceptions — thousands of exceptions — to the regular patterning of subset relations that would be predicted on the assumption that each parameter has a default value which (when other parameters are held constant) yields a subset of the language licensed by the non-default value. Many subset relations between languages arise instead from unruly 'conspiracies' between two or more parameters, and they can even run completely counter to the default values.[18]

If these exceptions prove to be irreducible, it will have to be concluded that as-needed-projection of the lattice is not possible and that the lattice must indeed be biologically inscribed in the infant brain. We hold out hope that some refinement of the principles that define the defaults may eventually bring the exceptions under control. What encourages this prospect is the realization that the languages that linguists are aware of may be a more or less haphazard sampling from a much larger domain that is more orderly. The Subset Principle concerns relations between *languages*, which do not closely map relations between grammars. So the innate *grammar* domain may be highly systematic even if the *language* domain is pitted by gaps. Gaps would arise wherever the innately given lattice contains a superset-generating grammar lower than a subset-generating grammar. The subset grammar would be UG-permitted but unlearnable because its position in the lattice happens to violate SP (or some other aspect of EM). Such grammars would be *invisible* to us as linguists, whose grasp of what is innate is shaped by observation of the languages that human communities do acquire. In that case, the priority relations among grammars in the innate domain may be much better-behaved than they seem at present, and may after all be projectible by learners on a principled basis. And there would be no need to suppose that the grammar lattice was intricately shaped by natural selection to capture just exactly the subset relations between languages.

NOAM CHOMSKY: [asks a question about undershoot, triggers and parameters; microphone not functioning]

JANET FODOR: The question was, when the child has learned topicalization and set the topicalization parameter, why can that knowledge not be retained?

JDF ANSWERS: The culprit is the ambiguity of triggers. Because the triggers are ambiguous, any parameter setting the learner adopts on the basis of them could be wrong. So the learner has to be always on the alert that sentences she projected on the basis of some past parameter setting may not in fact be in the target language. But you are right that there was a missing premise in the argument I presented. It assumed that the learner has no way to tell which triggers are ambiguous and which are not. That's important, because clearly the learner *could* hold onto her current setting for the topicalization parameter if she knew she had adopted it on the basis of a completely unambiguous trigger. In most current models the learner cannot know this — even if it were the case. This is because the model

[18] Chomsky (1986 op cit., p.146) observes of the approach to evaluation that relies on a default value for each parameter that "this is a necessary and sufficient condition for learning from positive evidence only, insofar as parameters are independent", but then warns that they "need not and may not be fully independent". We agree.

parses each sentence with just one new grammar (when the current grammar has failed to parse it). But parametric ambiguity can be detected only by testing more than one grammar; and *non*-ambiguity can be detected only by testing *all* possible grammars. A learner capable of full decoding would be able to recognize a sentence as parametrically unambiguous. The more psychologically plausible STLs that do partial decoding can also recognize unambiguity, if they register every time they encounter a choice point in the parse. Even though the serial parser is unable to *follow up* every potential analysis of the sentence, it can tell when there are multiple possibilities. If such a learner were to set a parameter indelibly if its trigger was unambiguous, could it avoid the retrenchment problem? The data from our language domain suggest that there are so few unambiguous triggers that this would not make a big dent in the problem (e.g., 74% of languages have one parameter value or more which lack an unambiguous trigger). However, we are currently testing a cumulative version in which parameters that are set unambiguously can then help to disambiguate the triggers for other parameters, and this may be more successful.

PARTICIPANT: I was wondering whether any statistical measures would come in, because I think Robin Clark has suggested something of this kind in his earlier work: entropy measures, for example. Also David LeBlanc tried to build in parameter setting in a connectionist network: there was a statistical measure before a parameter was set.

JDF ANSWERS: Yes, the STL model we have developed at CUNY is actually a family of models with slight variations. The one we like best is one that has some statistics built into it.[19] What we have discovered, though, is the importance of using statistics over linguistically authentic properties. Statistical learning over raw data such as word strings without structure assigned to them has not been successful, so far anyway. Even very powerful connectionist networks haven't been proved to be capable of acquiring certain syntactic generalizations, despite early reports of success.[20] In our model — and Charles Yang's model has a similar feature — we do the statistical counting over the parameter values. A parameter value in a grammar that parses an input sentence has its activation level increased. This gives it a slight edge in the future. Each time the learner needs to postulate a new grammar, it can pick the one with the highest activation level, i.e., the one that has had the most success in the past. In the lattice model we have extended this strategy by projecting the activation boost up through the lattice, so that all the supersets of a successful grammar are incremented too, which is appropriate since they can license every sentence the lower grammar can license. Then, if a grammar has been quite successful but is eventually knocked out, all of its supersets are well activated and are good candidates to try next. Preliminary results (see footnote x above) indicate that this does speed acquisition.

[19] Fodor, J. D. (1998) Parsing to learn. *Journal of Psycholinguistic Research*. 27.3, 339-374.

[20] Kam, X-N. C. (2007) Statistical induction in the acquisition of auxiliary-inversion. *Proceedings of the 31st Boston University Conference on Language Development*. Somerville, MA: Cascadilla Press.