Language Acquisition, 19:83–143, 2012

ISSN: 1048-9223 print/1532-7817 online
DOI: 10.1080/10489223.2012.660553

Psychology Press
Taylor & Francis Group

ARTICLE

Disambiguating Syntactic Triggers

William Gregory Sakas
Hunter College and the Graduate Center, City University of New York

Janet Dean Fodor
The Graduate Center, City University of New York

The types of learning algorithm that could match the achievements of child learners depend in large part on how much parametric ambiguity there is in their input. For practical reasons this cannot be established for the domain of all natural languages. Our tactic is to estimate the incidence of unambiguous triggers by examining a constructed domain of languages whose syntactic parameters and structural properties are precisely specified. We succeeded in identifying unambiguous triggers for all non-default parameter values in all languages in this domain. In order to do so, we had to invoke between-parameter relations which disambiguate triggers that otherwise would have been parametrically ambiguous. The discovery of unambiguous triggers in this artificial domain does not prove a sufficiency in natural languages, but it may revive investigation of the psychological plausibility of deterministic models of human language acquisition.

1. THEORETICAL FRAMEWORK

1.1. Background

In this article, we present data from an artificial language domain that point to some new possibilities within the theory of syntactic triggers and that may help to resolve some current uncertainties in the field of computational psycholinguistics about how best to model parameter setting by children.

Correspondence should be sent to William Gregory Sakas, Department of Computer Science, Hunter College, North Building #1008, 695 Park Avenue, New York, NY 10065. E-mail: sakas@hunter.cuny.edu.

An appropriate computational model must be able to identify target language parameter values on the basis of a realistically limited sample of the language and to do so in a way that is compatible with what is currently known or surmised about the psychological capacities of 1 to 5-year-olds. But it is also important for a learning model to be compatible with what is known about the natural language domain: with what degree of precision is it *possible* for its languages to be projected from finite language samples? This is an empirical question; a range of answers can be imagined. If distinct languages have many sentence structures in common, a child's input sample may not be very effective in singling out the correct grammar, so some mechanism may be needed to aggregate information from a collection of sentences, none of which is decisive in itself. On the other hand, if every language has some unique structures of its own, the route from sample to target is more narrowly constrained and the learner's primary task would be to identify those informative sentences in the input. Clearly, the degree of ambiguity or unambiguity in the input has implications for what sort of computational system could achieve what children routinely do achieve. Studying the properties of the domain of natural languages can therefore help in constraining models of the child language acquisition mechanism.[1]

Our study asks whether, or to what extent, the triggers for syntactic parameters in natural languages are unambiguous. (An unambiguous trigger for value v of parameter P_i is a sentence, or sentence property, that is licensed only by grammars that have P_i set to v; see section 1.2.1.) For obvious practical reasons this cannot be established directly for the entire domain of possible natural languages. But that does not mean that no inroads can be made on the problem. Our tactic is to estimate the incidence of unambiguous triggers by examining a constructed domain of languages whose structural properties are precisely specified. It is simplified compared with the domain of natural languages but is rich enough to present some interesting challenges to learning models. The domain was created by the Computational Language Acquisition Group (CoLAG) at the City University of New York (CUNY) and consists of just over 3,000 artificial languages (described in section 2.1.1), constructed to be as much like natural languages as possible, compatible with keeping the size of the domain to manageable proportions for detailed study. All languages in the domain share general structural principles, which constitute the Universal Grammar (UG) of CoLAG.[2] The individual languages are generated by grammars defined by 13 binary syntactic parameters, which control familiar phenomena such as head direction, null subjects, *wh*-movement, topicalization, and so forth.

An initial inspection of overlaps among the languages in the domain revealed that almost half of the parameter values lack unambiguous triggers in some or all of the languages that need those parameter values. For 3 of the 13 parameters there are insufficient unambiguous triggers for *both* values, so that not even the adoption of a default value would circumvent

---

[1] Parametric ambiguity of input sentences is one of several manifestations of 'the poverty of the stimulus,' which has long been a central issue in theories of language acquisition (see Ritter 2002, and references there). We do not examine here other poverty issues such as the absence of positive exemplars or the scarcity of negative evidence. Also, we present no child language development data here, though clearly it will be important in future work to relate language domain findings to child language facts.

[2] For convenience, we use the term "CoLAG" as shorthand for "the CoLAG language domain."

the indeterminacy of input information. This would seem to provide clear support for learning models that do not rely on unambiguous triggers but are capable of patching together scraps of information derived from a collection of ambiguous input sentences. However, we will argue here that this first impression is misleading. On looking more closely, and with tolerance for some unconventional types of triggers, we have been able to identify unambiguous triggers for all nondefault parameter values in all of the CoLAG languages (see section 2, Empirical Investigation). This is quite surprising in view of the fact that when we created the domain we deliberately packed it with a variety of sources of parametric ambiguity in order to be able to compare the performance of different learning models under maximum pressure. Therefore, the positive outcome of our present investigation could encourage models of syntax acquisition in which learners seek out unambiguous information and rely on it primarily or even exclusively.

The discovery of unambiguous triggers for all of the CoLAG parameters is subject to two caveats that are obvious but need to be clearly stated at the outset of discussion. The first is that a sufficiency of unambiguous triggers in the CoLAG domain by no means proves a sufficiency in natural languages at large. But it would at least make it worthwhile to investigate further, e.g., to expand the domain in future research, progressively adding more natural language parameters to see how far the positive result continues to hold. The second point to be underscored is that in uncovering the initially "hidden" unambiguous triggers we had to employ various nonstandard tools of parameter theory, including *between-parameter defaults* and what we call *conditioned triggers*;[3] so the psychological plausibility of these devices will need to be scrutinized in the overall evaluation of this approach (see section 3.2). Even so, since the existence of triggers is at least a necessary condition for their use by children, our present findings may contribute to shaping the direction of research efforts in this area.

There are three main parts to this article. To set the scene for the empirical findings we continue in the present section (Theoretical Framework) by drawing a contrast (in section 1.1.1) between two very different conceptions of the process by which parameters are set, which diverge specifically with respect to how the learner responds to trigger ambiguity. This is followed by answers to some frequently asked questions (section 1.1.2) related to this line of research. Then in section 1.2 (Definitions of Triggers) we propose some extensions to the theory of triggers originally developed in foundational work by Gibson & Wexler (1994) and review the status of triggers in relation to E-language and I-language as discussed by Lightfoot (1999).

Two major sections follow. Section 2 (Empirical Investigation) includes Procedure (section 2.1), in which we present details of the CoLAG domain and our method for searching for triggers, together with some preliminary findings. The following subsections report two case studies in which triggers needed to be *created* by disambiguation (section 2.2, Triggers for the Optional Topic Parameter; section 2.3, Triggers for Verb Movement Parameters).

Finally, in section 3 (Implications for a Theory of Triggers) we summarize the techniques employed in establishing unambiguous triggers for the "problem" parameters (section 3.1,

[3]Theoretical devices such as these have been employed, to the best of our knowledge, only by Dresher and Kaye (1990) and Dresher (1999) in a model for setting phonological parameters. See discussion below.

Strategies for Disambiguating Triggers), and we make a start (section 3.2, Could a Learner Know and Use these Triggers?) on the task of assessing whether the triggers that we found are of an appropriate kind to play a role in a psychological model of human language acquisition and the extent to which they need to be biologically programmed. An outline of the structure of the article is as follows:

1.1.1. Divergent Approaches to Modeling Parameter Setting

As a mechanism for syntax acquisition, parameter setting was hailed as an explanation for the rapidity and precision of child language development. Yet efforts to model it over the intervening years have been few and far between, and most have drifted far from the classical concept of parametric triggering, which was originally regarded as a mainspring of learning efficiency. No model, either abstractly described or computationally implemented, has captured the notion of triggering as it was originally conceived (Chomsky 1981, 1986). That notion was only lightly sketched at the time, but it was widely understood to mean that setting a parameter on the basis of an input sentence is more or less instantaneous, and automatic in the sense of requiring little or no linguistic computation by the learner. A sentence was received, and the learning mechanism somehow knew which parameter(s) it had to reset to accommodate that sentence. Trigger recognition was regarded as accurate, with the consequence that parameter setting could be deterministic in the sense that a parameter value once triggered would never need to be revised.[4]

Prominent studies of parameter setting to date have departed from this characterization in almost all respects, for reasons reviewed below. While the learning process is sensitive to triggering information in the input, it is portrayed as a largely trial-and-error (nondeterministic) search through the class of possible grammars until one is found that licenses all the target sentences encountered. Though they vary greatly in other respects (compare Clark 1992; Nyberg 1992; Clark & Roberts 1993; Gibson & Wexler 1994; Kapur 1994; Briscoe 2000; Yang 2002),

[4]Determinism as we characterize it here permits default values to be revised to the marked value on encounter with appropriate triggers. We will assume throughout that deterministic learning does not preclude default settings. Except where it happens to be confirmed by input, a default setting amounts to a guess, but where deployed appropriately a default is a *safe* guess: if it is not correct there will be positive evidence to reset it.

these proposals share the assumption that syntactic parameter setting in the natural language domain could not proceed via the learner's recognition of—as opposed to mere reaction to—unambiguous triggers.

This dismissal or neglect of the classical 'switch-flipping' concept of triggering in recent computational research has its source in the realization that there is a considerable amount of parametric ambiguity in the natural language domain. Clark (1989) drew attention to examples of syntactic phenomena that could be licensed by more than one combination of parameter values, suggesting that a learner would have to resort to guessing between the alternatives; but a wrong guess could lead to serious errors, perhaps even incurable superset errors.[5] In the face of this warning about the prevalence of ambiguity, there are two opposite tacks that might be taken: the learning procedure might be either loosened up or tightened up.

Loosening it would mean giving up the goal of error-free learning and creating a more flexible trial-and-error procedure that could arrive at the target grammar eventually (even in the face of subset-superset relations), despite making ambiguity-induced errors along the route. The learning procedure finds its way to the right answer by being nudged back and forth by many inputs each unreliable in and of itself. (See Yang 2002 for an extended defense of this approach.) In particular, such a learner expends no effort in trying to identify which input sentences are unambiguous triggers. Instead, it adopts the strategy of treating all input sentences alike—because any input *might* be ambiguous, for all such a learner can tell.

The opposite response to the rampant parametric ambiguity among natural languages is to tighten the learning model by including rigorous tests to sieve out the ambiguous triggers, so that learning can be based solely on the unambiguous ones in order to avoid errors. Note that ambiguous input is incompatible with deterministic learning only in conjunction with the additional premise that the learning mechanism cannot discern which inputs are ambiguous and which are not. For a learner that could make this discrimination, even a high degree of ambiguity would be no hindrance. Suppose 95% of sentences in a learner's input sample were parametrically ambiguous. Error-free learning from unambiguous evidence would nevertheless be possible, at least in principle, as long as every (nondefault) parameter value has even a single trigger that is recognizably unambiguous and occurs reliably in learners' input.[6]

The first approach to parametric ambiguity (nondeterministic) is characteristic of the majority of recent parameter setting models of syntax acquisition (see references above), including some of our own (Fodor 1998b; Fodor & Sakas 2004). The second approach above (deterministic) is much closer to the spirit of Chomsky's original conception (Lightfoot 1991; Fodor

[5] Some solutions have been proposed in the literature for superset problems. For recent discussion, see, for example, Pearl & Lidz (2009) and references there.

[6] The value of unambiguity, even if awash in a sea of ambiguity, is what we will explore below. Others have advocated modeling syntactic parameter setting as a search for unambiguous triggers. See for example, Pearl (2007) who proposes that learners impose an *Unambiguous Data Filter* on their input. The evidence available for parameter setting is thereby reduced in quantity but improved in quality. Her approach and ours differ, however, since we aim here for strict unambiguity such as would support a fully error-free deterministic learner (in order to stave off excessive retrenchment forced by the Subset Principle; see below), while Pearl's filter is more lax, not designed to exclude *all* ambiguous triggers from the learner's intake; the imprecision of the filter permits some mislearning, which offers an explanatory account of language change in the manner of Lightfoot (1991, 1999) and others.

1998a; Sakas & Fodor 2001). These complementary research directions lead to very different concepts of what triggers are and how they function within the learning mechanism.

In a trial-and-error system, as modeled by Gibson & Wexler, Yang, and others, the learner tries out a hypothesized grammar to find out whether or not it licenses (i.e., can parse) the current input sentence; yes/no feedback from the parsing routines is then used to reinforce that grammar hypothesis either positively or negatively. In these models the learning mechanism does not need to be able to recognize the triggers for a parameter. Whether a parse attempt succeeds or fails does depend, of course, on the content of the parameter values in the grammar and their effects on the derivation of a word string, but the learning mechanism need not have any understanding of that: for the learning mechanism the parameter values can be completely anonymous. It needs to know only which values it used, so that it can adopt or reinforce them appropriately afterward; it does not need to know why those values succeeded or failed, or how they relate to the input sentence at all.

By contrast, for error-free learning based on unambiguous triggers, the learning mechanism must attend to the properties of an input sentence to discern which parameter values could have licensed it. Using input sentences only as a source of success/fail feedback on selected grammar hypotheses is not enough. To avoid even temporary errors, the learning mechanism must use the linguistic properties of a sentence to *guide* it toward a good grammar hypothesis to adopt. This is what we have called *parametric decoding* (e.g., Fodor & Teller 2000). Decoding was clearly a characteristic of the original 'switch-setting' model of parameter setting. Though the details of the mechanism were never filled in, we must presume that each switch had associated with it a pattern-detector or template of some sort, which specified the properties a sentence must have if it is to trip the switch. For instance, the switch for preposition stranding (versus pied-piping) might have a template specifying a preposition and its object separated by a word or phrase that is not part of the PP. Questions can be raised about the source of those templates (see section 3.2). Presumably they must be innately supplied, because they could not themselves be learned, but how or why they might have arisen in the course of language evolution is an open question (made even sharper in a pared back conception of the narrow faculty of the language such as that of Hauser, Chomsky & Fitch, 2002). For present purposes, what matters is that the feasibility of a model of this type depends crucially on whether a set of such templates can be formulated that will perform accurately for all learnable natural languages.

The learning procedure of a template-based model is more or less trivial once the templates are defined. However, defining the templates is not at all a trivial task, and to the best of our knowledge it has not been undertaken on any substantial scale for syntax. (Note that scale matters: evaluating triggers one parameter at a time does not suffice because the number of potential between-parameter interactions explodes combinatorially in the number of parameters.) Dresher & Kaye (1990) pioneered such a model for the setting of parameters for metrical phonology (see further discussion below). But Dresher (1999:64) points out that there is little carry-over to syntax: "Having arrived at a learning path for metrical parameters, for example, we cannot simply slot syntactic parameters into their place to arrive at a learning algorithm for the acquisition of word order. Rather we must establish anew what the cues and their ordering are for this domain." Thus, syntax could follow the lead of phonology in a general fashion, but must do its own work if it is to create a deterministic parameter setting model.

It is not our purpose here to decide between these divergent pictures of how triggers enter into the process of parameter setting, but only to decide whether the latter is computationally feasible (though see section 3.2 for some musings on psychological feasibility). In our previous work we have explored both deterministic and nondeterministic models (all of them making use of parametric decoding in one way or another). Indeed, we have urged the merits of each approach, at the risk of (real or apparent) inconsistency; compare Fodor (1998a) and Fodor (1998b).

The primary issue in the present article is whether high-precision deterministic parameter setting can succeed even in principle, in light of the properties of the natural language domain. It is our impression that a majority of syntacticians still hold to the classical concept of unambiguous triggers for every parameter, despite the pervasive doubts about this on the part of computational linguists. But neither view has been put to a sufficient test. The skeptical stance on unambiguity may well prove to be correct as research extends into more complex and more realistic language domains, but that has not been clearly demonstrated to date.[7] If unambiguous triggers do ultimately prove to be too scarce, it is important that computational linguists and psycholinguists should know it, and should *understand* why it is so, before moving on to other approaches.

1.1.2. Questions about the Project

A number of questions may arise about the theoretical and practical motivations for this project, as we know from previous presentations. In order to reserve space for the results of the empirical study, we deal with these questions as expeditiously as possible here (though we would agree that on many of these points there is much more of interest to be said). Readers familiar with these background issues may wish to skip to section 2, Empirical Investigation.

Why work with artificial languages? Whether unambiguous triggers exist for a particular language depends on *all* the languages in the learner's hypothesis space: ambiguity is a matter of how extensively and in what ways languages overlap. So it is impossible to check for unambiguity of triggers by examination of a corpus of sentences from one language, or even several languages. What is needed is an exhaustive representation of the properties of all languages in the hypothesis space, and at present this can only be achieved by *creating* the space of languages.

Why parameters? To establish whether every (nonuniversal) fact about a language could be acquired by a certain sort of learning mechanism, there needs to be a systematic classification of the facts that have to be acquired. Parameter Theory provides such a classification.

But other theoretical frameworks do so too and would be equally suited to an investigation of the availability or nonavailability of unambiguous evidence such as we present here. Optimality Theory specifies a set of constraints whose rankings are what need to be acquired. Categorial

[7] Bertolo et al. (1997a) offer a formal proof that unambiguous triggers must be insufficient in principle in domains that exhibit certain patterns of overlap between languages. However, the proof relies in part on the unsupported premise that a deterministic learner cannot employ default parameter values. Their result therefore does not apply to our analysis.

Grammar specifies a set of phrasal categories that need to be established in the lexicon. Head-driven Phrase Structure Grammar, LFG, TAG, Construction Grammar, and other frameworks likewise imply a certain class of facts that learners must establish in order to distinguish the target language from others. Any of these frameworks could be the basis for an investigation of whether the evidence for its acquirenda (to borrow a useful word from Pullum & Scholz 2002) is unambiguous and sufficient.

Traditional syntactic parameters may be falling on hard times these days, as they are relegated to the lexicon or the interfaces in the Minimalist Program, and especially as more and more microparameters are uncovered by descriptive and theoretical work.[8] If the descriptive work is correct in its implication that there are myriad (if not infinite) ways in which natural languages can differ from each other, then that will create comparable scaling-up challenges for learning models of all theoretical stripes. It seems likely in general that similar issues concerning the amount and reliability of evidence will arise in all linguistic frameworks—though it would be of great interest if it were to turn out that some learning problems are significantly diminished by adoption of one particular framework rather than another.

A practical reason for framing our investigation in terms of parameters is that we already have extensive data from simulation studies comparing different learning models for the parameterized domain of languages described here (see for example, Fodor & Sakas 2004). If anything like the classical triggering model with its reliance on unambiguous triggers could after all be implemented, then it would be valuable to test its performance on the same domain as in those previous studies, in order to be able to compare its efficiency with that of other learning models.

Why GB parameters? The parameters in the CoLAG domain are quaint by current standards. They stem from the era of Government Binding theory (GB), when parameters were in their heyday. This is not because we believe that those parameters are superior to latter-day refinements of them, but because GB constitutes a sort of lingua franca that many linguists, regardless of their own theoretical commitments, can comprehend and work with. Contributions to learnability theory are made by psychologists and computational linguists and by theoretical linguists of several persuasions, and since the problems are hard, it is very important to draw these different sources of expertise together. So a widely comprehensible vocabulary for presenting the issues and discussing possible solutions is beneficial.

Why UG at all? Data-driven approaches to language acquisition have been gaining ground during the past several years. Statistical learning procedures of several varieties have been developed that dispense with all preknowledge of possible grammars: not just parameters but all of the universal principles that linguists ascribe to UG. The more powerful of these systems, such as simple recurrent networks (SRNs), have registered notable successes with some complex syntactic phenomena, with no reliance at all on any linguistic concepts innate or otherwise (Elman 1993, and others). If these UG-free approaches should prove to be correct models of human language acquisition, then our current project of evaluating the quality of the

[8] See Biberauer (2008) for discussion of the status of parameters in the Minimalist Program, including the balance between macro- and microparameters; also Snyder (2007) for discussion of Minimalist parameters in relation to children's syntax acquisition. Gianollo, Guardiano & Longobardi (2008) have proposed a systemization of the class of possible syntactic parameters based on the inventory of syntactic features, with interesting explanatory implications.

input for triggering innately predefined acquirenda would be completely beside the point—as would most of linguistics. But to the extent that that outcome is in doubt (see Yang 2010), we believe it is of value to continue to develop the UG-based approach.[9] Proponents of UG must bear in mind, however, that UG can survive the competition from the new empiricist models only if its own capabilities are positively demonstrated. What is needed is some hard evidence that at least one UG-based learning model is capable of convergence on any of a wide range of realistically rich languages while consuming only limited time, input, and computational resources. In short: one step in the direction of proving the necessity of an innate grounding for language acquisition would be to hone the efficiency and psychological plausibility of UG-based models to the point at which they convincingly fulfill their theoretical promise of mirroring the achievements of child language learners.

Are deterministic models of interest? If it should turn out that unambiguous triggers are abundant after all, a natural next step would be to develop a learning model that makes maximum use of them. As noted above, a deterministic model is not feasible without them. But feasibility is of no interest if deterministic models can be rejected out of hand on grounds that they are computationally or psychologically implausible in other respects.

Deterministic learning has known disadvantages. It is very brittle, susceptible to possibly fatal errors based on ungrammatical examples in 'noisy' input (Nyberg 1992; Kapur 1994; Fodor 1998b). Determinism also typically (though not necessarily) implies instantaneous parameter setting, whereas children often waver, shifting gradually from one grammar to another (van Kampen 1997; Yang 2002). Determinism does not allow for retreat if a learner adopts a marked parameter value in error (though see Snyder 2007 for discussion of the rarity with which that occurs). Moreover, a deterministic system that succeeds in avoiding all errors would offer no acquisition-based explanation for language change or for the rapid development of creole languages (see Lightfoot 2006 and references there).[10] These are valid observations. But they do not necessarily call for a wholeheartedly nondeterministic learner. It may be possible to introduce some flexibility or statistical monitoring into what is fundamentally a deterministic design (e.g., to protect against noise by demanding multiple repetitions of a sentence type before accepting it as a parametric trigger, or to permit guessing if informative input is not available as in the case of pidgins). Such softenings of a basically precision-oriented mechanism might well fit the child language acquisition facts better than a fully nondeterministic model that does not seek out unambiguous information at all. For example, Snyder (2007) argues that the nondeterminism of models such as Yang's (2002) Variational Model predicts far more errors en route to the target than actually occur in children's spontaneous productions.

[9] We have drawn the contrast between UG-based and non-UG-based approaches very sharply here, in order to clarify the fundamental issues. But there has been recent interest in developing mixed approaches, such as frequency-based models that are grounded in UG (notably Yang 2002), or models that rely on some form of innate predisposition for language though unlike any standard version of UG (such as Bayesian 'priors,' cf. Perfors, Tenenbaum & Regier 2006).

[10] Determinism as conventionally (though loosely) understood in the language learning literature has three interrelated aspects:

(i) unrevisability of grammar decisions once made ('indelibility');
(ii) accuracy of first-pass grammar decisions; and
(iii) decision-making based on unambiguous information.

But these are not necessarily tied together. For example, creole development clearly lacks (iii) but might nevertheless involve (i).

Determinism has several compensating advantages. The most obvious is eliminating the labor of repeated resetting of the same parameter. Also, each parameter that is set by the learner halves the space of alternative grammars that remain in the hypothesis space for setting later parameters (see the Parametric Principle in Sakas & Fodor 2001:195). Importantly, determinism renders default values useful. Defaults are frequently posited in psycholinguistics (e.g., Hyams 1986 inter alia). They are essential for parameters whose values stand in a subset-superset relation, and in other cases they can contribute significantly to the efficiency of acquisition since no learning work is required for default values. However, a default setting is of very limited worth in a nondeterministic system. This is because in a trial-and-error process it might be set forward to the marked (nondefault) value by mistake, which could derail the subsequent setting of other parameters. How to prevent default values from being given up too soon by a nondeterministic learner is discussed by Gibson & Wexler (1994:432 ff.) and at length by Bertolo (1995) but with no fully satisfactory solution. For a deterministic learner, by contrast, a parameter remains safely at its default value until or unless unambiguous evidence for the marked value is encountered.

A significant new benefit of determinism has recently emerged. A severe problem in implementation of the Subset Principle has become apparent (Fodor & Sakas 2005), for which determinism just might be the solution. The problem concerns an unfortunate interaction between the Subset Principle (SP) and incremental learning, two mainstays of current thinking about child language acquisition. For any learner lacking access to (sufficient) negative evidence, SP requires the learner to posit the least inclusive language compatible with the evidence available.[11] For a strictly incremental learner, which has no memory for accumulating past inputs, the available evidence consists of just the current input sentence. SP insists on the least inclusive hypothesis compatible with that one sentence, so all previously acquired facts not reconfirmed in that sentence would have to be relinquished by the learner. This results in hopelessly weak grammar hypotheses, which in some cases may never achieve the richness of the target language (permanent undergeneration). We have called this the problem of *excessive retrenchment* and have considered potential solutions to it (Fodor & Sakas 2005; Fodor, Sakas & Hoskey 2007).

One solution is error-free deterministic learning. If the learning mechanism is *confident* that the parameter values it has already adopted are correct, because it knows they were based on unambiguous evidence, then the evidence currently available to the learner would include those parameter values as well as the current input sentence. The established parameter values would serve as a kind of memory, so retrenchment would no longer be excessive.

Why search for triggers? Plausible candidates for unambiguous triggers can be hypothesized by considering the linguistic contrasts that parameters induce. Linguists who present a newly discovered parameter often accompany it with a suggestion of potential triggers for it. So is it really necessary to scour three thousand languages in order to discover unambiguous triggers? Our experience from working with a large domain of parameterized languages has been that it can contain some nasty surprises, and that unambiguity of triggers is rarer than might be anticipated, due to just the sorts of complex interactions among parameters that

[11] For purposes of this study we are assuming that learners have little or no access to negative evidence, including 'indirect negative evidence' such as is available in some models, for example those with a statistical component.

Clark (1989) drew attention to. It is by no means a waste of time, therefore, to check such proposals empirically, to be sure that the prospective triggers do not dissolve into ambiguity in unexpected ways.

What counts as success? Even for a learning model that relies exclusively on unambiguous triggers, it is not necessary that *all* UG-compatible languages have unambiguous triggers for all of their nondefault parameter values. This is because it is not self-evident that all UG-compatible languages are humanly learnable. As noted elsewhere (Fodor 1989; Fodor & Sakas 2005; Fodor 2009), it is entirely possible that UG encompasses many languages that are not spoken by any human community (and hence are not known to linguists) precisely because they lack adequate triggers. Gibson & Wexler (1994:427) rightly point out that this defense is sustainable only if the languages that are predicted to be unlearnable do not include any attested language. But in any case, in the present project we will resist the temptation to resort to this kind of excuse if adequate triggers prove difficult to identify. Instead, we look for other ways in which unambiguous input information can be freed up to make learning possible.

Also pertinent to whether the outcome of a trigger hunt can be deemed successful is whether the triggers identified are simple, general, linguistically authentic, and psychologically plausible. There is room for disagreement as to what it takes to satisfy these criteria. In our project we have done our best to respect them at least informally. Where our search uncovers multiple triggers for a parameter, we attempt to subsume them under a single simple universal characterization, which it would not be unreasonable to suppose is innate.

What is known already about the incidence of unambiguous triggers? There are few precedents for this kind of investigation. While syntacticians continue to propose unambiguous triggers, computational linguists have tended to focus attention on ambiguity rather than unambiguity, as noted above. For phonological acquisition, Dresher & Kaye (1990) and Dresher (1999) identified unambiguous triggers ("cues") for setting 11 parameters controlling metrical structure;[12] see also the recent follow-up study of their parametric system by Pearl (Pearl 2007, 2009). Why the success of the DK/D model did not spur the development of comparable models for syntactic parameter setting is unclear.[13] Perhaps it was thought that syntax, with its greater number and abstractness of parameters, would be too complex or unruly for such an approach. But also, DK/D argued that deterministic learning from unambiguous triggers is incompatible with *incremental* (on-line) learning of phonology. They maintained, instead, that a batch learning approach is necessary: input items (words) must be stored and cross-compared

[12]The articles by Dresher & Kaye (1990) and Dresher (1999) contain related material. To save space here we do not include references to both articles where they make similar points, but abbreviate them jointly as DK/D. The 1990 article is of considerable interest because of its remarkably prescient recognition of many of the points of theoretical interest in this area; the 1999 article represents Dresher's more recent thinking on the topic and contrasts the deterministic cue-based approach with other models. Interested readers should consult both works.

[13]The proposals for syntax acquisition by Lightfoot (1999, 2006, and elsewhere) do have close affinities with the DK/D model for phonology, particularly the emphasis on I-language cues; see section 1.2.2 below. However, Lightfoot's ideas have not been realized in an implemented learning model; they have been pursued in investigations of two or three parameters at a time in the service of accounting for language change.

in order to set the parameters.[14] Mental storage of individual input items is not difficult to imagine as a normal aspect of phonological processing where the items are words, but it is far less plausible for syntax where the input items are sentences. In the present project we remain committed to incrementality; grammatical hypotheses are adopted on the basis of individual sentences. Despite this major difference between our approach and DK/D's, the results we report here are very close to theirs in their general implications for the theory of triggers. We flag some detailed points of convergence as they arise in what follows and sum up on the resemblance at the end of section 3.1 (Strategies for Disambiguating Triggers).

For syntax, only a few small-scale studies have assessed the actual availability of unambiguous triggers. Pearl (2007) examined triggers for one parameter, VP-headedness, in one language, Old English (following proposals by Lightfoot 1991). Some of the triggers for VP-headedness were rendered ambiguous by their overlap with the Verb Second parameter. Pearl's *Unambiguous Data Filter* netted some unambiguous triggers, not enough to guarantee convergence on the original target language, but enough to explain the slow rate of language change away from OV order to VO order over the course of 200 years or so. Previously, Gibson & Wexler (1994) had reported a striking absence of unambiguous triggers in an artificial domain of eight languages defined by two parameters fixing basic word order (SVO, OVS, SOV, OVS) plus a parameter for Verb Second. This was a very influential finding. However, hindsight shows it to have been due to a limited view of what a trigger can be (e.g., it must be an entire sentence). On the definitions of triggers we offer below (section 1.2), which we believe to be psychologically more appropriate, the Gibson & Wexler 3-parameter domain does indeed have unambiguous triggers for all three parameters in all eight languages. Subsequent work on a larger scale by Bertolo et al. (1997b) and Kohl (1999), employing extensions of the Gibson and Wexler domain up to 12 parameters, reported some parameter values unlearnable (though not always attributable to parametric ambiguity).

For our CoLAG domain of 13 parameters, we made one attempt in the past to assess whether all parameters have unambiguous triggers in all the languages that need them. We tested a deterministic decoding learner that employed sentence processing routines to identify parametric ambiguity in input sentences so that ambiguous items could be discarded for learning purposes (the "weak" version of the Structural Triggers Learner; see Sakas & Fodor 2001). However, to be quite sure of avoiding wrong settings this learner had to err on the side of caution and reject all inputs that the parsing routines identified as even possibly ambiguous. This had the result that valuable unambiguous triggers were sometimes discarded along with the ambiguous ones. Results of a simulation study showed that this model was very efficient when it succeeded, but for the majority of the target languages it failed (Sakas & Fodor 2003). Evidently it had been unable to identify unambiguous triggers for some target parameter values.

However, that initial study was inconclusive for two reasons. It tested not just the existence or nonexistence of unambiguous triggers in the languages of the domain, but also the efficacy of the learning regime being used to identify them. The latter was given a very challenging task. Equipped only with an abstract characterization of the content of each parameter value,

[14]The DK/D computer model YOUPIE relied heavily on such comparisons between words in batch mode and also on the negative evidence afforded by batch learning. Dresher & Kaye (1990:175) explicitly state: "it appears that a learner that is both purely incremental and strictly deterministic is unworkable, given the current set of parameters and cues". But in their theoretical discussions DK/D contemplated incremental alternatives (see footnote 34 below).

the parsing routines had to identify unambiguous instances of that value in the unstructured surface word strings that were its sole input. The observed learning failures might have been due to the absence of unambiguous triggers in the domain, but they might instead have been due to the insufficiency of the learning algorithm. Also, this early study did not make use of default parameter values, nor did it implement the important distinction between parameters irrelevant to a sentence and parameters that were relevant but ambiguously expressed. Thus, the negative results obtained in that study were almost certainly unduly pessimistic.

What does the present study add? This study approaches the issue from the more fundamental perspective of whether (to what extent) unambiguous triggers exist. Instead of simulating a learning model at work, we directly examine the domain of languages created for the simulations, to find out what unambiguous triggers it contains. With knowledge of that, we will be in a better position to decide whether or not it could be worthwhile to renew efforts to design a learning mechanism capable of recognizing unambiguity and taking advantage of the certainty which that confers.

1.2. Definitions of Triggers

1.2.1. Global and Local Triggers

The specific aim of this project is to determine whether or not the languages in the constructed language domain, serving as a surrogate for natural languages at large, have (recognizably) unambiguous triggers for all of their syntactic parameters. This requires a definition of what counts as a trigger. The motivations for the following definitions are given in Appendix B, where we take as our starting point the classic definitions of Gibson & Wexler (1994:409) and explain why some additional distinctions are needed. Note that our definitions are framed in terms of *sentence patterns*, by which we mean a possibly abstract characterization of some part (or all) of a sentence, such as a finite verb immediately followed by its object regardless of what else is in the sentence; further illustrations can be found in the discussion of the findings in section 2 (Empirical Investigation) and in Appendix A.

(1) A *globally valid trigger* for value v of parameter P_i, $P_i(v)$, is a sentence pattern π such that sentences instantiating π occur only in languages whose grammars have $P_i(v)$.

(2) A *globally available trigger* for value v of parameter P_i, $P_i(v)$, is a globally valid trigger for $P_i(v)$ that occurs in every language whose grammar has $P_i(v)$.

(3) A *locally valid trigger* for value v of parameter P_i, $P_i(v)$, is a sentence pattern π such that, among languages whose grammars have certain specified values for one or more parameters other than P_i, sentences instantiating π occur only in languages whose grammars have $P_i(v)$. The specified values for the other parameters will be said to *condition* the validity of the trigger π for $P_i(v)$.

(4) A *locally available trigger* for value v of parameter P_i, $P_i(v)$, is a (globally or locally) valid trigger for $P_i(v)$ that occurs in some but not all languages whose grammars have $P_i(v)$.

Note that availability and validity are distinct characteristics of triggers that differ importantly with respect to how a learner should respond to them. A globally *valid* trigger (whether it is globally or only locally available) is what we have called an *unambiguous* trigger in the discussion in earlier sections and in our previous work (Fodor 1998a, inter alia). Global validity means that there is no danger of mis-setting a parameter on the basis of that trigger when a learner encounters it. The consequence is that a learning mechanism can respond to all globally valid triggers, even if they are only locally available, in exactly the same way, employing them without risk of error. Their validity is not conditioned on knowing the values of any other parameters. By contrast, if a trigger sentence is only *locally valid*, this does make a difference to how a learner should respond to it, if the aim is to avoid errors. The sentence may occur in many languages and yet be a valid trigger for parameter value $P_i(v)$ in only some of them, depending on how their other parameters are set. Thus, a locally valid trigger can be employed safely only if the learner knows the values of the other parameters that condition its validity (and knows that those are the ones that matter).[15] Therefore locally valid triggers can be used safely only by a deterministic learner, which can be confident that the parameters it set previously are set correctly. For a nondeterministic learner, with parameters set on the basis of guesswork—even intelligent guesswork—these locally valid triggers would be unreliable.

Trigger validity and trigger availability differ not only with respect to reliability but also with respect to the learning costs associated with them. If a trigger for $P_i(v)$ is only locally available, the cost is that there must be additional triggers for $P_i(v)$ in order for $P_i(v)$ to be settable in all $P_i(v)$ languages. This relates to how economically the triggers for a parameter could be mentally characterized. Gibson & Wexler (1994) observed that the idea that triggers are innately specified becomes less plausible if many different triggers must be listed for a single parameter value because each one is very local. For a trigger that is only locally valid, the price is paid in the parameter setting process. $P_i(v)$ should not be set until after its conditioning parameters have been (correctly) set. In the meantime, whenever that trigger is encountered in the input, it would seem that the learner must first check the values of those other parameters, in order to know whether to adopt $P_i(v)$.[16] If so, then a parameter value that has only *locally* valid triggers is likely to be set later than an otherwise comparable parameter value that has *globally* valid triggers. Note that, since a locally valid trigger can be used only for languages with its conditioning parameter value(s), all locally valid triggers are also only locally available.

In view of these apparent drawbacks, why would a learner bother with local triggers? A learning model that used only global triggers would evidently be simpler and more efficient. But this is to the point only if global triggers exist. An error-avoiding learning mechanism has no choice but to employ locally valid/available triggers for any parameter that lacks globally valid or available ones; examples in our language domain are discussed below. Locally valid triggers could also speed up attainment of the target even if global triggers do exist but are encountered

[15] DK/D refer to "cross-parameter dependencies," explicated as follows: "if the cue for one parameter depends on having established a result for the setting of another parameter" (1990:173). We believe these dependencies are, or perhaps include, what we call conditioned triggers in the sense of triggers whose validity (not just availability) depends on the setting of one or more other parameters. DK/D encode these dependencies as an obligatory order in which phonological parameters must be set, which they call the "learning path." Proposals for the ordering of syntactic parameters include Baker's Parameter Hierarchy, which he relates to the sequence in which children set parameters (Baker 2001:192–195).

[16] A way of eliminating this on-line checking process is considered in section 3.2 below.

only rarely in the input. However, in our investigation of the language domain (section 2, Empirical Investigation), we made the simplifying assumption that the learning mechanism would rely solely on globally available triggers for any parameter that has them (regardless of whether this is the case for child learners). When we found a parameter lacking globally available triggers, we took the next step of seeking local triggers to take their place. Among locally available triggers we assumed that the learning mechanism favors globally valid (fully unambiguous) ones over locally valid (conditioned) ones, because the latter require reference to one or more other parameters as well. Note that definition (3) allows a conditioned trigger for $P_i(v)$ to be conditioned by the values of any number of other parameters (one or more). In our search for triggers we gave priority to triggers with the fewest conditioning parameters, since these have greater coverage and are psycholinguistically more economical. Though elaborate examples with many conditioners are perfectly possible, it is not clear whether learners would need to resort to them.

To summarize, unambiguity of triggers is the target of our investigation, and it aligns precisely with global validity in our trigger classification scheme, so we use the terms ‘globally valid’ and ‘unambiguous’ interchangeably in what follows. The most pertinent property of an unambiguous/globally valid trigger is that whenever it is encountered by a learner, it reliably signals the associated parameter value.[17] However, in the search for enough reliable triggers to support deterministic learning, it may be necessary to resort to locally valid/available triggers—with all due caution by the learner making use of them.

1.2.2. E-triggers and I-triggers

Characterizing triggers as sentence *patterns* permits generalization across many specific sentences. For instance, there are 8,650 distinct sentences in the CoLAG domain that are globally valid triggers for the parameter value +Preposition Stranding, but there are obvious commonalities among them. All are instantiations of the general pattern: a preposition, P, and its object, *O3*, are both present but not adjacent; see examples (5)–(7). (Concerning other details of these examples see section 2.1.1.)

(5) O3 Aux Verb P Adv

(6) O3[+WH] S P O1 Verb

(7) O3-wa Verb O1 O2 P S

That numerous individual trigger sentences can be schematically collapsed in such fashion is important for some learning models, those which assume that not only the parameter values but also their respective triggers must be innately specified. Obviously, that would be more plausible psychologically if the innate characterizations can take the form of one (or two or three) general schemas for each (nondefault) parameter value, rather than huge catalogs of

[17] Even ‘unambiguous’ triggers cannot guarantee protection against lower-level miscategorization errors such as taking a noun to be a verb. As is standard in computational studies of syntactic parameter setting, we presuppose here that the learner has already represented each input string correctly in terms of grammatical categories; this may not be so unrealistic (e.g., Mintz 2003 and related work).

individual sentences, most of which a learner will never encounter. It is important to note, however, that these trigger schemas must be formulated in terms of observable properties of word strings if they are to serve as the means by which learners recognize a trigger when they encounter it.

By contrast, Lightfoot (1999, 2006) has argued that the real triggers (or "cues" in his terms) for parameter values are elements of I-language, not of E-language (cf. Chomsky 1986).[18] For example, he proposes (8) as the I-language trigger/cue for the positive value of the Verb Second (V2) parameter, on the assumption that verb movement into C is necessary if SpecCP is filled (Lightfoot 2003:7).

(8) $_{\text{SpecCP}}$[XP]

Note that (8) identifies a property of the syntactic structure of sentences that is not directly observable in input word strings; it involves nonterminal nodes and a domination relation among them. So it does not provide a straightforward recipe for the recognition of triggers in input sentences. Rather, this characterization of V2 defines the linguistic essence of the parameter value, from which all of its derivational consequences will follow; in fact it can be regarded as *constituting* the parameter value. This is what in our own work we have referred to as a *structural trigger* (Fodor 1998a; Sakas & Fodor 2001). Structural triggers are 'treelets' (fragments, large or small, of sentential tree structure) that are made available to learners by UG and which can be adopted into the grammar of a target language if and when they are needed to license its sentences. For example, the structural trigger for +Preposition Stranding in CoLAG grammars is the feature [SLASH O3] as in the node label PP[SLASH O3], which by definition dominates a PP whose internal object (*O3*) has been extracted. With this feature in the grammar, sentences with stranded prepositions are generated; without it they are not.

Lightfoot's I/E terminology can be adapted to provide a framework for the presentation that follows. We will say that learners have access to innate, mentally specified, abstract *I-triggers* (*structural triggers*), which constitute the parameter values, while the observable patterns that realize I-triggers in input word strings encountered by learners are *E-triggers*. (Note that our definitions (1–4) above characterize what we are now calling E-triggers.) The question then arises how learners are able to relate each abstract structural I-trigger with the appropriate observable E-triggers. One possibility mentioned above is that these associations are simply stipulated in the innate basis for language acquisition: each I-trigger is mentally listed along with its E-triggers. A more ambitious proposal would be that only the I-triggers (the parameter values) are innately specified and that learners can derive the corresponding E-triggers from them. Lightfoot adopts this latter approach. He states explicitly that his model "makes no reference to elements of E-language or to the output of the grammar" (2003:19). However, for such a model to succeed in actually setting parameters, it is essential to establish that I-trigger characterizations suffice for E-trigger recognition by learners. Lightfoot does informally

[18] We quote here from a summary article in GLOT (Lightfoot 1998); page numbers are from Cheng & Sybesma (2003), which contains a reprint of the original. Similar ideas are developed in several of Lightfoot's other works. Note that in CoLAG, fronting of XP and raising a verb to C are independent and governed by different parameters.

address the question of how a child might detect an abstract I-trigger in input word strings. For I-trigger (8), for example, he proposes (1998:7; 2003:19) that the child could detect (9), which is what we would call an E-trigger.[19]

(9) A sentence-initial non-subject immediately followed by a finite verb.

However, this still leaves open the question of what kind of computations would enable a learner to translate back and forth between (8) and (9), or to establish I/E relationships more generally. Gibson & Wexler (1994) observed that such computations have never been articulated, and expressed doubt as to whether they would conform to plausible psychological constraints on children's processing capability. Because this issue has not been well-studied in any work to date, our policy here will be to let our investigation of the language domain provide more instances of actual E-triggers, and then (in section 3.2) assess whether their relationship with the corresponding I-triggers could be characterized in a principled, transparent fashion.

2. EMPIRICAL INVESTIGATION

2.1. Procedure

2.1.1. Database: The CUNY CoLAG Domain

The CUNY CoLAG language domain used in our experiments is a collection of 3,072 languages that was designed for the purpose of conducting simulation tests to compare the efficiency of different models of syntactic parameter setting. The domain is defined by 13 binary parameters encoding familiar syntactic differences between natural languages:

- Subject Position
- Headedness in IP, NegP, VP, PP
- Headedness in CP
- Preposition Stranding
- Topic Marking
- Null Subject
- Null Topic
- Wh-Movement
- Affix Hopping
- VtoI Movement
- ItoC Movement
- Q-Inversion (ItoC in questions)
- Optional Topic

[19]Though useful as an illustration of how I and E triggers relate, note that (7) is not in fact an unambiguous E-trigger for V2, since it is also compatible with other parameter value combinations, e.g., in head-final languages with null subjects without V2, such as Japanese.

For example, (the closest approximation in CoLAG to) English is head-initial in CP, in contrast to Japanese; it lacks VtoI Movement in contrast to French; it lacks null topics in contrast to German.

We use these conventional names for the parameters for convenience of reference, but it must be borne in mind that the actual linguistic consequences of these parameters are not entirely self-evident because they depend to various extents on how they interact with each other and with the 'universal grammar' of the CoLAG domain.[20] These parameters are all binary-valued and we assigned a default value to each, applying the following criteria: for parameters controlling movement, we took the nonmovement value as the default; for parameters controlling null elements, we assumed the default value disallowed null items; for parameters regulating optionality versus obligatoriness of some phenomenon, we took obligatoriness to be the default as required by the Subset Principle. (However, as will become clear in light of the empirical data, the correct defaults for some movement parameters are open to question; see discussion in section 2.3.3.[21]) For some parameters (e.g., those controlling headedness) none of these criteria applied; there seemed to be no good linguistic or learnability reason for designating one or other value as the default. However, for uniformity of presentation in this article we have designated a default value for every parameter, selecting it arbitrarily for parameters with no inherent asymmetry.

The CoLAG UG includes some constraints relating the values of different parameters, e.g., the positive values of Null Subject and Null Topic are mutually incompatible, as are those of Affix Hopping and V-to-I Movement (see details in sections 2.2 and 2.3 respectively); this is why there are only 3,072 distinct grammars, not the 2^{13} that might have been expected.[22]

The domain has a universal lexicon consisting of the following items: *S*, *O1*, *O2*, *O3*, *P*, *Adv*, *Aux*, *Verb*, *not*, *never, that*, *ka*, suffix *–wa*,[23] and the features, NULL, WH, FIN, SLASH, and features marking illocutionary force (the feature ILLOC with values DEC, Q, and IMP). The features NULL and SLASH are not realized in surface level sequences that the learner is exposed to, though they play a role in defining certain parameter values. *O1* is a direct object; *O2* is an indirect object; *O3* is the object of a preposition (or postposition, thus technically an adposition); *that* is a declarative complementizer, which is always phonologically null in main clauses; *ka* is an interrogative complementizer; *−wa* is a topic-marking suffix; *not* is the head of NegP above VP; *never* is an adverb located in specifier position (non-head daughter)

[20] Details of the most recent version of the CoLAG domain are available at http://www.colag.cs.hunter.cuny.edu/projects.html. (An earlier version is described in Sakas 2003 and Fodor and Sakas 2004.) The domain is also downloadable in its entirety from that site (all sentences of all languages with their tree structures).

[21] In addition to the movement parameters discussed in detail in this article (ItoC Movement and Q-Inversion), we note that the Wh-Movement parameter does not conform to these criteria. In CoLAG it has the values No Movement and Obligatory Movement. The Obligatory value had to be designated the default (see Table 1 below) because it licenses languages that are proper subsets of languages licensed by the No Movement value. This is because in CoLAG *wh*-phrases can be optionally topicalized into sentence-initial position even in languages which have the No Movement value for the Wh-Movement parameter. We introduced this ambiguity into the CoLAG domain as one of several ways to challenge learning models.

[22] Since some of the grammars license weakly equivalent languages, there are only 1,839 surface-distinguishable languages. The counts of missing triggers as in Table 1 below include *all* grammars, whether or not the languages they license are surface-distinguishable. Some cases of weak equivalence, for which distinguishing triggers are unneeded, are noted in section 2.

[23] The morpheme *wa* is the realization of a feature [+WA] on topics in the CoLAG domain.

of NegP. SLASH is a category-valued feature used to represent 'filler-gap' dependencies in the style of Generalized Phrase Structure Grammar (GPSG).

Sentences that are input to the learner consist of sequences of non-null lexical items (*S*, *O1*, *O2*, *O3*, *P*, *Adv*, *Aux*, *Verb*, *not*, *never, ka*, *−wa*) and non-null features (DEC, Q, IMP, FIN and WH).[24] For example, some possible sentences (from different CoLAG languages) are shown in (10)–(13).

(10) S Aux[+FIN] Verb Adv [ILLOC DEC]

(11) Verb[+FIN] S P O3 [ILLOC Q]

(12) O1[+WH] S O2 Verb[+FIN] ka [ILLOC Q]

(13) Verb[-FIN] O1 [ILLOC IMP]

Note that in what follows, we will omit features from sentence representations except when specifically relevant to the discussion.

Every sentence has one or more fully specified syntactic tree structures assigned by the grammars that license it. Structure is in general minimized in CoLAG except where specifically relevant to setting the parameters. There is no DP-internal structure. VP structure is flat: the complements of *Verb* are all sisters, ordered with *O1* closest to *Verb*, *O2* next closest, PP next, and *Adv* (a manner-type adverb within VP) furthest from *Verb*. All sentences are degree-0 (have no embedded clauses) and have no other recursion. The languages are therefore finite; they contain 827 sentences on average (range 288–2,148). The domain contains 48,077 distinct sentences, with 93,768 distinct syntactic trees, which means there is substantial syntactic ambiguity. Another indicator of the ambiguity level is that every sentence is licensed by 53 grammars on average (range 2–3,072). For instance, example (10) above has 12 distinct tree structures, two of which are shown in Figure 1. The sentences of each language were generated by a C++ implementation of a parameterized phrase structure grammar, with slash categories creating 'movement' chains in the manner of GPSG.[25] The general rules/principles of the grammar are fully universal (identical for all grammars in the domain); only the parameter values introduce language-specific structural options.

For the three parameters Optional Topic, ItoC Movement, and Q-Inversion that are the focus of following discussion in the present section, we present in Figure 2 the treelets (see section 1.2.2) that define their default and marked parameter values.[26] In sections 2.2 and 2.3 we will review their derivational consequences in detail.

[24]Note that in common with much previous work in this tradition, adoption of this lexicon presupposes, no doubt unrealistically, that learners have antecedently identified grammatical functions such as subject, direct and indirect objects, and the roles of morphological markers such as *wa*. (See, for example Fujimoto (2008) on imperfect early acquisition of Japanese particles.) Also, apart from the speech act indicators DEC, Q, and IMP, the input sentences are limited, as in much current computational research, by absence of semantics and prosody which, for child learners, can convey valuable phrase structure information; see Morgan (1986) and Morgan, Meier & Newport (1987).

[25]For a more recent version of this type of analysis, in Head-driven Phrase Structure Grammar (HPSG), see Sag, Wasow & Bender (2003:Chapter 14).

[26]Note that in section 2.3.2 we will consider reversing the marked and default values for ItoC Movement and Q-Inversion. Ramifications of this reversal are discussed in section 2.3.3.