Language Acquisition, 19:83–143, 2012 Copyright © Taylor & Francis Group, LLC ISSN: 1048-9223 print/1532-7817 online DOI: 10.1080/10489223.2012.660553



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

¹Parametric 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.

²For 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,

³Theoretical 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. THEORETICAL FRAMEWORK
 - 1.1. Background
 - 1.2. Definitions of Triggers
- 2. EMPIRICAL INVESTIGATION
 - 2.1. Procedure
 - 2.2. Triggers for the Optional Topic Parameter
 - 2.3. Triggers for Verb Movement Parameters
- 3. IMPLICATIONS FOR A THEORY OF TRIGGERS
 - 3.1. Strategies for Disambiguating Triggers
 - 3.2. Could a Learner Know and Use These Triggers?

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),

⁴Determinism 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

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

⁶The 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 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

⁷Bertolo 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. Headdriven 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

⁸See 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.

⁹We 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).

¹⁰Determinism 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

¹¹For 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

¹²The 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.

¹³The 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,

¹⁴The 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 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

¹⁵DK/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).

¹⁶A 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

¹⁷Even '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) _{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

¹⁸We 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

¹⁹Though 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 selfevident 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 2¹³ 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; *-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)

²⁰Details 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).

²¹In 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.

²²Since 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.

²³The 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.

²⁴Note 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).

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

²⁶Note 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.



FIGURE 1 Two fully specified CoLAG tree structures for S Aux[+FIN] Verb Adv [ILLOC DEC]. These trees are from languages that differ with respect to two relevant parameters. The tree depicted in Figure 1a is generated by a grammar with the Subject-initial value of the Subject Position parameter, and the no-movement value of the ItoC Movement parameter. The tree in Figure 1b is generated by a grammar with the Subject-final value of the Subject Position parameter with the Subject-final value of the Subject Position parameter and the obligatory-movement value of the ItoC Movement parameter. (Note that the SLASH feature is represented by '\'.)

Parametric treelets include only properties that do not follow from UG. For the treelets in Figure 2, CoLAG's UG will insert predictable nodes and features. For example, it will insert a Cbar right daughter of CP with necessary features including [SLASH X] when X is present as a left daughter; so the Optional Topic treelets do not need to explicitly include Cbar. Also when X is present, [ILLOC IMP] is prohibited, so UG will supply the disjunction of [ILLOC DEC] and [ILLOC Q] on the CP. Universal principles will also add the feature [ILLOC DEC] to the C node over *that* and [ILLOC Q] to the C node over *ka* in the default treelets for ItoC Movement and Q-inversion. The marked treelet for ItoC Movement also does not need to explicitly specify an ILLOC feature because CoLAG's UG stipulates that verb raising to C is limited to questions and declaratives (not imperatives). In contrast, the [ILLOC Q] feature



FIGURE 2 Parameter value treelets for three parameters in the CoLAG domain. Parentheses indicate an optional element. The figure depicts minimal specifications of the treelets; requisite features are automatically supplied by general principles of CoLAG's UG; as discussed above.

on the C node in the marked value for Q-inversion *must* be specified in the treelet in order to limit the verb raising to questions. Similar considerations apply in establishing the necessary details of the treelets for the remaining 10 parameters in CoLAG. Though we do not discuss them in detail in this article, we provide summary information in Table 4 in Appendix A.

With the CoLAG domain as our database, we can engage with a variety of learnability questions that would be unmanageable for the domain of all natural languages, due to current uncertainties about the range of languages that UG permits and the range of languages that have ever been acquired by a human community, not to mention uncertainty about the full range of languages that *could* be learned by humans. We reiterate here our disclaimer above: results from this domain clearly cannot be generalized to all natural languages. In particular, a demonstration that every parameter in this artificial domain has plausible unambiguous triggers does not entail that every parameter in the full domain of natural languages does. But we believe that the CoLAG domain comes closer than any other such domain presently available in systematically mirroring, despite simplification, the situation in the natural language domain.²⁷ It was designed to exemplify as many as possible of the phenomena that learnability theory concerns itself with, such as subset-superset relations, nonindependence of parameters, phonologically null elements, as well as considerable parametric ambiguity, in order to put learning models to a stringent test and find out which problems they could or could not solve. Within the bounds of our imposed size limit for the domain we strived to stay close to realistic natural language facts. Some concrete assumptions had to be made in order to move the CoLAG project forward, but it is true that other decisions might have been made that could alter the dynamics of the learnability issues that arise.

2.1.2. Method and Preliminary Data

Our procedure was as follows:

Step 1 (global validity): A computer program searched the domain to identify, for each value of each parameter, sentences that met the basic criterion for unambiguous (globally valid, though possibly only locally available) E-triggers. By definition (1) above, a globally valid trigger for parameter value $P_i(v)$ is a property of sentences that is instantiated in some or all languages with that value and in no languages with the opposite value. In this first-stage search of the domain, the triggers that were identified were very superficial; each was a complete surface sentence, a string of overt lexical items and features, with no analysis yet as to which of its properties qualified it as a trigger. (Later, we looked for relevant properties common among them; see step 5 below.)

Step 2 (availability): We then checked to see whether every language with a given parameter value $P_i(v)$ contained at least one of the unambiguous E-triggers that had been found for that value. As shown in Table 1, for 5 of the 13 parameters (Subject Position; Headedness in

²⁷Other language domains that have been created for purposes of learnability research on natural language syntax include those reported by Gibson & Wexler (1994), Bertolo et al. (1997a, b), Kohl (1999), Briscoe (2000), Villavincencio (2001). Readers should check with the original authors concerning domain availability and further details. Other computational studies have employed corpora of child-directed speech without specifying a full domain of grammars so they are not suited to the present goals for reasons noted in section 1.1.2.

Parameter (P_i)	Default Value	$P_i(v) = Default Value$	$P_i(v) = Marked Value$
Subject Position	Initial	1	1
Headedness in IP, NegP, VP, PP	Initial	1	1
Headedness in CP	Initial	1	1
Preposition Stranding	No	1	1
Topic Marking	No	1	1
Null Subject	No	66% missing	1
Null Topic	No	49% missing	1
Wh-Movement	Obligatory	100% missing	1
Affix Hopping	No	50% missing	1
VtoI Movement	No	75% missing	1
ItoC Movement	No	16% missing	25% missing
Q-Inversion (ItoC in questions)	No	50% missing	100% missing
Optional (versus obligatory) Topic	Obligatory	50% missing	23% missing

 TABLE 1

 Results of a Preliminary Search for Globally Valid (though Not Necessarily Globally Available) E-triggers in the CoLAG Domain

Note. \checkmark indicates that all CoLAG languages with $P_i(v)$ contain at least one globally valid E-trigger for $P_i(v)$; "missing" indicates the percentage of languages with $P_i(v)$ that have no globally valid E-triggers for it. Data for the three 'problematic' parameters are highlighted.

VP, NegP, IP, and PP; Headedness in CP; Preposition Stranding; Topic Marking), both values of the parameter were such that every language with that value had at least one unambiguous E-trigger for it.

This was not the case for the other eight parameters. Recall that for convenience of presentation here, every parameter is assigned a default value and a nondefault value in Table 1, even if there is no real (linguistic or learnability) basis for recognizing one value as the default. Five parameters in Table 1 (Null Subject, Null Topic, Wh-Movement, Affix Hopping, and VtoI Movement) exhibit an asymmetry in trigger availability indicative of a genuine default/nondefault distinction. Since the value with insufficient triggers could be designated as the default, and the other value (nondefault) had sufficient triggers to set it, we regarded these cases as unproblematic.

For lack of space we cannot discuss the 10 unproblematic parameters in detail here. Some of their triggers are characterized in Table 4 in Appendix A. For the remaining three parameters (highlighted in Table 1), neither value had an unambiguous E-trigger in every language with that value (ItoC Movement; Question Inversion; Optional Topic). These we classified as potential "problem" cases, which would need some sort of rescue in order for them to be set without guesswork by an error-averse learner. They will be discussed in detail in sections 2.2 and 2.3 below.

Step 3 (irrelevance): For each problem parameter P, we identified languages for which P was irrelevant, i.e., languages licensed by grammars differing in the value of P but otherwise identical. Our criterion for learning success was that the learner should arrive at a language that is surface-indistinguishable from the target language; in other words, only weak equivalence with the target is required (because strong equivalence cannot be demanded when sentences

have no other structural indicators such as prosody or semantics). Therefore when a parameter is irrelevant as defined here, either one of its values is equally satisfactory as the end state of learning. Hence no triggers are needed to distinguish them; the learner may retain the default value or may choose freely if the parameter has no default. Weakly equivalent languages were grouped into equivalence classes for purposes of all further analysis. In some cases this reduced the number of problem languages for a parameter (e.g., Findings 3 and 14 below), but in no case did it remove a parameter entirely from the problem list.

Step 4 (local validity): For the three problem parameters for which unambiguous E-triggers were still lacking, we searched the domain for locally valid (conditioned) triggers to substitute for the missing globally valid triggers. We looked for triggers with n conditioning parameters before seeking triggers with n + 1 conditioning parameters, and we terminated the search once we had found sufficient triggers to set the parameter in all languages for which it is relevant. As we report below, by means of *between-parameter defaults* and *conditioned triggers* we were able to identify unambiguous triggers for all three of these parameters.

Step 5 (compactness): Once a sufficiency of unambiguous E-triggers had been identified for all parameters, attention turned to the issue of how compactly those triggers could be characterized. This step involved traditional linguistic analysis, which we did by hand; a computer search then checked the generalizations we had devised. Specifically, we looked for commonalities among a parameter's unambiguous E-triggers to see whether they could be subsumed under one or a few general schemas. For instance, in section 1.2.2 (E-triggers and I-triggers) we noted that all of the sentences in the CoLAG domain that exemplify preposition stranding, though they differ in many details due to the settings of other parameters, can be subsumed under the simple generalization that a preposition and its object are both present in the sentence but are not adjacent. This formulation is more abstract than a list of complete, specific E-trigger sentences (as in Step 1), but it still makes reference to observable properties of word strings that a learner could recognize in unstructured input. It does not refer to unobservables such as empty categories or the tree structures of sentences, which are important ingredients of the Itriggers that capture the real linguistic essence of parameter values. Our empirical methodology was designed to detect E-triggers only. Deeper issues concerning how the E-triggers we found relate to I-triggers are addressed in section 3 (Implications for a Theory of Triggers).

Step 6 (confirmation): Finally, we rechecked the domain to confirm that every non-default parameter value had at least one (schematic) E-trigger in every language with that value and to find out how few such triggers were needed to cover *all* the languages with that value.

Having made in this section a preliminary sort of the parameters into unproblematic and problematic ones, we now work through the problem cases, showing how additional unambiguous triggers can be liberated for them. The three problem parameters are Optional Topic, I to C Movement, and Question Inversion. As Table 1 shows, while the Optional Topic parameter has some globally valid E-triggers, neither value has valid triggers for all languages with that value; the minus value is worse off in this regard than the plus value. For the ItoC Movement parameter, triggers are missing for both values, affecting somewhat more languages with the plus value than with the minus value. For Question Inversion, half of the languages with the minus value lack a trigger for it, and no language has triggers for the plus value.

In the next two subsections (Triggers for the Optional Topic Parameter and Triggers for Verb Movement Parameters) we address these problem parameters in turn. For each one we will enumerate relevant findings followed by explanations and potential solutions. For the most part the findings are presented in the sequence outlined in Steps 1–6 above; occasionally we depart from that order for clarity of the explanations. Throughout this section it should be borne in mind that both the problems and their solutions are bound to differ in some details from those that would arise for "real" languages, due to the simplified nature of the parameter space we are working with. But even so, they can make a significant contribution to theory development by revealing the *kinds* of problems that can arise in a complex parameter domain and the range of solutions available for dealing with them.²⁸

2.2. Triggers for the Optional Topic Parameter

The Optional Topic parameter (henceforth OptTop)²⁹ provides a good illustration of how the actions of other parameters can rob a parameter value of what would otherwise be unambiguous triggers. OptTop interacts with a number of parameters, including most notably the Null Topic parameter (henceforth *NullTop*), but also the Null Subject and Topic Marking parameters and even the underlying word order parameters. Even though these other parameters have unambiguous triggers of their own, they form a tangle in which each can have an impact on the triggers for the others.

All languages in the CoLAG domain have a topicalization operation that moves an XP element (subject, direct/indirect object, object of preposition, PP, VP-adverb) leftward into Spec, CP in declarative sentences and *wh*-questions. (*Aux, Verb, not, never,* and *ka* are not topicalizable.) The role of the OptTop parameter is to modulate this operation: a -OptTop language has obligatory topicalization (as in German); a +OptTop language has optional topicalization (as in English). In the CoLAG domain there are no languages that lack topics entirely. In the grammars that define the CoLAG languages the treelet that licenses a topic (in both -OptTop and +OptTop languages) consists of a CP node whose daughter Spec, CP is one of the topicalizable elements (as depicted in Figure 2 above). The grammar of a +OptTop language contains another treelet as well, a CP that has no Spec daughter (as allowed by the parentheses in Figure 2); this derives sentences without any topic. In some languages (the +NullTop languages) a topic may carry the feature +NULL, which entails that it has no phonological realization in the surface word string; otherwise, topics are overtly realized.

²⁸As noted above in section 1.1.2 (What Counts as Success?), we recognize that some learnability problems may have no solution, since it is not a necessary truth that every language in a language domain should be learnable. But persistence in seeking workable triggers for all parameters is worthwhile as a way of assessing the psycholinguistic potential of parameter theory.

²⁹Terminological note: the parameter that we refer to in this article as the Optional Topic parameter, with values -OptTop and +OptTop, is called the Obligatory Topic parameter in our earlier publications and on our Web site (see footnote 20). We have found it is clearer if a parameter is named after its marked value, the one for which there must be triggers. (This naming convention should apply also to the parameter that governs *wh*-movement, which would then be called, unconventionally, the Wh-in-Situ parameter since the movement value yields subset languages relative to its in-situ value; see footnote 21 above. We refrain from this renaming in the present article, however, since *wh*-movement is not central to the discussion.)

Pre-publication manuscript. Please do not quote without authors' permission.

DISAMBIGUATING SYNTACTIC TRIGGERS 107

Possible Grammars	OptTop	NullTop	NullSubj	# of Languages in the domain
'topic-oriented'	_	+	_	768
	_	_	_	768
'subject-oriented'	+	_	+	768
	+	_	-	768

TABLE 2 'Topic-Oriented' Versus 'Subject-Oriented' Languages in the CoLAG Domain. Other Combinations of These Three Parameter Values Are Excluded by Between-Parameter Constraints

This parameter was included in the CoLAG domain as a prototypical representative of the class of subset-superset parameters. In general, optionality of a linguistic phenomenon creates a superset language relative to obligatoriness of that phenomenon (unless other parameters intervene). Thus, +OptTop could be expected to yield superset languages relative to -OptTop. It could be anticipated, then, that +OptTop would have unambiguous triggers, and -OptTop would not. But that is not how the facts lie. Both of these expectations are wrong or at best oversimplistic.

2.2.1. Globally Valid Triggers for Both Values

FINDING 1: Half of the -OptTop languages have unambiguous triggers for that value.

EXPLANATION: This is evidently not a problem, since it is over and above what learnability requires, given -OptTop as the default value. Its explanation is that the UG of the CoLAG domain includes a between-parameter constraint, which requires that all +NullTop languages be -OptTop. We introduced this constraint into the domain as a test-tube case for studying the consequences of nonindependence of parameters. This particular example has some plausibility in the intuitive division between 'subject-oriented' and 'topic-oriented languages.' We regarded languages that allow null topics as 'topic-oriented,' construed as entailing that their topics are obligatory. These topic-oriented languages contrast with 'subject-oriented languages,' which may permit null subjects but do not permit null topics and are always +OptTop.³⁰ These between-parameter constraints in the CoLAG domain's UG have the consequence that three binary parameters yield only four language types, as shown in Table 2. For a learner, these constraints provide handy shortcuts, setting two parameters with one trigger. As row 1 of the table shows, any language that has unambiguous triggers for +NullTop (and all +NullTop languages do; see Table 1 and discussion below) is thereby also established to be -OptTop, even if it contains no triggers specific to the -OptTop value. This is why -OptTop can have some unambiguous triggers despite it being a subset value.

³⁰These terms echo the classic grouping of languages into *subject-prominent* and *topic-prominent* by Li & Thompson (1976) although without all of the distinguishing properties that Li & Thompson adduced. Li & Thompson acknowledged languages that are both subject- and topic-prominent, which we do not. An influential analysis by Huang (1984) maintained that Chinese has both null subjects and null topics, which the CoLAG constraints disallow. But see Modesto (2008) for a recent analysis of null subjects in Chinese as null topics.

FINDING 2: +OptTop has unambiguous (globally valid) triggers in 1,176 of the 1,536 languages with that value. The languages with +OptTop triggers include all the (768) +NullSubj languages.

EXPLANATION: The +NullSubj cases are explained by the between-parameter constraint, which entails that all +NullSubj languages are +OptTop (see row 3 of Table 2). Thus, this is another case of one parameter value getting a free ride from another. All it takes to establish +OptTop in these languages is a trigger for +NullSubj; and all the languages in question do have unambiguous +NullSubj triggers (which are not circularly dependent on +OptTop; see Table 4 in Appendix A).³¹ Hence, in a null subject language no direct evidence concerning the absence of a topic in sentences is needed as a means of establishing that topics are optional.

2.2.2. Languages Not in Need of Triggers

FINDING 3: For +OptTop languages that are not +NullSubj (768 languages), the value of OptTop is irrelevant for 120. These languages all have –TopMark (no overt marking of topics; see below), and are all subject-initial, –NullTopic and either –ItoC Movement or Complementizer-final.

EXPLANATION: Direct evidence (an E-trigger) for +OptTop, to distinguish it from -OptTop, would have to consist of

- (i) evidence that there is no overt topic in a sentence, and
- (ii) evidence that there is no null topic in that sentence.

At least (i) is unavailable in these 120 languages, due to the combination of other parameter values noted above. The significance of -ItoC Movement or Complementizer-final is that the finite verb (whether in C or not) will never precede the subject. Thus, nothing can precede the subject except an overt topicalized complement of the verb. There are two possibilities. If there were a topicalized complement, it would be licensed by either +OptTop or -OptTop. If the sentence were subject-initial it would be impossible to tell from the surface string whether the subject is in topic position or is unmoved. So again, such sentences would be licensed by either +OptTop or -OptTop. Therefore the same surface language is licensed by both values of the parameter, and no triggers are needed because it makes no difference which way the parameter is set.

2.2.3. Some Psychologically Implausible Triggers

FINDING 4: Of the remaining 648 –NullSubj languages with +OptTop, 408 have globally valid triggers for +OptTop. The triggers each have one of five specific items in sentence-initial

³¹CoLAG does not follow Hyams (1986) in that we took –NullSubj to be the default (initial) setting. This was because the CoLAG domain lacks expletive subjects, clitics, and other sources of evidence that have been proposed for acquiring the –NullSubj value. Also, as noted, we applied (rightly or wrongly) the general guideline that any parameter value introducing phonologically null elements should be the marked value, requiring triggering by the input.

position: ka, Verb, Aux, not, never; they also have an overt subject and a complete set of complements in VP (i.e., all of O1, O2, P O3, and Adv).

EXPLANATION: Part (i) of the evidence for +OptTop is any sentence that clearly contains no overt topic. The five sentence-initial items in the triggers are all and only the elements in the CoLAG domain that cannot be topics. Since topic position is universally sentence-initial in CoLAG, a non-topic in initial position establishes that the sentence contains no overt topic. This is the first step toward establishing that it contains no topic at all. But part (ii) is also needed: evidence that the sentence does not contain a *null* topic.

Whether or not a null element is present in a sentence can be hard to discern. In general, distinguishing null items from absent items is an exacting task for a trigger-based learning mechanism. But this is the source of the curious fact that all of the (non-null-subject) E-triggers for +OptTop exhibit what we will call a 'full house' of complements, i.e., the sentence contains all possible objects and adjuncts in VP, as well as an overt subject, for example (14).

(14) Verb O1 O2 P O3 Adv S

No sentence that lacked an indirect object, for example, or a PP complement to the verb, showed up among the unambiguous triggers. This 'full house' property of +OptTop triggers, though decidedly peculiar, does offer proof that the sentence contains no inaudible topic. If everything that *could* have been topicalized and then phonologically erased (due to +NullTop) is overtly present in a non-topic position, it is clear that nothing has in fact been topicalized and then phonologically erased.

Even though these full-house sentences beginning with a non-topicalizable element are the only unconditioned triggers that our domain search found for +OptTop, we rejected them on linguistic and psychological grounds. Even where they exist,³² they are far from ideal because of the unnaturalness of the full-house requirement, which relies on a property of the CoLAG domain that is not representative of natural languages at large. The absence of recursion in the CoLAG languages means that the notion of all (topicalizable) items being overtly present in a sentence is well-defined. By contrast, natural languages contain a rich array of optional adjuncts, with no bound on how many can appear within a sentence. Even considering only single clause (degree-0) sentences, the number of topicalizable constituents that *could* have been present is high. Moreover, even if the concept of a maximum length sentence were well-defined, it seems perverse to suppose that children could set a parameter only on the basis of these enormous sentences.

In short: Apart from the +NullSubj 'free-ride' triggers, the only globally valid triggers for +OptTop are the full-house initial-non-topic triggers described above, which are unsatisfactory. They are globally valid but only locally available. They do not account, either alone or together, for all the +OptTop languages that need triggers. They can only exist in languages where sentence structure is limited in ways atypical of natural languages. We will not consider them further here. In the absence of any other globally valid triggers, the only recourse was to open

 $^{^{32}}$ Even if full-house triggers were deemed acceptable, Findings 3 and 4 together leave 240 of the –NullSubj languages with +OptTop lacking any full-house triggers for it. This is because of their word order (e.g., verb-final), which precludes any of the nontopicalizable items in sentence-initial position in a full-house sentence (see discussion of Finding 9 below).

the search for locally valid (i.e., conditioned) triggers. Note that at that point, evidence of type (i) was in place for only some of the +OptTop languages (those with sentences with an initial-non-topicalizable element), and evidence of type (ii) was absent for all the +OptTop languages now that full-house triggers have been ruled out (See (i) and (ii) in the Explanation of Finding 3).

FINDING 5: +OptTop has some 1-conditioned triggers (i.e., triggers conditioned by just one parameter value), as well as some conditioned by multiple parameter values. One conditioning parameter value is +TopMark (overt topic marking). The +TopMark-conditioned triggers occurred in 384 of the 648 languages still in need of +OptTop triggers. The +TopMark-conditioned triggers were all *wa*-less sentences in +TopMark languages. All were full-house sentences.

EXPLANATION: These triggers violate the ban we have imposed on full-house triggers, a problem concerning component (ii) of evidence for +OptTop that we will address below. Meanwhile, it is instructive to note the contribution that +TopMark makes to component (i). +TopMark languages have a topic-marking morpheme -wa suffixed to any non-null topicalized element. In such languages, though clearly not in the whole domain, a sentence of a kind that *could* have a topic (i.e., a declarative sentence or *wh*-question) but which contains no -wa morpheme ipso facto has no overt topic. Thus, this conditioned trigger provides the same sort of information as the sentence-initial non-topic triggers discussed above: information that no *overt* topic is present. What is gained by +TopMark conditioning is that the sentence-initial element need not be a non-topicalizable constituent. Even a sentence such as (15) qualifies as unambiguous type (i) evidence for +OptTop, given +TopMark.

(15) Adv O3 P O2 O1 Verb Aux S

An important guideline for use of conditioned triggers by a deterministic learner is that the learner should be confident that the conditioning parameter value is correctly set; hence, the conditioning value should itself have sufficient unambiguous triggers (see further discussion in the Explanation of Finding 7 below). In the present case, +TopMark is safe as a conditioner for +OptTop triggers because +TopMark meets this requirement. There is an ample supply of +TopMark triggers in all +TopMark languages: every CoLAG language contains some overt topics, and every overt topic in a +TopMark language is overtly marked by *wa* (though see footnote 24). However, since +TopMark-conditioned triggers for +OptTop are not available in -TopMark languages, part (i), as well as part (ii), of the OptTop parameter is not yet fully resolved.

2.2.4. Unsafe Conditioning by -NullTop

FINDING 6: +OptTop has triggers conditioned by -NullTop, which occur in 528 of the 648 -NullSubj languages for which +OptTop is relevant. Some are full-house sentences, but most are not.

EXPLANATION: Conditioning by –NullTop substitutes for the linguistically dubious full-house requirement, because it does the same work: it guarantees that there is no null topic in the

sentence, i.e., component (ii) of a trigger for +OptTop. If a language can be established as lacking null topics, then a sentence starting with a recognizable non-topic can trigger +OptTop even if it is not a full-house sentence. In fact, –NullTop conditioning does even more work than the full-house requirement, because it applies in a wider range of languages, including those that lacked full-house versions of the triggers with sentence-initial non-topicalizable items due to their parameter settings for word order (see footnote 32 above). Given –NullTop, even a simple intransitive sentence such as *Verb S* is an unambiguous trigger for +OptTop (though a sentence such as *S Verb* still is not, except in a +TopMark language). The languages that we found with triggers for +OptTop conditioned by –NullTop do have sentences starting with a recognizable non-topic, thereby satisfying (i) while the –NullTop conditioning guarantees (ii). However this does not exhaust the languages in need of triggers for +OptTop.

FINDING 7: +OptTop has triggers conditioned jointly by -NullTop and +TopMark, in all 648 -NullSubj languages for which +OptTop is relevant.

EXPLANATION: Conditioning by +TopMark is another way of satisfying requirement (i), as noted above (Finding 5). It can render even a sentence with an initial subject (e.g., *S Verb*) valid as a trigger for +OptTop, if conditioned by -NullTop as well as +TopMark. What Finding 7 shows is that +TopMark conditioning supplements the initial-non-topic triggers sufficiently to cover all the -NullSubj languages that need +OptTop to be set.

Drawing this together, conditioning of +OptTop by +TopMark satisfies requirement (i) where needed. Conditioning by -NullTop satisfies requirement (ii) where needed. Conditioning by -NullTop is not open to the linguistic objections that led us to reject full-house triggers. Conditioning of +OptTop by -NullTop is also very productive: it frees up many more sentences to serve as +OptTop triggers. In fact we have found that together with +TopMark, it can supply unambiguous triggers for *all* of the +OptTop languages for which OptTop is relevant (and not entailed by +NullSubj; see Finding 2). However, despite this considerable potential, it appears that -NullTop is not qualified to serve as a conditioner. This is because it does not meet the requirement that a conditioner must itself have sufficient unambiguous triggers (see Finding 5 above). Almost half of the languages with -NullTop have no triggers for it, which is why we took it to be the default value. +NullTop can be the marked value because it does have sufficient unambiguous triggers (see Table 1 and below).

The ban on conditioning +OptTop by -NullTop is not specific to this particular pair of parameters. For a deterministic learner, default values are broadly ruled out as conditioners for other parameter values by the principle that a conditioning parameter value must have reliable triggers of its own.³³ How important is the principle? Could it be dispensed with, or flouted in some cases? DK/D considered the consequences of ignoring it. A default value that is assumed by learners in the absence of input evidence is not guaranteed to be correct: subsequent input might overturn it. If that happens, the learner would switch to the marked value, at which

³³More precisely, the real principle is that, for deterministic learning, a parameter value should never be used as a conditioner in any language in which it is not reliably supported by input evidence. Thus, if a default value happens to have unambiguous triggers in language L_i , it can safely be used to condition other parameter values in L_i , though it cannot in general. (A nondeterministic learner might employ conditioned triggers without any such safeguards, since it cannot avoid errors in any case. *None* of its parameter values would be fully trustworthy as a conditioner because the learner would have no certainty that they were correct.)

point any values of other parameters that had been adopted based on the default would be left stranded, lacking justification. For safety, the learning mechanism would need to retrench on those stranded parameter values, setting them back to their initial values. If this needed retrenchment failed to occur, the result could well be overgeneration, which would be a fatal error for a deterministic learner. If retrenchment did occur, it might or might not be targeted specifically on the parameter settings that had been conditioned by the default. If retrenchment unselectively returned *all* marked settings to their default values, that would constitute a substantial shift in the learner's grammar and unnecessary loss of reliably acquired correct parameter settings as well. In order for retrenchment to target just the relevant parameters, the learning mechanism would have to have been keeping a running record of which parameters had been set on the basis of which others, and that would add considerably to processing load.

In short, employing a default parameter value as a conditioner for another parameter value might make up for a lack of triggers, but the means of doing so appear to be unsafe or wasteful or computationally cumbersome for an error-avoiding learner. If conditioning by unsupported defaults should prove to be essential for parameter setting by a deterministic learner, one or other of these consequences would have to be endured or else the deterministic learning hypothesis would have to be abandoned. For metrical phonology, DK/D did contemplate the use of default values as conditioners, with retrenchment when necessary, though they considered other options as well.³⁴ For our syntax domain, we are able to maintain the general proscription against defaults as conditioners because, as we now explain, there is another way to handle the case of NullTop/OptTop (and no other cases arise in the CoLAG domain).

The relationship between NullTop and OptTop is a special one, which removes the risk from using -NullTop to provide triggers for +OptTop, even if conditioning by unsupported default values is not generally condoned. This is because the necessary retrenchment from +OpTop to -OptTop would occur reliably and without need for any sort of record-keeping, if and when the conditioning value -NullTop is later reset to +NullTop. The between-parameter constraint that makes +NullTop incompatible with +OptTop (see Table 2 above) means that the learner's adoption of +NullTop will *automatically* reset +OptTop back to -OptTop. The learning mechanism does not need to attend to it.

While the mechanism of conditioning could thus be hedged in ways that permit default conditioners in special circumstances that render them safe, such as automatic retrenchment, the CoLAG domain does not in fact require anything more than the basic principle that only

³⁴DK/D's original discussion of these matters is of interest. They clearly perceive the risk of employing default values as conditioners in a deterministic learning model, but find it difficult to do without them in the phonological domain and outline several ways of dealing with this tension:

⁽i) Permit defaults as conditioners only where the default has been confirmed by unambiguous triggers. (DK/D do not assess whether this would admit sufficient conditioners to set other needy parameters.)

 ⁽ii) Permit defaults as conditioners, but keep tabs on which parameters are set on that basis, so that those parameters can be returned to their starting values if the conditioning parameter is later set forward to its nondefault value. (Note that this undermines determinism.)

⁽iii) Permit defaults as conditioners and give up incremental learning. Batch mode (storing all inputs before committing to any parameter settings) provides a kind of negative evidence that could remedy overgeneration errors resulting from use of the default conditioner. (Note that determinism and nondeterminism are not in sharp contrast in a batch learning model.)

While DK/D seem drawn toward (ii) in their theoretical discussions, their computer implementation clearly engages in nonincremental learning as in (iii).

input-supported parameter values may condition other parameter values, allowing for a more constrained theory of triggers. This is because, once the complementarity between +NullTop and +OptTop is taken into account, the relationship between them need not be regarded as conditioning at all, but can be seen simply as a *between-parameter default*, as explained below. (N.B. a between-parameter default is not the same as a between-parameter constraint; the difference will be clarified in the next subsection.) It should be noted that a between-parameter default cannot substitute for conditioning in the general case. It cannot, for example, substitute for conditioning by +TopMark, or for other instances of conditioning presented below. In section 3.1, with examples in hand, we will define and formally distinguish conditioning and between-parameter defaults; the two are not equivalent.

2.2.5. The Solution: A Between-Parameter Default

We now show how a between-parameter default allows the NullTop parameter to assist the OptTop parameter in finding triggers. The proposal to be considered is that there is a default relation between the OptTop and NullTop parameters that gives +OptTop priority over +NullTop in response to any input sentence that is ambiguous between an analysis with no topic and an analysis with a null topic. Given such a sentence, the learning mechanism will adopt +OptTop (which is in need of triggers) rather than +NullTop (which has a sufficiency of other triggers; see below). Let us track through what the learner's experience would be in such a case.

The learner would start with both parameters at their own default values: -NullTop and -OptTop, i.e., the beginning learner presupposes that every declarative sentence has an obligatory overt sentence-initial topic. If the target grammar has these values, the learner would never encounter a sentence requiring that one or the other of these parameters be set to its marked value. However, for other target languages the learner would hear a sentence that is not licensed by the default values and that may be ambiguous between a null topic and no topic (e.g., Verb S). Rather than discarding this ambiguous trigger, the between-parameter default imposes a disambiguation in favor of +OptTop. So now the learner's grammar would have -NullTop,+OptTop. If the target grammar does in fact have +OptTop, the settings would thereafter remain at -NullTop,+OptTop permanently. If instead the target values are +NullTop,-OptTop, the learning mechanism would at some point encounter an unambiguous trigger for +NullTop. That would reset the grammar from -NullTop to +NullTop, and si*multaneously* from +OptTop to -OptTop as a result of the between-parameter constraint.³⁵ Thereafter, the settings +NullTop,-OptTop would remain in place permanently, because the input sample for a +NullTop,-OptTop target language would contain no sentences requiring +OptTop. Hence, there is no risk of a repeated flip-flop between -NullTop,+OptTop and +NullTop,-OptTop. All that is needed for this scenario to succeed is that unambiguous triggers for +NullTop are available in all +NullTop languages, and this is so.

 $^{^{35}}$ Note that OptTop is reset twice: from default –OptTop to +OptTop, and then back to –OptTop when +NullTop is later set. This need not be ruled out in a deterministic system, because the second shift is not a triggering but an entailment from another parameter value. Note also that this prioritization of one parameter over another is not identical to an obligatory order of parameter setting, since it is perfectly possible for NullTop to be set to its marked value before OptTop is, if the input happens to offer the learner an unambiguous trigger for +NullTop before it offers any inputs ambiguous between the two.

FINDING 8: Every +NullTop language in CoLAG has unambiguous E-triggers, consisting of the absence from the surface word string of an obligatory complement.

EXPLANATION: The CoLAG UG permits an indirect object (O2) only in sentences containing a direct object (O1). Also, all CoLAG languages have adpositions but no particles (or they have transitive but not intransitive adpositions), so a sentence that contains P must also contain its object (O3). Thus, the surface absence of O1 when O2 is present, or of O3 when P is present, reveals the presence of a phonologically null O1 or O3 respectively as in examples (16) and (17). These are unambiguously associated with +NullTop because the only null constituents in CoLAG other than null subjects are null topics.

- (16) S Aux Verb O2
- (17) Verb O1 O2 P Adv S

To sum up the discussion of Findings 6, 7, and 8, for +OptTop the dearth of unambiguous triggers can be solved if the learner, on encountering a word string that is ambiguous between +NullTop and +OptTop, *treats* it as a trigger for +OptTop. This amounts to stipulating a particular disambiguation of the ambiguous trigger, in favor of the parameter that has too few unambiguous triggers of its own. This stipulation can be overridden where necessary by an unambiguous +NullTop trigger, available in every +NullTop language. In this way, the between-parameter default ensures that both NullTop and OptTop (the latter with the assistance of +TopMark conditioning) ultimately arrive at their correct values. Thus, for the first row of Table 2 above, the parameter settings are achieved by default values of OptTop and NullSubj and a globally valid trigger for +NullTop; the second row requires no resetting of the defaults at all; it is the third and fourth rows that are achieved by means of the between-parameter default.

The theme of this work is that the disambiguation of what would otherwise be ambiguous triggers is the key to how a deterministic learner can learn safely from ambiguous input. Although not discussed as such in the literature, trigger disambiguation is in fact a familiar device in the guise of standard within-parameter defaults. The role of a within-parameter default is precisely to disambiguate an input compatible with both values of a parameter, in favor of one of them (often, a subset value; as in Finding 9).³⁶ Between-parameter defaults and conditioned triggers are simply extensions of this basic idea. As we have demonstrated here, these extensions are essential for a deterministic learner to succeed in the CoLAG domain.

Could any parameter at all be provided with unambiguous triggers by this means, trivializing the debate about deterministic learning? To the contrary, imposing a disambiguation on ambiguous input is not a general panacea. It is appropriate (i.e., safe) only when certain criteria are satisfied that ensure that there will be no untoward consequences. In the present case, the fact that +NullTop and +OptTop are in complementary distribution in CoLAG provides this

³⁶Disambiguation of potentially ambiguous triggers is also the role of *designated triggers* in the sense of Fodor (1994). These are sentence types that are ambiguous between parametric triggers and peripheral constructions, but it is proposed that they are innately stipulated as the former. Otherwise, on the assumption that no sentence is inherently barred from peripheral status, SP would entail that no parameters are ever set, since SP would always disambiguate in favor of the peripheral construction.

insurance: the learner does not end up with an overpowerful grammar with the marked values of both parameters. Further examples of safeguards are mentioned below, and we formulate the safety criteria in section 3.1. Where they are not met, it remains essential that a deterministic learner should reject ambiguous triggers. Hence it is still an empirical issue, not a foregone conclusion, whether sufficient unambiguous/disambiguated triggers exist to set every parameter.

2.2.6. Optional Topic and the Subset Principle

A deterministic learner needs to know which of two parameters should take priority when they are in competition for the same input. Without this knowledge incurable errors could occur. Assuming that the relevant considerations (as in the Explanations for OptTop above) are too complex for a learner to be able to *deduce* the correct prioritization, it must be innately encoded in some fashion. However, it may not be necessary to stipulate such priorities case by case, if they follow from some general principle such as the Subset Principle (SP) (and if the learning mechanism is able to determine that they do; see Fodor and Sakas 2005). We now show that in CoLAG, OptTop, and NullTop do stand in a partial subset-superset relation.

FINDING 9: A domain search showed that 180 –NullTop,+OptTop languages are proper subsets of the corresponding languages with +NullTop,-OptTop (i.e., when all other parameters are held constant). (This is out of a total of 768 possible cases: there are 1,536 –NullTop, +OptTop languages but only 786 +NullTop,-OptTop languages.) These subset languages are all –TM, and either Head-final or Subject-initial or both.

EXPLANATION: There are two points to address. Why doesn't a subset relation hold between *all* –NullTop,+OptTop and +NullTop,-OptTop languages? And why does it hold when it does? The answer to the first question is that though +NullTop and +OptTop have largely similar surface consequences, each is able to license some word strings that the other cannot; see Figure 3. +OptTop can license full-house sentences with an initial non-topicalizable item,



FIGURE 3 Illustration of how a subset relation between +OptTop and +NullTop languages arises depending on the values of other parameters—in this case, the Subject Position parameter. In Figure 3a, the Subject-final +OptTop and +NullTop languages both have distinguishing sentences, exemplified by s_1 and s_3 respectively, though they also share many sentences (e.g., s_2). In Figure 3b, there are no distinguishing sentences for the +OptTop language (i.e., no sentences within the dotted line) because subject-initial full-house sentences (e.g., s_4) occur in both +OptTop and +NullTop languages.

which +NullTop cannot license (the very sentences rejected above as triggers for +OptTop, e.g., *Verb O1 O2 P O3 Adv S* in Figure 3a). +NullTop can license sentences in which an obligatory complement is not overt, as already noted (e.g., *Verb O1 O2 P Adv S* in Figure 3a). The answer to the second question is that some +OptTop languages (those that are subjectinitial and/or verb-final) do not have the sentences that distinguish +OptTop from +NullTop. A full-house subject-initial sentence must begin with a subject; recall from Finding 2 that all languages under consideration are -NullSubj. The subject might or might not be in topic position, i.e., topicalization is string-vacuous in this case. If the subject is parsed as a topic, such a sentence satisfies the constraint that a topic is obligatory in a +NullTop language, so it is compatible with +NullTop as well as with +OptTop; see sentence s_4 in Figure 3b. A similar point holds for full-house verb-final sentences; if not subject-initial these must begin with *Adv*, which might or might not be in topic position. Thus, in these two pockets of +OptTop languages, a +OptTop language has no distinctive sentences of its own, so it is a proper subset of the corresponding +NullTop language. In those cases, SP is applicable and it demands that +OptTop take priority over +NullTop.

There are theoretical options here that deserve further investigation in the future. One possibility is that a SP-motivated default should be restricted to *only* the actual subset-superset cases, where it is essential.³⁷ However, the theory of triggers and its innate basis would be simpler if it excluded the possibility that a default relation between two parameter values could vary depending on other properties of the grammars involved. In the CoLAG domain we have found no situations that require context-sensitivity of parameter defaults.³⁸ So we hold here to the tighter theory that entails that +OptTop be treated as the default relative to +NullTop across the board.

To conclude this discussion: As observed, unambiguous triggers can be *created* rather than found, by imposing a disambiguation on input that is in fact ambiguous between two (or more) grammars (i.e., is compatible with both). SP is a very familiar and very general means of trigger disambiguation. What OptTop shows is that this kind of disambiguation can be needed *between* parameters, not merely within individual parameters. It is impossible to find unambiguous triggers for all parameters (in the CoLAG domain) unless parameters are permitted to depend on each other, via between-parameter defaults or via conditioning. Such relationships are counterexamples to the Simple Defaults Model (Fodor & Sakas 2005), which is often presupposed (see for example Manzini & Wexler 1987:434–435) and which holds that every default relation needed to satisfy SP can be captured by assigning a default value within a single parameter.

2.3. Triggers for Verb Movement Parameters

We now tackle the two other parameters in CoLAG that are short of triggers for both of their values (see Table 1 above). These are the parameters that control movement into C: the ItoC

 $^{^{37}}$ It may be that in the natural language domain, unlike the CoLAG domain, *every* +OptTop, -NullTop language is a proper subset of the corresponding -OptTop, +NullTop language, if indeed the notion of full-house sentences is not well-defined.

³⁸DK (1990:162–163) proposed what we would call a *conditioned default*, where the setting of one parameter determines which value of another parameter is its default. This is an interesting notion though not one that we found need for our study.

Movement parameter (movement from I to C, none versus obligatory; hereafter *ItoC*), and the Q-Inversion parameter (question inversion, none versus obligatory; hereafter *QInv*). The latter is identical to ItoC movement except that it applies only in questions, while ItoC movement applies in both questions and declaratives. In both cases what moves is a finite verb, either *Aux* if that is present in the sentence, or Verb[+FIN] otherwise. Note that CoLAG does not have a split I, and C is the only head above I. Since head movement is local, movement into C is only permitted from I, hence only a finite verb (*Aux* or Verb[+FIN]) can move to C, and C is the only place a finite verb in I could move to. We exclude imperative sentences from discussion throughout this section, since they lack finiteness and verb movement in CoLAG; indeed, imperatives in CoLAG are immune to all parametric variation except headedness.

The two parameter values +ItoC and +QInv overlap considerably in their effects. While there is no systematic subset-superset relation between the languages they license, many of their potential triggers are ambiguous—which is why we had thought it would be an interesting challenge to include these parameters in the CoLAG domain. They also interact significantly with the parameters for V-to-I movement, headedness in CP, headedness in IP and VP, and affix hopping. We now document where and why our search procedure detected an insufficiency of unambiguous triggers for movement to C, and we examine whether despite this there are ways in which the learning mechanism could reliably establish the correct settings. We will consider QInv first, then ItoC and its relation to other parameters. Note that in Table 1, nonmovement is assigned as the default value for both of these parameters. Based on the linguistic-theoretic assumption that movement requires richer syntactic licensing than nonmovement (see, for example, Roberts, 2001), we ruled that for all movement parameters in CoLAG, the nonmovement value should be the default. However, as noted, we had cause to rethink that decision as we evaluate the domain facts.

2.3.1. Triggers for Movement to C

FINDING 10: Half of the languages with –Qinv have globally valid triggers for that value. All contain ka.

EXPLANATION: With –Qinv as the default, no triggers for it are needed but some exist nonetheless. The interrogative complementizer ka provides a clear morpholexical marker in the CoLAG domain showing that no verb has moved into C (where lower-case 'verb' from here on includes both *Verb* and *Aux*). The CoLAG UG determines that ka surfaces in C in questions just in case the C position is not preempted by any other element. This complementary distribution between movement and morphololexical marking is not uncharacteristic of natural languages (as in the suppression of verb movement in German subordinate clauses with $da\beta$). We included ka in the domain in order to explore the case of a parameter whose triggers are of different kinds (morpholexical versus syntactic) in different languages and to assess the relative efficiency of morpholexical versus syntactic triggers.

Movement into C is assumed to be obligatory wherever it is possible; hence any question containing ka unambiguously establishes both –QInv and –ItoC. Languages with this combination of values constitute half of the –Qinv languages (and also half of the –ItoC languages, relevant below), accounting for the positive aspect of Finding 10. The negative aspect of Finding 10 for QInv (that half of the languages with –Qinv lack any globally valid

triggers) is due to the fact that the other half of -Qinv languages have +ItoC, which overtakes and hides any effect of +/-QInv. The QInv parameter is always irrelevant in the presence of +ItoC. This is why there are no unambiguous triggers for -QInv in such languages, but it also means that -QInv does not need any triggers in those languages.³⁹

So far we have seen that the -QInv setting has sufficient triggers when in a grammar with -ItoC and needs no triggers when in a grammar with +ItoC. So it is unproblematic across the board. But in any case, with -Qinv as default, it is only +QInv that needs triggers.

2.3.1.1. A between-parameter default for QInv and ItoC.

FINDING 11: +QInv has no globally valid triggers.

EXPLANATION: Absence of ka in a question could be partially informative: it reveals that the language has either +QInv or +ItoC or both. But it is ambiguous between these alternatives, and hence for a deterministic triggering learner it does not qualify as a valid trigger for either one. In order to set either +QInv or +ItoC it seems that languages without ka in questions need some other kind of evidence: nonmorphological evidence such as word order. For +QInv, however, there is no hope of such evidence because +QInv will always be in the shadow of +ItoC. Any word order evidence for verb movement into C in a declarative would unambiguously set +ItoC, but word order evidence of movement to C in a question would be ambiguous between the marked values of the two parameters; thus, there is no pattern that is distinctive for +QInv.

Therefore this is another case that can be rescued only by imposing a disambiguation on the ambiguous triggers. A between-parameter default will do the job. Clearly it is +QInv that needs assistance, so +QInv must be given priority over +ItoC when the input evidence is ambiguous between them. Then +QInv does not need any specific triggers of its own. If the target language in fact has +ItoC, that could be established by evidence of movement to C in declaratives. (In that case, depending on whether the evidence of movement is first encountered in declaratives or in questions, the learner might end up with the values +ItoC,-Qinv or +ItoC,+QInv, but these languages are weakly equivalent so either outcome is satisfactory.) All that is required for this disambiguation strategy to succeed is that the nonpriority value +ItoC should have enough triggers to establish it in all +ItoC languages. Therefore attention turns now to the ItoC parameter. In what follows we will see that it does not rise to the occasion: +ItoC has unambiguous triggers in some languages but not in all. We offer a potential solution in section 2.3.2 below.

We will now focus on declarative sentences, which must be the source of any triggers that distinguish +ItoC from +QInv.

FINDING 12: Both +ItoC and -ItoC have globally valid triggers in only some of the languages with those values: 1,152 out of the 1,536 languages for +ItoC, and 1,280 out of the 1,536 languages with -ItoC.

³⁹Parameter irrelevance entails both the absence of triggers and that triggers are not needed. But a caution: the absence of triggers does not entail that triggers are not needed. This is because languages may differ from each other (no weak equivalence) in ways that do not qualify as triggers, at least for an incremental deterministic learning system. Some examples arise in this section (e.g., Finding 14). See general discussion in section 3.1.

EXPLANATION: Movement from I to C is string-vacuous in some contexts, making it impossible to detect in the surface word string (as for topicalization in section 2.2). This deprives both the movement value and the nonmovement value of unambiguous triggers. (Despite this symmetry, the + and - values of ItoC differ in their proportions of unambiguous triggers for other reasons, noted below.) With -ItoC as the default, this poses a potential learning problem only for +ItoC. We now review the movement facts in more detail, with the aim of understanding the triggers that do exist, and applying that insight to resolve cases where at first glance they do not. We need to know which languages lack +ItoC triggers, and whether they are languages that need triggers (i.e., the ItoC parameter is not irrelevant); if so, is there any kind of disambiguation strategy that could liberate triggers for them?

In the absence of morphological evidence of movement to C in declaratives, any triggers for +ItoC must be syntactic, presumably a matter of surface word order. We therefore sorted languages by their word order, as shown in Table 3, in order to see whether one particular word order is responsible for the incomplete coverage of triggers for +ItoC. Each "surface word order" cell in the table contains 192 languages. For convenience in the table and associated discussion, we use an abbreviated notation for the various word orders; for example, CSIVO denotes the underlying sequence of an initial C (always surface null in CoLAG declaratives except when a verb has moved into it), followed by the subject, followed by I (which might or might not underlyingly dominate Aux^{40}), followed optionally by negative elements that are not represented in this abbreviated format, followed by the main verb with its complements, the latter represented generically here as O. For uniformity we include all of these elements in all the representations of surface order; those that are relevant for triggering the parameter are shown in bold font. The fact that some languages can have null subjects or topics is not represented in Table 3 because those languages all contain sentences with overt subjects and topics respectively.

Table 3 shows that the cases lacking triggers are not scattered but are systematically associated with certain underlying word order types: IVOSC and CSOVI for -ItoC, and CIVOS and SOVIC for both + and - values. With -ItoC as the default, only the latter two word orders present potential learnability problems. We therefore ran a targeted search on the languages with those word orders (the cells marked X in the +ItoC column of Table 3).

2.3.1.2. Focus on the troublesome word orders.

FINDING 13: None of the +ItoC CIVOS and SOVIC languages contain globally valid triggers for +ItoC.

EXPLANATION: Since movement results in the separation of two items that were underlyingly adjacent, it is detectable if there is some intervening item that serves as a landmark; otherwise the movement is string-vacuous. Consider the CSIVO languages: an overt subject that is underlyingly between I and C can serve as a landmark revealing whether or not an auxiliary

 $^{^{40}}$ For space reasons, word order patterns without an auxiliary are not shown in the table. For them, movement to C is revealed by intervention of the subject between the *Verb* and its complements, or by *Verb* position incompatible with VP-headedness as revealed by the order of unmoved complements. These triggers occur in fewer languages than the triggers with *Aux*, but that is of no consequence because every language in CoLAG has sentences containing an auxiliary.

	Surface Word Order		
Underlying word order	-ItoC	+ItoC	
CSIVO	✓ XSAuxVO	✓ AuxSVO	
OVISC	✓ OVAuxS	✓ OVSAux	
IVOSC	✗ AuxVOS	✓ VOSAux	
CSOVI	✗ SOVAux	✓ AuxSOV	
SIVOC	✓ XSAuxVO	🖌 SVOAux	
COVIS	✓ OVAuxS	✓ AuxOVS	
CIVOS	✗ AuxVOS	✗ AuxVOS	
SOVIC	✗ SOVAux	✗ SOVAux	

TABLE 3 Eight Underlying Word Order Patterns for Declarative Sentences with an Auxiliary Verb, with Their Associated Surface Word Orders

Note. Surface orders marked \checkmark constitute globally valid (unambiguous) locally available syntactic E-triggers for the indicated value of ItoC; boldface is used to show those elements relevant to triggering. Surface orders marked \checkmark do not constitute unambiguous triggers for the indicated value of ItoC. X indicates that the topic position must be filled for the trigger to be unambiguous.

has moved up out of I. The surface order AuxSVO shows that it has; the surface order XSAuxVO shows that it has not. In languages of the COVIS type, objects can serve as landmarks; the surface order without movement would be OVAuxS; the surface order AuxOVS shows that the auxiliary has moved. For the CIVOS and SOVIC languages, by contrast, I and C are linearly adjacent underlyingly, so there is no element that can intervene between them in the surface form, which means that movement from I to C is not unambiguously detectable in these languages.⁴¹ (It should be noted that CoLAG has no adverbial landmark between I and C; see discussion in section 2.3.3.1 below). These +ItoC languages with CIVOS and SOVIC word orders thus lack both morpholexical (ka) and word order triggers for +ItoC. Note that from this point, for simplicity, we will report results only for the CIVOS languages; the results for SOVIC languages are parallel in all respects.

FINDING 14: ItoC is irrelevant to 64 of the 192 +ItoC CIVOS languages. The languages for which ItoC is irrelevant have +QInv and -AfHop (no affix hopping).

EXPLANATION: We have seen that the CIVOS +ItoC and -ItoC languages have the same surface order of major sentence constituents. If there are no other differences between a +ItoC

⁴¹Bertolo et al. (1997b) noted a related problem for languages that have what we are calling CIVOS word order. In their case, an adverb at the upper edge of VP gave evidence that the verb had moved up out of the VP, thus requiring that some verb-raising parameter must be set. But the artificial language domain they were considering provided no higher landmarks to show whether the verb had moved up to the next head position (Agr) or to the one above that (T), creating an ambiguity that the input could not resolve. However, this does not constitute a learning problem except on the assumption (which Bertolo et al. made) that head movement need not be local, so that the verb could move either to Agr, or directly to T without passing through Agr. Given locality, the input evidence of verb raising out of VP would *unambiguously* trigger movement to Agr. Given the absence of higher landmarks, there would be no input evidence for further movement to T.

language and its corresponding –ItoC language, they will therefore be weakly equivalent and the ItoC parameter will be irrelevant. This is true in some cases—the languages with both +QInv and –AfHop, and for them no triggers are necessary. (Since weak equivalence is symmetric, the ItoC parameter is irrelevant to the same number of –ItoC CIVOS languages; this will be pertinent in section 2.3.2.) However, the interplay between ItoC and QInv blocks irrelevance of ItoC in other cases. In languages with –QInv, the ItoC parameter is relevant because its values are differentiated by the presence of ka in –ItoC languages versus its absence in +ItoC languages. These languages therefore do need triggers for +ItoC but none have been identified so far since, as noted, the absence of ka does not constitute an unambiguous trigger for +ItoC. (The role of the affix hopping parameter will be explained below in discussion of Finding 16.)

Since no globally valid triggers exist for +ItoC in the CIVOS (and SOVIC) languages for which it is relevant, we continued the search by looking for conditioned (locally valid) triggers for +ItoC.

FINDING 15: For +ItoC in languages with CIVOS word order, no conditioned triggers were found, either 1-conditioned or multiply-conditioned. (Likewise for the SOVIC languages.)

EXPLANATION: To summarize what we have discovered: The sentences of the CIVOS (and SOVIC) languages have no unique positive properties that could be used to trigger +ItoC, neither morpholexical properties like ka, nor syntactic properties such as relations to a word order landmark. The fact that the value of ItoC nevertheless makes a difference to some of those languages means that this cannot be dismissed on grounds of irrelevance. In many cases the +ItoC CIVOS and SOVIC languages do differ from the corresponding -ItoC languages, but they do so only in negative respects, which on our assumptions do not qualify as triggers. Finding 15 shows that no other parameter values are able to convert ambiguous triggers to unambiguous ones for +ItoC (as +TopMark did for +OptTop, see Finding 5). The data thus imply that the stock of +ItoC triggers is simply inadequate. The usual logic of learnability in a triggering system then entails that +ItoC cannot be the marked value. Conceivably, making +ItoC the default value might save the day, even though the default would then be movement, contrary to our original working assumption. We bite this bullet in section 2.3.3 below. But first, since the nonmovement value would then have to serve as the marked value, it is necessary to check whether -- ItoC has sufficient triggers. This cannot be taken for granted. Table 1 shows that in some CoLAG languages -ItoC lacks globally valid triggers, so if indeed those languages are learnable, -ItoC will stand in need of assistance from one or more disambiguation strategies.

2.3.2. Triggers for Nonmovement

Some of the findings already presented take on a different significance as the search shifts now to finding triggers for -ItoC and -Qinv rather than for their positive values. Table 3 shows that even fewer underlying word orders have unambiguous surface word order triggers for -ItoC than for +ItoC. However, we have observed one respect in which -ItoC is better equipped with triggers than +ItoC is. As noted under Finding 10, -ItoC has *ka* triggers for

half of the languages with that value (the half that are –Qinv, across all word orders), while +ItoC has no morphological triggers.

The following findings show that disambiguation strategies provide –ItoC with unambiguous triggers in all languages that need them.

2.3.2.1. Triggers for – ItoC in CIVOS and SOVIC languages.

FINDING 16: In 64 of the 192 –ItoC CIVOS languages (likewise for SOVIC), domain search found declarative sentences that unambiguously require –ItoC. These sentences all contain the adverb *never*, a finite *Verb*, and one or more noninitial complements of the verb. There are no such sentences in the corresponding +ItoC languages. The –ItoC languages that have these triggers are all +AfHop and –VtoI (no VtoI movement).

EXPLANATION: Why does a sentence such as (18), with a negative adverb, a finite verb, and a direct object entail –ItoC?

(18) Never Verb[+FIN] O1 S

The differentiating role of these *never* sentences has a natural explanation. The fact that *never* is negative is not important; what matters is that it is an adverb located just above the VP (in Spec, NegP). Its role is exactly like that of adverbs such as *souvent* and *often* in the work of Pollock (1989), who used them to demarcate the upper edge of VP, distinguishing languages like French in which a finite verb has moved out of VP into an inflectional head position, from languages like English in which the verb remains in VP. In CoLAG we included the lexical item never as representative of this class of adverbs. If a learner can recognize that a finite verb is lower than this adverb, a direct inference would be that the verb has not raised out of VP into I. Thus, the sentence would trigger -VtoI. Since the verb, still in VP, is finite, the sentence would also trigger +AfHop because affix hopping is the only means by which a verb inside VP could acquire finiteness. Secondarily, such sentences prove that nothing (neither the Verb nor Aux) has moved into C. Since a verb cannot move to C except via I, it follows from -VtoI that the Verb has not moved to C. The alternative would be for Aux to move to C, but that is excluded by the fact that the Verb in these sentences is finite, which is possible only if the sentence contains no Aux. Thus in a sentence in which Verb[+FIN] is detectably beneath *never*, neither of the elements that could move to C has done so.⁴² Since movement to C is obligatory where permitted, the language must be -ItoC.

What is the role of the object of the verb in a trigger such as (18)? It is needed to reconcile the fact that the real trigger (the I-trigger) for raising or nonraising of the verb is a tree configuration property (the finite verb is *higher* or *lower* than the VP-edge adverb), with the fact that an E-trigger has to be cast in *linear* terms if it is to be surface-detectable. The linear order of Verb[+FIN] and *never* will show that the verb is to the left or right of *never*, but not that it is above or below *never*. However, there are some usable correspondences. In a CIVOS

 $^{^{42}}$ As an alternative to the VP-edge adverb with *Verb[+FIN]* trapped beneath it, it might have been possible to assume that learners can recognize a dummy (semantically empty) auxiliary and infer from it that the verb is unable to move to I. But the CoLAG domain, which (regrettably) lacks semantics, is unable to represent the distinction between contentful and dummy items. It encodes the *do* of English-style do-support as an auxiliary like any other.

(head-initial) language, a verb to the left of *never* is above it and a verb to the right of *never* is below it; in a SOVIC (head-final) language, the converse is true. This is because in CoLAG, *never* is the left daughter of NegP in head-initial languages and the right daughter of NegP in head-final languages (i.e., it is universally on the opposite side of the negative head from its complement VP). This dependence of the E-trigger on head-position is captured by the inclusion of an unmoved complement whose position in relation to the verb reveals headedness. Thus in (18) above, the *O1* following the *Verb* shows that the language is head-initial and therefore that the *Verb* to the right of *never* is beneath *never*. Thus (18) is an E-trigger for –VtoI and indirectly for –ItoC. In a SOVIC language an example of an E-trigger for –VtoI and thereby for –ItoC is sentence (19). Note that it is essential in this case that some element precedes the *O1*, to exclude the possibility that *O1* has been topicalized from a position to the right of the *Verb*. Thus, the general schema for these E-triggers requires *never*, *Verb[+FIN]*, and at least one *noninitial* complement of the verb.

(19) S O1 Verb[+FIN] never

Note that these 'portmanteau' triggers for -VtoI, which fold a trigger for headedness into the trigger for verb position, have conceptually tidier counterparts as conditioned triggers for -VtoI, in which the trigger for verb position includes only Verb[+FIN] and the VP-edge adverb, and their left-right order is explicitly conditioned by the value of the Headedness parameter.⁴³ For convenience, from now on we will refer to these triggers as VP-edge triggers.

These VP-edge triggers for –ItoC are only locally available. As noted, they occur only in +AfHop languages because a VP can contain a finite verb only via affix hopping. In –AfHop languages there are no comparable triggers for –ItoC because a VP-edge adverb does not differentiate –ItoC from +ItoC. There are two cases. If a –AfHop language is –VtoI, no main verb can be finite, so every (nonimperative) sentence must contain an auxiliary, which appears above a VP-edge adverb in both +ItoC and –ItoC languages. If a –AfHop language is +VtoI, a finite main verb can appear above the adverb in both +ItoC and –ItoC languages. However, although the VP-edge triggers for –ItoC are only locally available, they are globally valid: when encountered they always correctly signal –ItoC.

Taking stock at this point: Of the 192 –ItoC CIVOS languages, –ItoC is irrelevant to 64 (Finding 14). Ninety-six have *ka* triggers⁴⁴ and 64 have VP-edge triggers (Finding 16), but it happens that there is an overlap of 32 between these cases (i.e., 32 languages have both *ka* triggers and VP-edge triggers), so between them they cover just 128 languages. Thus all 192 (= 64 + 128) CIVOS languages are accounted for (and likewise for the 192 SOVIC languages). We now turn to one further finding that does not add or subtract from the stock of unambiguous triggers for –ItoC, but underscores the fact that +ItoC *must* be the default value of the ItoC parameter in CoLAG.

⁴³Using the Head-initial value of the Headedness parameter as a conditioner does not violate the principle that a conditioning parameter value must have reliable triggers of its own. Although the Head-initial value is listed as the default in Table 1, this is merely an arbitrary assignment for convenience here; the Head-initial value has sufficient triggers. A similar point applies to the Headedness in CP parameter discussed in Findings 19 and 20 below.

⁴⁴See Explanation of Finding 10. One half of the -ItoC languages have *ka* triggers. This holds for all eight word order patterns, so 96 of the 192 -ItoC CIVOS languages have *ka* triggers.

FINDING 17: 32 of the -ItoC CIVOS languages are proper supersets of the corresponding +ItoC languages. Both the subset and the superset languages are all -VtoI and +QInv, and the superset languages are all +AfHop.

EXPLANATION: The 32 superset languages are one half of the –ItoC CIVOS languages that have VP-edge triggers (which presuppose –VtoI and +AfHop, as above). Their +ItoC subsets do not have these triggers, and a computer search found none of the properties that elsewhere distinguish +ItoC languages from their –ItoC counterparts. Because they have CIVOS word order, there are no simple surface word order triggers that distinguish +ItoC from –ItoC. Because the subsets are all +QInv, their questions lack *ka* regardless of the value of ItoC. Furthermore, their combination of –VtoI with +ItoC entails that every sentence must contain an auxiliary, which raises from I to C, because the main verb cannot do so. Their counterpart superset languages with –ItoC also have sentences containing an auxiliary, but because they are +AfHop they have sentences without an auxiliary as well; the finite main verb remains inside VP and nothing needs to raise to C. These sentences without auxiliaries are what make the –ItoC languages supersets of the corresponding +ItoC languages.

For these +ItoC languages (if indeed they are acquirable), SP demands that +ItoC be the default value. They are only a small minority (just 4.2% of each of the CIVOS and SOVIC language groups), but given our working assumption above (section 2.1) that a default relation between parameter values must remain constant, it follows that +ItoC should be the default in *all* cases. Thus the switch from -ItoC to +ItoC as the default, which we made reluctantly, has empirical justification in the CoLAG domain.

2.3.2.2. Triggers for -ItoC in IVOSC and CSOVI languages. Whether the CoLAG domain contains sufficient unambiguous triggers for the ItoC parameter now rests on the IVOSC/CSOVI group of languages, which constitute the other potential weak spot for -ItoC, as Table 3 shows. At first sight, -ItoC is at a disadvantage for these languages compared with +ItoC. (The other four word order types in Table 3 have unambiguous triggers for both values since they have subject and object landmarks that clearly distinguish between movement and nonmovement into C.) To simplify exposition, we focus on findings for IVOSC word order; the findings and explanations for CSOVI languages are parallel.

FINDING 18: Of the 192 languages with underlying word order IVOSC, domain search found only 128 with globally valid triggers for –ItoC: 96 languages have *ka* triggers; 32 have VP-edge triggers.

EXPLANATION: C position is not overtly lexicalized in declarative sentences in –ItoC languages in CoLAG, so C position is invisible in those sentences. Therefore, the surface order of the major sentence constituents in –ItoC IVOSC languages is the same as in CIVOS languages with either +ItoC or –ItoC; i.e., AuxVOS as shown in Table 3 (or VOS in sentences with no Aux). Thus there are no reliable word order triggers for –ItoC in the IVOSC languages, and the 64 –ItoC IVOSC languages (= 92 – 128) which do not have ka triggers or VP-edge triggers have no globally valid triggers at all. However, although AuxVOS is not a globally valid trigger for –ItoC, it may be rescuable as a locally valid (conditioned) trigger for –ItoC in the IVOSC languages, a possibility we now turn to.

FINDING 19: All 192 –ItoC languages with underlying word order IVOSC have locally valid triggers for –ItoC conditioned by the Complementizer-final value of the Headedness in CP parameter. (For underlying word order CSOVI, the search found locally valid triggers conditioned by the Complementizer-initial value.)

EXPLANATION: In IVOSC (and CSOVI) languages, the Headedness in CP parameter pins down C in a position not underlyingly adjacent to the finite verb, but in fact at the opposite end of the word string. So if C position has been established, there are abundant landmarks (subject, objects) to reveal movement or nonmovement of the finite verb to C. These would suffice to set –ItoC in all the –ItoC IVOSC (and CSOVI) languages—as long as the Headedness in CP parameter is acceptable as a conditioner. Finding 20 indicates that this is so.

FINDING 20: The Headedness in CP parameter has unambiguous triggers for Complementizerinitial and for Complementizer-final in every language for which those values are relevant.

EXPLANATION: Though Complementizer-initial is designated the default value in Table 1, this is just a stipulation for convenience; there is no asymmetry in the supply of triggers for the values of this parameter. Some languages have declarative triggers for Headedness in CP in which intervening landmarks reveal movement. See example (20), where the *Aux* has moved away from the *Verb* and is in final position. Since the only position the *Aux* could have moved to is C, a language containing (20) must be Complementizer-final. This is evidently an IVOSC language, with movement to C. The corresponding sentence in an IVOSC language with the nonmovement value of ItoC would be (21), which is not informative about the position of C (or therefore about whether there has been movement to C).

- (20) Verb O1 S Aux [ILLOC DEC]
- (21) Aux Verb O1 S [ILLOC DEC]

However, in all languages in CoLAG, the position of C is clearly revealed in questions, with or without landmarks revealing movement.

In questions, C position is occupied either by the interrogative complementizer ka, or else by the finite verb (Aux or Verb) that blocks ka from surfacing. In either case the position of C is recognizable. The ka triggers are easily identified (and they set not only C position but also –QInv and –ItoC, as in Finding 10 above). But a ka-less interrogative can also reveal the position of C, via the position of its finite verb. In sentence (22), from a CIVOS language, the Aux remains adjacent to the Verb but it must have moved up into C because of the absence of ka. This establishes that C is sentence-initial.

(22) Aux Verb O1 S [ILLOC Q]

In an IVOSC language, a *ka*-less question such as (23) entails that the *Aux* must be in C, and hence shows that C is final, though note that in this case there is also movement evidence for the position of C. A language containing (23) must be an IVOSC language with either +QInv or +ItoC, and Complementizer-final.

(23) Verb O1 S Aux [ILLOC Q]

It is of interest here that, although questions cannot speak directly to the value of the ItoC parameter (except where *ka* is present; Finding 10), they can provide crucial information about C position, which *indirectly* contributes to setting the ItoC parameter for declaratives. Suppose, for example, that Complementizer-final has been established for a language, such as one of the IVOSC languages, through an interrogative trigger such as (23). Then a declarative sentence in the same language, such as (21), repeated here as (24), would establish –ItoC.

(24) Aux Verb O1 S [ILLOC DEC]

In other words, (24) is a conditioned trigger for the negative value of the ItoC parameter, conditioned by the Complementizer-final value of the Headedness in CP parameter.

Because the Headedness in CP parameter is well-supported in all CoLAG languages, as we have just observed, it is a legitimate conditioner for –ItoC. Finding 19 showed that it is a sufficient conditioner for the classes of languages that lack globally valid triggers for –ItoC. Between them, the two values of Headedness in CP provide conditioners for all of the –ItoC languages in the IVOSC and CSOVI groups.

Putting this outcome together with results above for the CIVOS/SOVIC languages, it can be concluded there is a sufficiency of triggers (including ka and VP-edge triggers, and triggers conditioned by complementizer position) for setting –ItoC for all the languages for which it is relevant in the CoLAG domain. Therefore, with +ItoC as the default, the learnability problem for the ItoC parameter disappears. The QInv parameter is also secure, on the further assumption of a +QInv default. QInv is irrelevant in a language with +ItoC. If the default +ItoC has been reset to –ItoC on the basis of evidence from declaratives, then –QInv can be set by ka triggers. But though the missing triggers problem is thus solved, we are left with concerns about whether the movement value of a parameter could or should be its default setting.

2.3.3. Movement as the Default?

2.3.3.1. Learners' access to landmarks for detecting movement. We are on the brink of drawing the conclusion, based on the ItoC and QInv parameters, that the movement value of a parameter may be its default, taking priority for learners over the nonmovement value. Therefore, it is well to consider whether the data forcing this conclusion are representative of the natural language domain or are only due to some quirk of the CoLAG domain. A quite different take on the situation as presented so far would be that sufficient triggers for the movement value of ItoC do exist in a richer domain than CoLAG. The fact that there is no landmark for movement from I to C in CoLAG's CIVOS/SOVIC languages may merely reflect the poverty of higher projections in CoLAG tree structures, compared with natural languages where a rich variety of topic/focus-related, illocutionary, aspectual, and other projections can provide informative landmarks at all levels of the structure, as proposed by Rizzi (1997), Cinque (1999), and others.

We could have included these additional landmarks in the CoLAG domain but we did not, both for tractability reasons and also on the assumption that many of the relevant landmark items (adverbs such as *apparently*, *allegedly*, *intentionally*) are too sophisticated conceptually for young children to recognize and make use of (see also Rizzi 2005, and elsewhere, on children's truncation of higher functional structure). Perhaps this was an overcautious decision

on our part, since a common or garden temporal or locative adverb, such as *today*, might serve the purpose well enough to reveal +/–ItoC; however, that class of adverbs is not well-behaved with respect to Cinque's analysis (see Cinque 1999:28–29, 87).

This is a topic that deserves to be considered more deeply in the future. If indeed there are *child-accessible* adverbial or other landmarks at all relevant tree levels in all languages, the triggering problem for head movement would be solved, with no vacuous-movement ambiguities after all. On the other hand, if these landmarks can be recognized by learners only later in development, then deterministic learning models would need to make some allowance for that. For example, parametric underspecification and/or a hierarchy of parameter values could allow learners to lock in facts that their input reliably confirms, without having to commit to details on which their input has as yet provided them no decisive information. It remains to be seen whether greater access to landmarks could fend off the conclusion that in some cases the movement value of a parameter must be its default. However, in case that conclusion cannot be averted, it is essential to consider its consequences for a psycholinguistically plausible theory of triggers.

2.3.3.2. Linguistic defaults and learning defaults. In the case of ItoC and Qinv, movement as the default receives intriguing support from the consistently greater range of triggering properties observed in CoLAG for nonmovement: there are more triggers, and more *kinds* of triggers, for nonmovement to C than for movement to C. For example, we have observed that ItoC movement must be fed by VtoI movement in a sentence that has no auxiliary; when it is clear that VtoI movement has not occurred, ItoC movement is ruled out also. The situation is strongly asymmetric. If the *Verb* has not moved to I, it ipso facto cannot have moved to C; but if the *Verb* has moved to I, that does not (by itself) reveal whether or not the *Verb* moved on from I to C. Thus, the VtoI parameter collaborates with the ItoC parameter by creating triggers for setting –ItoC, but not for setting +ItoC.

Nevertheless, the indications that +ItoC is the default value are unwelcome as well as unexpected. We began the project with the presupposition that since movement is subject to heavier licensing conditions than nonmovement, movement should always be the marked value of a parameter, posited by learners only if specifically demanded by the input. Also, the learning models we have developed (e.g., Fodor & Sakas 2004) make much use of the parsing mechanism, and in parsing research there has been an accumulation of evidence that nonmovement analyses are preferred (cf. The Minimal Chain Principle of De Vincenzi 1991). In all of our work on learnability we have hoped to be able to maintain the principle that linguistic and psycholinguistic defaults translate into defaults for learners. However, we must report that this precept has been very difficult to maintain in the CoLAG domain. The pressure in favor of movement as the default for the ItoC parameter is not the only case.⁴⁵ Our data also require a movement default for Wh-Movement, for reasons of SP. Domain search shows

⁴⁵A mismatch between linguistic and psycholinguistic defaults can arise also for parameters that control empty categories. For the Null Subject parameter, Hyams (1986) argued that children first assume +NullSubj, although she endorsed a linguistic analysis in which a null subject had to satisfy more licensing conditions than a non-null subject. Hyams wrestled with this question of default values versus initial values in her Chapter 6 and proposed a possible solution: the Isomorphism Principle. (See also discussion in Nyberg 1992:Chapter 3.) Note, incidentally, that if movement is assumed to create an empty category (trace), then positing movement defaults undermines also the general principle that whichever value of a parameter minimizes null elements should be its default.

that *wh*-in-situ has more unambiguous triggers than *wh*-movement does. A *wh*-phrase in situ is recognizable as such as long as it is not sentence-initial, while a *wh*-phrase in sentence-initial position is ambiguous between *wh*-movement, base generation, and topicalization; see footnote 21.

On the other hand, not every movement parameter demands a reversal of the theoretically expected default. For example, the movement value +VtoI has sufficient triggers, with *not* and *never* as landmarks, thus allowing the default to be -VtoI. Which cases will fall which way is not apparent at present, but some theoretical elucidation is clearly needed. There is no simple generalization along the lines of: "Where there is a landmark between X and Y there is evidence for or against movement from X to Y, and where there is no such landmark there are no surface consequences of movement." We saw a more complex situation above for ItoC, where lack of movement from I to C can be evidenced by a landmark between V and I. Recall also that the *ka*-marking facts, not involving landmarks at all, were more sharply drawn as evidence for -ItoC than for +ItoC, again favoring a movement default. It may be true more generally that morphological markers are more reliable cues than word order, since a word order cue may be obscured by other derivational processes.

Perhaps we must simply accept that it is *detectability* that dominates the learning process. This makes practical sense: however much innate knowledge a learner may have, parameter setting requires input evidence. If or when that need for evidence clashes with linguistic markedness, the former may have to be heeded in order for language acquisition to succeed. In this way the epistemological priorities of learners may not always align with the (presumed) linguistic markedness values.

While it has been instructive to see how such clashes can arise, they raise a serious question concerning the innate predispositions that learners can be assumed to bring to the learning task. It could be expected that biological tendencies would shape both the trends in human language structure and learners' hypotheses in the absence of definitive evidence. That picture would be more intelligible if the two kinds of defaults always coincide.

3. IMPLICATIONS FOR A THEORY OF TRIGGERS

3.1. Strategies for Disambiguating Triggers

The origin of this investigation was a simulation study that showed that a learning model seeking unambiguous triggers was unable to find enough of them to set all the parameters in the CoLAG domain (section 2.1.2). This sparked the current project, designed to shed light on whether it was the domain or the learning model that was at fault. Were there indeed insufficient unambiguous triggers in the domain, or was the learning model too weak to be able to identify them? The answer we have arrived at is less cut and dried than either of these alternatives. The domain is at fault if a learner needs parameter values to be mutually independent, but the current findings show that the domain is rich enough if a learner is capable of operating with parameter values that are innately interrelated in ways that can disambiguate triggers that would otherwise be parametrically ambiguous.

This has bearing on whether deterministic parameter setting is or is not deemed to be practicable. In section 1.1 we reviewed the advantages of deterministic learning: it makes

defaults usable, halving the extent of the task; it reduces the number of grammar changes the learning system makes en route to the target; it could explain why child learners make few syntactic errors of 'comission' (Snyder 2007); it solves the acute conflict between incremental learning and the Subset Principle that pits possibly incurable undergeneration against incurable overgeneration (Fodor & Sakas 2005). Our domain search results have led to the conclusion that these benefits are attainable only through the agency of certain dependencies between parameter values. For lack of any more practical research alternative, our investigation treats the CoLAG domain as a microcosm of the domain of natural languages (while we fully acknowledge that unexpected issues can arise as a domain is scaled up). To the extent that it is representative, the empirical findings for the CoLAG domain have revealed two distinct types of interrelation between parameters that any deterministic learner must rely on to resolve ambiguities. Both do their work by disambiguating triggers that would otherwise be parametrically ambiguous and hence would be unusable by a deterministic learning mechanism, but they do so in different ways.⁴⁶

(i) In the case of a *between-parameter default*: A sentence property, π , (i.e., a pattern in the sense of section 1.2.1) that is licensed exclusively by the marked value of some parameter, $+P_i$, and the marked value of some other parameter, $+P_j$, is *stipulated* as triggering $+P_i$, leaving $+P_j$, if needed for the target language, to be set by other triggers. This is helpful in cases where $+P_i$ suffers from a shortage of triggers but $+P_j$ does not; in effect, P_i makes a *gift* of trigger π to P_i .

Example: Absence of an overt topic, which is compatible with both +OptTop and +NullTop, is stipulated as triggering +OptTop only; where needed, +NullTop is set by a different trigger, the absence of an obligatory sentence constituent.⁴⁷

Safety requirements for between-parameter defaults: A between-parameter default, although it is only a default and not guaranteed to be correct, will not lead to errors as long as two conditions are met. A correctability condition requires that $+P_j$ has sufficient triggers of other kinds, so that the learner will subsequently encounter unambiguous evidence of $+P_j$ if the target is $-P_i, +P_j$. A no-aftereffects condition requires that if $+P_j$ is subsequently established, there will be no lingering consequences of the incorrect $+P_i$ that was adopted previously; if there were, the result would be a grammar with $+P_i, +P_j$, which could overgenerate if the target grammar were $-P_i, +P_j$. This condition can be satisfied in two ways: if $+P_i, +P_j$ is disallowed by UG so that P_i must revert to its default value $-P_i$ when $+P_j$ is set; or if $+P_i$ is irrelevant when $+P_j$ is

⁴⁶What we are calling between-parameter *constraints*, such as in Table 2, also facilitate parameter setting but not by disambiguation. For the learner, their effect is that two or more parameters can be simultaneously set by one trigger. For linguistic theory, their effect is to reduce the exponential explosion of possible languages as more parameters are identified. On the significance of the latter point see discussion by Gianollo, Guardiano & Longobardi (2008).

⁴⁷Whereas a simple (within-parameter) default can be conceptualized as an aspect of the starting state of the learning mechanism, a between-parameter default cannot be. In the starting state each parameter has its own individual default value, and the between-parameter default becomes relevant only when input requires that one of the two parameters be set forward to its nondefault value. So, as another modest contribution to the theory of triggers, it seems that default parameter values must be characterized not as initial-state settings prior to any experience, but as what learners are entitled to assume in the absence of sufficient input evidence.

set, so that $+P_i$, $+P_j$ licenses no more sentences than $-P_i$, $+P_j$ does. The first case is illustrated by +OptTop in relation to +NullTop in section 2.2.5, and the second case by +QInv in relation to +ItoC in section 2.3.1.1 (prior to changing the default to +ItoC).

(ii) In the case of a *conditioned trigger*: A sentence property, π , compatible with two (or more) parameter values v_i and v_j (values of the same parameter or different ones) becomes unambiguous as a trigger for v_i once the value(s) of some other parameter(s) have been established.

Example: The surface word order pattern *AuxVOS* is compatible with either –ItoC or +ItoC but becomes an unambiguous trigger for –ItoC once the Headedness in CP parameter has been set to Complementizer-final.

Safety requirement for conditioned triggers: A trigger for value v_i conditioned by the value(s) of other parameter(s) must meet a *confidence requirement*: the learner must be certain that those other parameters are correct; otherwise v_i might be permanently mis-set.⁴⁸ This is an important difference between a conditioned trigger and a between-parameter default. The latter, like any other default (e.g., between the two values of a single parameter), amounts to a temporary guess which may have to be corrected on the basis of subsequent information. By contrast, a conditioned trigger meeting the confidence requirement is a fully valid trigger on a par with "normal" (unconditioned) triggers; it is completely and permanently reliable, given that the conditioning parameter value(s) were correct. As noted above, this confidence requirement rules out using a default value as conditioner except in languages in which the default has unambiguous triggers.

One more aspect of the theory of triggers has made itself felt in this investigation and is worth highlighting here. Not every observable effect of a parameter value qualifies as a trigger for that value: languages sometimes differ in ways that do not constitute triggers in the classical sense. Such differences might be used informatively by a more reflective type of learner (a hypothesis-formation-and-testing model or a linguist), but not by a triggering system. This can create a precarious state of affairs in which a parameter value needs to be set because it makes a difference to the language generated (it is not irrelevant), yet that difference in the language cannot itself be the trigger. On standard assumptions for an incremental triggering learning device such as the original switch-box model, a trigger must be a property exemplified in a single sentence encountered by the learner, which uniquely establishes the value of one (or more) parameters. The coexistence of two (or more) different sentence types in a language does not qualify as a trigger for an incremental learner. For example, -ItoC is evidenced in CoLAG by a difference in the finite verb location in questions versus declaratives (e.g., sentence-final Aux in questions but initial Aux in declaratives). But in a strict (incremental) triggering model, this between-sentence comparison would not be usable. Also, a sentence with disjunctive implications for parameter values (e.g., a sentence with no overt topic entails

 $^{^{48}}$ In describing the relation between OptTop and NullTop in section 2.2.5, we opted for a between-parameter default but observed that conditioning might achieve the same result. That is, +OptTop might be regarded as conditioned by -NullTop, though only if the confidence requirement on conditioning were waived in cases where retrenchment of the conditioned value occurs automatically when the conditioning value is reset. Our definition in (ii) does not include this provision. However, the possibility remains open; CoLAG provides no decisive examples.

either +OptTop or +NullTop; a question without ka entails either +QInv or +ItoC) is not a trigger in this sense. A universally quantified fact about the sentences of a language (e.g., every declarative sentence contains an auxiliary) is also not a bona fide trigger, even though it can contain valuable information (that the language is either -AfHop,-VtoI or -VtoI,+ItoC). Likewise, negative facts (e.g., no sentence of the language has a finite main verb) are not usable by a triggering learner that relies on positive input only.

However, there are some familiar strategies for recasting such nontriggers into the format of legitimate triggers. For example, where universal or negative observations appear to be necessary, they can be assigned as the default (if not incompatible with other facts), with the trigger for the opposite (specific, positive) value being what the learner must observe in the input. For instance, the learner's default might be that all nonimperative sentences have an auxiliary verb, the trigger to counteract this being a single input sentence with a finite main verb. Also, where a language fact would lead to a disjunctive parametric conclusion, equally compatible with two marked parameter values, a between-parameter default can sometimes allow it to be recast as an unambiguous trigger for one of them, to be overridden later by input evidence for the other one, as in the case of +OptTop versus +NullTop. Where a parameter value is signaled by a combination of language facts exemplified in different sentences, conditioning may allow triggering to proceed as a two-step process: one fact sets a parameter value, which then legitimizes the setting of another one. For example, an Aux-final question without ka can be used to set Complementizer-final; then Complementizer-final can condition an Aux-initial declarative as a trigger for –ItoC.

Such strategies for converting illegitimate triggers into usable ones are what made the difference between finding too few E-triggers in our initial search of the domain, and finding sufficient E-triggers in our final analysis. It is noteworthy that these strategies are close relatives of proposals made by DK/D in their search for unambiguous triggers ("cues") for metrical structure in phonology. Our project was independent of Dresher & Kaye's: it addressed a different level of linguistic structure; the specific puzzles it confronted were not similar; our methodology was unrelated to theirs. Nevertheless, both investigations arrived at the conclusion that error-free (deterministic) parameter setting cannot be achieved by adherence to a maximally simple theory of triggers that makes no allowance for certain kinds of interrelationships between parameter values. From the broad perspective of learnability theory, this convergence may be a hint that the general approach is on the right track. The key concept, as became evident in the course of our own empirical study, is the innately imposed disambiguation of intrinsically ambiguous triggers. This succeeds because it creates unambiguous triggers, rather than merely searching for them. It is encouraging, also, that for both phonology and syntax, despite their divergent formal profiles, relatively little needs to be added to the classical notion of triggers in order to achieve this. On current showing it might even be speculated that all apparent weaknesses of the input can be weathered by a deterministic learning mechanism with just this modest toolkit of disambiguating devices: assignment of default values to avoid quantified, disjunctive, and negative triggers; default relations between as well as within parameters; triggers for one parameter conditioned by the value of another.

3.2. Could a Learner Know and Use These Triggers?

Findings such as ours and those of DK/D do not, and cannot in principle, prove that deterministic

learning is how children acquire language. The most these findings can do is to defuse objections that have been raised against determinism on grounds of input ambiguity, and encourage further study of the circumstances in which a deterministic approach is feasible. An additional necessary requirement for the viability of this approach is that it could be incorporated into a learning model that functions well in other respects. As a foundation for deterministic learning, the psycholinguistic naturalness of triggers is as important as their raw existence. Thus, what remains to be established is whether these new twists on the nature of triggers could be integrated comfortably into existing acquisition theories. For instance, how could conditioned parameter settings be implemented within a psychologically realistic learning algorithm? If they can be incorporated easily, and especially if they can be incorporated in an explanatory manner, then determinism will be paying its way; if not, their relevance for theories of human language acquisition is marginal. We cannot do justice to this topic here. But we can set up the issue for future debate by outlining some alternatives. One alternative is an implementation that incorporates the new types of triggers into a classic 1980s model. Another embeds them within the learning-by-parsing approach of our own previous research.

Interrelations between the triggers for different parameters can be incorporated easily enough into a classical deterministic switch-flipping model of parameter setting. The E-trigger schemas that we have identified in the body of this article, and in Appendix A, would do duty as the pattern detectors (templates) associated with the parametric switches, their role being to recognize the input sentences that are qualified to flip each of the switches. An input sentence that matches the template for the marked value of parameter P_i would cause the P_i switch to be set to that marked value. As we have observed, a crucial issue is how input ambiguity is handled. In the simplest incarnation of this model, an input sentence that matches the templates for more than one parameter value would be prevented from triggering either one. But the consequence, as our investigation has shown, is that some parameters would then lack sufficient triggers. In an updated model that incorporates between-parameter defaults and conditioning: when an input sentence matches the templates for more than one parameter value, the defaults and/or conditioning relations would regulate the competition between them. In the case of a between-parameter default, the template for one parameter would take priority over the other, and would grab the incoming trigger for itself. In the case of a conditioned trigger, the template for the conditioned parameter value would be inaccessible to input until the template for the conditioning parameter value had been satisfied. So conceived, the interrelations between parameters constitute straightforwardly implementable, minimal modifications of the old switch-setting model; they amount to no more than traffic rules, to guide which switch, if any, an input sentence will gravitate to in case of potential conflicts.

However, the classic model has been criticized in the past for lack of explanatoriness in the relation between its trigger templates (E-level) and the parametric switches (at the I level) that they attach to. Gibson & Wexler (1994) raised important questions along these lines. They considered two starkly contrasting alternatives. If the learning mechanism had to deduce which E-triggers are the surface realizations of which parameter values, it would need far richer computational powers than a mere switch-setting mechanism; the concept of triggering would not, after all, render language acquisition any more simple or transparent than earlier models. On the other hand, if the parametric templates (E-trigger schemas) and their deeper derivational consequences (the I-triggers or parameter values) were independently specified in the biological preprogramming of an infant's language faculty, there would be massive

redundancy, since the relation between the two is not at all arbitrary or accidental. In other words, it is unclear how the gap between I and E levels can be bridged satisfactorily in a traditional switch-setting model, even if a practical implementation can be created.⁴⁹ Is it possible that there could be a meta-component of the learning mechanism which, given innate I-trigger specifications, deduces how they would translate into E-level templates? Gibson & Wexler were clearly skeptical. But even if there were, it is difficult to imagine that it could also anticipate all the surface overlaps between potential triggers at the E-level and devise just the right strategies to disambiguate them, as we have seen is essential for deterministic learning in a domain fraught with parametric ambiguity.

Explanatory questions may receive stronger answers if the new trigger types are incorporated into the learning-by-parsing approach to parameter setting that we have advocated on independent grounds in previous work (Fodor 1998b; Sakas & Fodor 2001). We have developed a family of learning algorithms, known as Structural Triggers Learners (STLs), all of which employ treelets as I-level triggers (section 1.2.2 above). The variants of the model differ from each other in precisely how the treelets are employed by the learning algorithm, but those differences can be overlooked for present purposes. We will first sketch out how E-triggers could be grafted onto the basic STL mechanism. In a minimal adaptation they would function much as described above for a switch-setting model. Then we will take a more speculative tack with the promise of gaining some deeper explanations. Specifically, in the STL framework, the existence of conditioned triggers, far from being an added complication of the learning process as might reasonably be anticipated, can be seen as a simple consequence of the learner's commitment to determinism.

The STL models use the parametric treelets (equivalently: I-triggers) in parsing input word strings. When a treelet is found to be necessary in order to build a parse-tree for an input string, it is adopted into the learner's grammar.⁵⁰ In this way, the work of correlating E-level facts with the I-level parametric treelets is effected by the parsing routines. A parsing device is independently needed for language comprehension by both adults and children, and in that role its normal job is to take in a word string and compute a syntactic structure for it, i.e., to translate from E-level facts to I-level facts. (Note that here the direction is from E to I, unlike deduction from I to E as Gibson & Wexler contemplated.) At each point in a word string encountered during normal sentence comprehension, the parsing mechanism has to find a legitimate set of syntactic nodes (a treelet) by which to connect the incoming word into the tree structure it has built for the sentence so far. The only difference between adult parsing and child grammar acquisition is that the adult has a stable grammar, so the parser is working with a fixed set of such treelets, while the learner is not yet sure of the target grammar and hence has a larger pool of potential treelets to draw on, which will be narrowed down over time as some of them prove useful in parsing target sentences and some do not. How could innate

⁴⁹How sizeable that gap is depends on what a particular linguistic theory takes the I-triggers to be. The degree of abstractness of the E-I association may not be the same, for example, in Construction Grammar and the Minimalist Program. Each theory adheres to the conclusions of its own argumentation in this regard. However, it is inescapable that there must be *some* distinction between E and I on *any* theory of syntax, on our assumption that I-triggers necessarily encode structure while E-triggers are surface-recognizable in word strings.

⁵⁰See Pearl (2007:Chapter 5) for comparison of the relative merits of using the parser to identify unambiguous triggers for metrical phonology versus assuming that they are innately listed as DK/D-type cues. Her conclusion is that a compromise between the two approaches offers the greatest chance of success.

knowledge of unambiguous E-triggers contribute to the parser's task? The straightforward answer is as follows. When the parser encounters a word string that falls beyond the scope of the learner's current grammar, the string could be checked against the innate collection of E-triggers schemas, functioning in much the same fashion as templates did above: if the string was found to fit more than one of the schemas, the various E-trigger prioritizations (defaults, conditioning relations) would indicate to the parser which I-level treelet, if any, would be safe to use in parsing the string.

The merit of this approach relative to classic switch-setting does not lie in its treatment of E-triggers since they are similar in that respect, but in its implementation of trigger recognition, which it integrates more closely with the normal psychological mechanisms for language comprehension, postulating less mental apparatus whose *only* role is to achieve acquisition.

Now consider a more radical variant of the STL approach in which there is no need for the E-triggers to be specified in the learning mechanism at all. In this scenario, the E-triggers are simply the parsing problems that the learner will run into, which need to be solved by calling on a suitable I-trigger in the innately provided pool of treelets. For example, a child exposed to English will at some point encounter a P separated from its O3 in a word string (the E-level), which its current grammar does not license, and the parsing routines will discover that only the treelet consisting of a PP node with a 'SLASH O3' feature provides a means of parsing it (section 1.2.2). Note that E-triggers still play an essential role, since the parsing problems they pose to the learner are what drive adoption of new parameter values into the grammar; so it is just as important for this model as for any other that the input should provide enough E-triggers to get the learning done. But this learner does not need to be equipped with an innate list of E-triggers in addition to an innate list of the I-triggers.⁵¹ The innate stock of I-trigger specifications, together with the usual expertise of the innate parsing mechanism, is sufficient.

This dissolves the explanatory puzzle raised above concerning how the right E-triggers get to be associated with the right parameter values. In the learning-by-parsing framework, it would not be possible for a parametric treelet to be used to patch an incomplete parse-tree if it were not appropriately related in content to the word string being parsed; hence, E-level facts and I-level parameter values are *inherently* associated. This more radical version of the STL parsing-based model has another explanatory advantage: it does not require conditioning relations between parameters to be innately specified. Instead, as we now explain, they fall out automatically from the deterministic learner's commitment to avoiding ambiguity.

In the case of a conditioned trigger, the conditioned parameter value must be inaccessible for use by the parser until the conditioning parameter value has been established. In the kind of model under consideration, this temporary inaccessibility does not need to be stipulated case by case (i.e., for each pair of conditioner and conditioned parameter). This is because a committed deterministic learner adopts new parameter values only on the basis of input that is parametrically unambiguous (inherently so, or in consequence of a disambiguation strategy). A conditioned trigger is by definition parametrically ambiguous *before* its conditioning

⁵¹As noted in section 1.2.2 above, Lightfoot's proposals have a similar character in that the learner employs I-level treelets ("cues"), and the learning mechanism is not explicitly provided with associated E-level facts. A difference from the current proposal, as noted, is that the parsing process by which Lightfoot's E-level cues are recognized by the learner in surface sentences is approximate, based to some extent on guesswork since E/I relations are not articulated. For Lightfoot, this has the merit of allowing for imperfect transmission of the grammar from one generation to the next.

parameter value is set, so it will be shunned by the learner. If and when the conditioning parameter value is established, that eliminates the ambiguity, so there is now no bar to using the conditioned trigger.⁵²

For instance, a string such as (25) is ambiguous. It can be parsed using either the –ItoC treelet, or the +ItoC treelet in conjunction with Complementizer-initial. (See discussion of IVOSC languages in section 2.3.2.2.)

(25) Aux Verb O1 S [ILLOC DEC]

Thus for a learner that has not yet set the Complementizer Position parameter, the competition between these two analyses of (25) would prevent a deterministic learning mechanism from adopting either of them. But notice that (25) is also an unambiguous conditioned trigger for –ItoC, conditioned by Complementizer-final. If the learner subsequently encountered evidence to establish the conditioning value Complementizer-final, the competition from +ItoC would disappear; only –ItoC would afford a parse for the sentence. Once –ItoC has no competition, it can and would be adopted by a deterministic learner. Note that this would play out without any need for the learning mechanism to be innately informed about the connection between the target value –ItoC, the conditioning value Complementizer-final, and the *Aux*-initial trigger. The parsing mechanism is doing the crucial "reasoning" that establishes which parameter values must be conditioned by which others, and it does so not via some special module for deducing E/I relations, but entirely on its own terms: once Complementizer-final is set, +ItoC just fails to afford a parse for the input sentence. In this way, ambiguity detection by the parser, and commitment to determinism by the learner, can release the triggering power of input just when it is safe to do so.

We see this as an interesting prospect for future research, but this idea has the status of a speculation only, and is perhaps a long shot. This is because our previous learning simulation studies have pointed up practical problems in the parser's on-line recognition of unambiguous triggers (when E-triggers are not supplied in a list). Specifically, in the absence of full parallel parsing capacity (which is surely not realistic), the presence of other (irrelevant) ambiguities in a sentence makes it difficult to establish that a certain treelet is truly necessary for the parsing of that sentence. This forces a deterministic learner into a policy of extreme caution (see section 1, Theoretical Framework). There are several ways of tackling this situation, such as learning from sentence fragments (e.g., an isolated PP in answer to a question), which present fewer parametric alternatives. But a more general solution to this problem may perhaps be forged now that we have seen that the key to deterministic learning is the disambiguation of intrinsically ambiguous triggers. If this approach stands up to further scrutiny, then simulation tests can be conducted to determine how its efficiency compares with that of previously proposed models nondeterministic models and earlier deterministic models tested in our prior experiments. If it not only survives that test but even performs well, another important exercise would be to expand the language domain to see whether adding more parameters introduces new E-level

⁵²Unlike conditioning relations, default relations (within or between parameters) do not follow solely from ambiguity avoidance. It appears that the priority ordering in the case of a default relation must still be stipulated, unless perhaps the parsing/learning mechanism could derive it from the Subset Principle or perhaps from a general parsing principle, or general markedness criteria.

overlaps that undermine the unambiguous triggers established here. We welcome suggestions as to which parameters might have the potential to wreak some significant damage.

To conclude, in this final subsection we have taken a very preliminary look at how Etrigger disambiguation might fit into a psychologically realistic learning mechanism. A plausible conjecture was that disambiguation strategies would be bound to add to the complexity of the learning process and its innate underpinnings. However, if the ideas just floated can eventually be substantiated, then it seems that trigger disambiguation may not after all be psychologically costly and might even be a natural consequence of the deterministic learning that it serves.

ACKNOWLEDGEMENTS

We are grateful for feedback and suggestions we received at presentations of this work in earlier stages of development (the Workshop on Formal Models of Linguistic Diversity at the University of Trieste; the 2008 Joshua and Verona Whatmough Lecture at Harvard University; ANPOLL's First International Psycholinguistics Congress, Rio de Janeiro; the Input and Syntactic Acquisition Workshop at the University of California, Irvine; Mayfest 2010 at the University of Maryland; and colloquia at McGill University, the University of Connecticut, and the University of Pennsylvania). Marcel den Dikken deserves special gratitude as always for keeping us not too far from the bounds of accuracy with regard to our syntactic assumptions. Three anonymous reviewers and the action editor, Colin Phillips, provided wise advice on how to revise an unwieldy first draft.

REFERENCES

- Bertolo, Stefano. 1995. Learnability properties of parametric models for natural language acquisition. New Brunswick, NJ: Rutgers University dissertation.
- Bertolo, Stefano, Kevin Broihier, Edward Gibson & Kenneth Wexler. 1997a. Characterizing learnability conditions for cue-based learners in parametric language systems. *Proceedings of the Fifth Meeting on Mathematics of Language* [MOL5]. http://www.dfki.uni-kl.de/dfkidok/publications/D/97/02/abstract.html (30 November, 2011.)
- Bertolo, Stefano, Kevin Broihier, Edward Gibson & Kenneth Wexler. 1997b. Cue-based learners in parametric language systems: Application of general results to a recently proposed learning algorithm based on unambiguous 'superparsing.' In Michael G. Shafto & Pat Langley (eds.), Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society [CogSci-1997], 49–52. Mahwah, NJ: Lawrence Erlbaum.
- Biberauer, Theresa. 2008. Introduction. In Theresa Biberauer (ed.), *The limits of syntactic variation*, 1–72. Amsterdam: John Benjamins.
- Briscoe, Edward John. 2000. Grammatical acquisition: Inductive bias and coevolution of language and the language acquisition device. *Language* 76(2). 245–296.
- Cheng, Lisa & Rint Sybesma. 2003. *The second Glot International state-of-the-article book: The latest in linguistics*. Berlin: Mouton de Gruyter.
- Chomsky, Noam. 1981. Lectures on government and binding. Dordrecht: Foris.
- Chomsky, Noam. 1986. Knowledge of language: Its nature, origin, and use. New York: Praeger.
- Cinque, Guglielmo. 1999. Adverbs and functional heads: A cross-linguistic perspective. Oxford: Oxford University Press.
- Clark, Robin. 1989. On the relationship between the input data and parameter setting. In Julie Carter & Rose-Mar Déchaine (eds.), Proceedings of the 19th Annual Meeting of the North East Linguistic Society [NELS 19], 48–62. Amherst, MA: University of Massachusetts, GSLA.

- Clark, Robin. 1992. The selection of syntactic knowledge. Language Acquisition 2(2). 83-149.
- Clark, Robin & Ian Roberts. 1993. A computational model of language learnability and language change. *Linguistic Inquiry* 24(2). 299–345.
- De Vincenzi, Marica. 1991. Syntactic parsing strategies in Italian. Dordrecht: Kluwer.
- Dresher, B. Elan. 1999. Charting the learning path: Cues to parameter setting. Linguistic Inquiry 30(1). 27-67.
- Dresher, B. Elan & Jonathan D. Kaye. 1990. A computational learning model for metrical phonology. *Cognition* 34(2). 137–195.
- Elman, Jeffrey L. 1993. Learning and development in neural networks: The importance of starting small. *Cognition* 48(1). 71–99.

Fodor, Janet Dean. 1989. Learning the periphery. In Robert J. Matthews & William Demopoulos (eds.), *Learnability* and linguistic theory, 129–154. Dordrecht: Kluwer.

- Fodor, Janet Dean. 1994. How to obey the Subset Principle: Binding and locality. In Barbara Lust, Gabriella Hermon & Jaklin Kornfilt (eds.), *Syntactic theory and first language acquisition: Cross-linguistic perspectives: Volume 2, binding, dependencies and learnability*, 429–451. Hillsdale, NJ: Lawrence Erlbaum.
- Fodor, Janet Dean. 1998a. Unambiguous triggers. Linguistic Inquiry 29(1). 1-36.
- Fodor, Janet Dean. 1998b. Parsing to learn. Journal of Psycholinguistic Research 27(3). 339-374.
- Fodor, Janet Dean. 2009. Syntax acquisition: An evaluation measure after all? In Massimo Piattelli-Palmarini, Juan Uriagereka & Pello Salaburu (eds.), *Of minds and language: A dialogue with Noam Chomsky in the Basque Country*, 256–277. Oxford: Oxford University Press.
- Fodor, Janet Dean & William Gregory Sakas. 2004. Evaluating models of parameter setting. In Alejna Brugos, Linnea Micciulla & Christine E Smith (eds.), Proceedings of the 28th annual Boston University Conference on Language Development [BUCLD 28], 1–27, Somerville, MA: Cascadilla Press.
- Fodor, Janet Dean & William Gregory Sakas. 2005. The Subset Principle in syntax: Costs of compliance. *Journal of Linguistics* 41(3). 513–569.
- Fodor, Janet Dean, William Gregory Sakas & Arthur Hoskey. 2007. Implementing the Subset Principle in syntax acquisition: Lattice-based models. In Stella Vosniadou, Daniel Kayser & Athanassios Protopapas (eds.), *Proceedings of the Second European Cognitive Science Conference [EuroCogSci07]*, 161–166. Hove, UK: Lawrence Erlbaum.
- Fodor, Janet Dean & Virginia Teller. 2000. Decoding syntactic parameters: The superparser as oracle. In Lila R. Gleitman & Aravind K. Joshi (eds.), Proceedings of the Twenty-second annual Conference of the Cognitive Science Society [CogSci-2000], 136–141. Hillsdale, NJ: Lawrence Erlbaum.
- Frank, Robert & Shyam Kapur. 1996. On the use of triggers in parameter setting. Linguistic Inquiry 27(4). 623-660.
- Fujimoto, Mari. 2008. *L1 acquisition of Japanese particles: A corpus-based study*. New York: City University of New York dissertation.
- Gianollo, Chiara, Cristina Guardiano & Giuseppe Longobardi. 2008. Three fundamental issues in parametric linguistics. In Theresa Biberauer (ed.), *The limits of syntactic variation*, 109–142. Amsterdam: John Benjamins.

Gibson, Edward & Kenneth Wexler. 1994. Triggers. Linguistic Inquiry 25(3). 407-454.

- Hauser, Marc D., Noam Chomsky & W. Tecumseh Fitch. 2002. The faculty of language: What is it, who has it, and how did it evolve? *Science* 298(5598). 1569–1579.
- Huang, C.-T. James. 1984. On the distribution and reference of empty pronouns. Linguistic Inquiry 15(4). 531-574.

Hyams, Nina M. 1986. Language acquisition and the theory of parameters. Dordrecht: Reidel.

- Kapur, Shyam. 1994. Some applications of formal learning theory results to natural language acquisition. In Barbara Lust & Gabriella Hermon (eds.), Syntactic theory and first language acquisition: Cross-linguistic perspectives. Volume 2: Binding, dependencies, and learnability, 491–508. Hillsdale, NJ: Lawrence Erlbaum.
- Kohl, Karen T. 1999. An analysis of finite parameter learning in linguistic spaces. Cambridge, MA: MIT master's thesis.
- Li, Charles N. & Sandra A. Thompson. 1976. Subject and topic: A new typology of language. In Charles N. Li (ed.), *Subject and topic*, 457–489. New York: Academic Press.
- Lightfoot, David. 1991. How to set parameters: Arguments from language change. Cambridge, MA: MIT Press.

Lightfoot, David. 1998. The development of grammars. Glot International 3(1). 3-8.

- Lightfoot, David. 1999. The development of language: Acquisition, change, and evolution. Malden, MA: Blackwell.
- Lightfoot, David. 2003. The development of grammars. In Lisa Cheng & Rint Sybesma (eds.), *The second Glot International state-of-the-article book: The latest in linguistics (Studies in generative grammar 61)*, 1–24. Berlin: Mouton de Gruyter.

Lightfoot, David. 2006. How new languages emerge. Cambridge: Cambridge University Press.

- Manzini, M. Rita & Kenneth Wexler. 1987. Parameters, binding theory, and learnability. *Linguistic Inquiry* 18(3). 413–444.
- Modesto, Marcello. 2008. Topic prominence and null subjects. In Theresa Biberauer (ed.), *The limits of syntactic variation*, 375–409. Amsterdam: John Benjamins.
- Morgan, James L. 1986. From simple input to complex grammar. Cambridge, MA: MIT Press.
- Morgan, James L., Richard P. Meier & Elissa L. Newport. 1987. Structural packaging in the input to language learning: Contributions of prosodic and morphological marking of phrases to the acquisition of language. *Cognitive Psychology* 19(4). 498–550.
- Nyberg, Eric Harold, III. 1992. A non-deterministic, success-driven model of parameter setting in language acquisition. Pittsburgh, PA: Carnegie Mellon University dissertation.
- Pearl, Lisa. 2007. Necessary bias in natural language learning. College Park, MD: University of Maryland dissertation.
- Pearl, Lisa. 2009. Learning English metrical phonology: When probability distributions are not enough. In Jean Crawford, Koichi Otaki & Masahiko Takahasi (eds.), Proceedings of the 3rd Conference on Generative Approaches to Language Acquisition North America [GALANA 2008], 200–211. Somerville, MA: Cascadilla Press.
- Pearl, Lisa & Jeffrey Lidz. 2009. When domain general learning succeeds and when it fails. Language Learning and Development 5. 235–265.
- Perfors, Amy, Joshua B. Tenenbaum & Terry Regier. 2006. Poverty of the stimulus? A rational approach. In R. Sun & N. Miyake (eds.), Proceedings of the 28th annual Conference of the Cognitive Science Society [CogSci-2006], 663–668. Mahwah, NJ: Lawrence Erlbaum.
- Pollock, Jean-Yves. 1989. Verb movement, universal grammar, and the structure of IP. *Linguistic Inquiry* 20(3). 365–424.
- Pullum, Geofrey K. & Barbara C. Scholz. 2002. Empirical assessment of stimulus poverty arguments. *Linguistic Review* 19(1/2). 9–50.
- Ritter, Nancy (ed.). 2002. A review of "the poverty of stimulus argument" (special issue). The Linguistic Review 19.
- Rizzi, Luigi. 1997. The fine structure of the left periphery. In Liliane Haegeman (ed.), *Elements of grammar: Handbook of generative syntax*, 281–337. Dordrecht: Kluwer.
- Rizzi, Luigi. 2005. Grammatically-based target-inconsistencies in child language. In Kamil Ud Deen, Jun Nomura, Barbar Schulz & Bonnie D. Schwartz (eds.), Proceedings of the Inaugural Conference on Generative Approaches to Language Acquisition – North America [GALANA], 19–49. Cambridge, MA: University of Connecticut Working Papers in Linguistics (published by MITWPL).
- Roberts, Ian. 2001. Language change and learnability. In Stefano Bertolo (ed.), *Language acquisition and learnability*, 81–125. Cambridge: Cambridge University Press.
- Sag, Ivan A., Thomas Wasow & Emily M. Bender. 2003. Syntactic theory: A formal introduction, 2nd edn. Stanford, CA: CSLI Publications.
- Sakas, William Gregory. 2003. A word-order database for testing computational models of language acquisition. In Erhard Hinrichs & Dan Roth (eds.), Proceedings of the 41st annual Meeting of the Association for Computational Linguistics [ACL-2003], 415–422. Stroudsburg, PA: ACL.
- Sakas, William Gregory & Janet Dean Fodor. 2001. The Structural Triggers Learner. In Stefano Bertolo (ed.), Language acquisition and learnability, 172–233. Cambridge: Cambridge University Press.
- Sakas, William Gregory & Janet Dean Fodor. 2003. Slightly ambiguous triggers for syntactic parameter setting. Paper presented at the 9th Annual Conference on Architectures and Mechanisms for Language Processing (AMLaP-2003), August 25–27, University of Glasgow, Glasgow Scotland.
- Snyder, William. 2007. Child language: The parametric approach. Oxford: Oxford University Press.
- van Kampen, Jacqueline. 1997. First steps in wh-movement. Delft, The Netherlands: Eburon.
- Villavicencio, Aline. 2001. The acquisition of a unification-based generalised categorial grammar. Cambridge: Cambridge University dissertation.
- Yang, Charles D. 2002. Knowledge and learning in natural language. Oxford: Oxford University Press.
- Yang, Charles D. 2010. Who's afraid of George Kingsley Zipf? Ms. University of Pennsylvania, Philadelphia.

Submitted 24 August 2010 Final version accepted 20 January 2012

APPENDIX A

Below in Table 4 we present some globally valid triggers for the 10 'non-problem' parameters in the CoLAG domain, which were found to have unambiguous abstract E-triggers in all relevant languages, without need for conditioning or between-parameter defaults. To simplify the table, triggers are shown here for only the non-default value of each parameter, even if both values have sufficient triggers (in which case one value is arbitrarily designated the default for convenience in this presentation). For reasons of space, most of the I-trigger treelets are described, not displayed. Only one or two E-triggers are listed for each parameter, though often there exist more. Except where noted otherwise, all triggers shown are globally available (except in languages for which the parameter value is irrelevant). They are all in declarative sentences unless specified otherwise. The feature [+/-NULL] is omitted except where its value is relevant in an I-trigger. Also, the term "follows" in the rightmost column of the table does not mean immediately (adjacently) follows. Reference to non-sentence-initial position is to ensure that an item has not been fronted to the specifier of CP (which is the landing site for XP movement in CoLAG).

APPENDIX B

The definitions below of global and local triggers (E-triggers) are from Gibson & Wexler (1994). (Henceforth we refer to that article as GW.) These definitions have entered the learnability literature as a standard reference point, and so we have built on them, diverging only as necessary to accommodate the greater complexity of our language domain and the nature of our investigation.⁵³ Our project, unlike GW's, was to assess whether natural language parameters have a sufficiency of unambiguous triggers, a topic of general interest but especially relevant to the question of whether parameter setting can be error-free and deterministic. For this purpose it was essential to distinguish between the validity of triggers (i.e., their unambiguity) and their availability (i.e., their presence in languages needing the parameter value). This is a distinction not drawn by GW's definitions. The most straightforward case, presumably the optimal case, would be a parameter value with a single globally available globally valid trigger. But in the CoLAG domain we found several parameters that could achieve a full array of unambiguous triggers only by piecing together a combination of different triggers, each one only locally available, i.e., present in different languages. We also found parameters such that, for some languages, the only unambiguous triggers were locally but not globally valid; these were the conditioned triggers whose validity was dependent on the values of other parameters.

For these reasons, our own definitions, in section 1.2.1 above, needed to draw a three-way distinction between

- (a) triggers that are both globally valid (unambiguous) and globally available (ubiquitous);
- (b) triggers that are globally valid (unambiguous) but not globally available (present in only some of the languages with the parameter value in question); and
- (c) triggers that are only locally valid (unambiguous only in some languages) and therefore only locally available.

⁵³See also Frank & Kapur (1996) where the consequences of several alternatives to GW's definitions are explored.

Pre-publication manuscript. Please do not quote without authors' permission.

-	
4	
0	

TABLE 4 Some Globally Valid (Unambiguous) Triggers for the Nondefault Values of Ten Nonproblem' Parameters in the CUNY CoLAG Domain

Parameter	Nondefault Value	I-Trigger (Treelet) for the Nondefault Value	Some E-Triggers for the Nondefault Value
Subject Position	Subject final	S is the right sister of IBAR.	S follows Verb[-FIN]. S follows a non-sentence-initial O1.
Headedness in IP, NegP, VP, PP	Head final	For $X = I$, Neg, V or P: X is the right sister of any/all of its complements under XBAR.	Non-sentence-initial O3 P. O1 Verb [ILLOC IMP] (see footnote 55)
Headedness in CP	Complementizer final	C is the right sister of its IP complement.	 Overt complementizer ka in final position in a question. Aux in final position in a question if no ka is present. (Both triggers are locally available, but jointly sufficient.)
Preposition Stranding	Stranding	[SLASH O3]	P and O3 both present but not adjacent.
Topic Marking	Obligatory	A non-head daughter of CP carries a suffix <i>-wa</i> .	-wa is present.
Null Subject	Null subject permitted	IP dominates an S[+NULL] which does not have [SLASH S].	Subject absent from a declarative, and an overt complement out of obliqueness order (e.g., O2 not between O1 and P O3). ^a
Null Topic	Null topic permitted	The non-head daughter of CP has [+NULL].	O2 present but O1 absent.
Wh-Movement	No	Feature [+WH] within IP.	X[+WH] in non-sentence-initial position. ^b
Affix Hopping	Obligatory	VP[+FIN]	never Verb[+FIN] O1. Non-sentence-initial O1 Verb[+FIN] never. (Available only in head-initial and head-final languages respectively, but jointly sufficient.)
VtoI Movement	Obligatory	I dominates Verb.	Verb and a non-sentence-initial O1 both present but not adjacent.

^{*a*}See footnote 31 above concerning the default for Null Subject. Also, note that this specification entails that the subject is not absent due to +NullTop.

^bIn CoLAG wh-movement is obligatory when applicable. No more than one *wh*-item is permitted per sentence, so a *wh*-in-situ proves that no *wh*-item has been fronted.

Pre-publication manuscript. Please do not quote without authors' permission.

DISAMBIGUATING SYNTACTIC TRIGGERS 141

Gibson and Wexler's Table 2 (GW 1994:417)			
Specifier-Head	Complement-head	Data in defined grammar	
Spec-final	Comp-final	VS, VOS	
Spec-final	Comp-first	VS, OVS	
Spec-first	Comp-final	SV, SVO	
Spec-first	Comp-first	SV, SOV	

TABLE 5 T-1-1-0 (0)A/ 4004 447

GW's simpler two-way classification of global and local triggers is shown in (GW1) and (GW2).

- (GW1) A global trigger for value v of parameter P_i , $P_i(v)$, is a sentence S from the target grammar L such that S is grammatical if and only if the value for P_i is v, no matter what the values for parameters other than P_i are.⁵⁴
- (GW2) Given values for all parameters but one, parameter P_i , a local trigger for value v of P_i , $P_i(v)$, is a sentence S from the target grammar L such that S is grammatical if and only if the value for P_i is v.

Note that (GW1) corresponds to trigger type (a), triggers that are both globally valid and available. (GW2) characterizes what we have called conditioned triggers, type (c). Intermediate trigger type (b) is not covered. In order to refine these definitions, we distinguished between validity and availability by separating out the "if" and "only if" clauses in the originals.

GW provided illustrations of their definitions by applying them to a simple domain of artificial languages defined by two word order parameters: a specifier-head parameter and a complement-head parameter. Sentences contain a subject (the specifier), a verb, and an optional object (the complement). The parameters thus define the four languages shown in GW's Table 2, presented here as Table 5.

GW (1994:416) observed that "the pattern SV is a global trigger for the value spec-first of the specifier-head ordering parameter" and "the pattern VS is a global trigger for the spec-final value of this parameter." Note that by our definitions (1) and (2) in section 1.2.1, each of these triggers is globally valid, since it signals the correct parameter value for any language that contains it and is also globally available, since it is present in every language in the domain that has that value.

By contrast, GW (1994:416) maintained that "the complement-head parameter has no global triggers in this simple parameter space," given that there is no specific sentence that is present in both of the comp-first languages and in only those, and likewise none that is present in both and only the comp-final languages. The lack of global triggers follows from definition (GW1), which requires global availability as well as validity, and does not allow for type (b) locally available globally valid triggers. For instance, the sentence OVS is a globally valid trigger, in

⁵⁴The wording "... is a sentence S from the target grammar L such that S is..." in (GW1) and (GW2) seems to mix reference to grammars and reference to languages. Our own definitions avoid this, but it is not a matter of importance to the discussion that follows.

our sense, for comp-first because it yields the correct parameter value for any language in which it occurs in this domain, but it is not a globally *available* trigger for comp-first because there is a comp-first language that lacks it. Similarly, VOS is a globally valid trigger for comp-final because it yields the correct parameter value for any language it occurs in, but it is only locally available because there is a comp-final language that lacks it. Thus GW's definitions miss out on the fact that the complement-head parameter does have fully unambiguous triggers in all languages in this domain, though they are not the same triggers across the board.

Definition (GW2) characterizes locally valid triggers, type (c), which can be illustrated in a 3-parameter domain that GW also employed (1994:417 ff.). It consists of the 2-parameter domain above plus a parameterized Verb-Second (V2) transformation (as well as a number of additional constituent types such as indirect objects and adverbs). For the positive value +V2, the sentence SV is a locally valid trigger: SV is a valid trigger for +V2 just in case it has already been established that the target language is spec-final (i.e., the subject underlyingly follows the VP). The sentence SV occurs also in spec-first languages, but there it is not a valid trigger for +V2 because it is compatible not only with +V2 (cf. German) but also with -V2 in a spec-first language (cf. English). In the terminology we have used in the discussion above, the sentence SV in this mini-domain is a *conditioned* trigger for the positive value of the V2 parameter, conditioned by the spec-final value of the Specifier-head parameter. Conditioned triggers, as we have noted, are potentially dangerous because they can lead to error if the conditioning parameters are mis-set. But for an error-free learner they may usefully supplement globally valid triggers.

Our definitions depart from GW's in another significant respect: the term "sentence pattern" replaces the term "sentence" in (GW1) and (GW2). GW employed the term "sentence pattern" frequently when applying their definitions to specific examples, and we consider it to be more appropriate. Requiring triggers to be sentences guarantees that there will be few globally available triggers in any domain of interest, including the domain of natural languages. (There are exactly 70 in CoLAG out of a total of 48,086 distinct sentences; 35 for Head-initial and 35 for Head-final.⁵⁵) This is because sentences in different languages differ from each other in many specific ways due to the influence of other parameters (even setting aside lexical differences), making it unlikely that there is any sentence that is present in every language with parameter value $P_i(v)$.⁵⁶

A sentence pattern in our sense differs from a sentence in two ways: it may characterize only a part of a sentence; and it may characterize it abstractly, using variables that generalize over the sequence of terminal symbols in particular sentences. In consequence, a particular

All CoLAG languages contain either (i) or (ii) and other such sentences. Thus both the Head-initial parameter value and the Head-final parameter value have globally available triggers.

⁵⁵They are all imperative, a sentence type in which no movement operations apply. Therefore the surface sequence necessarily reflects the underlying order of head and complement. See examples (i) and (ii), which trigger the values *initial* and *final* respectively of the CoLAG parameter Headedness in IP, NegP, VP, PP.

⁽i) Verb O1 [ILLOC IMP]

⁽ii) O1 Verb [ILLOC IMP]

⁵⁶Defining triggers as sentences also entails that a parameter value will have an infinite number of distinct triggers within each language if there is no bound on the length or complexity of sentences that instantiate the relevant pattern (e.g., preposition stranding).

complete sentence that would not qualify as anything more than a locally available trigger for $P_i(v)$ according to GW's definitions may well be promotable to global trigger status by our definitions if it can be seen as an instance of a general pattern that abstracts away from irrelevant sentence-particular or language-particular properties. Consider in this regard GW's judgment that their 2-parameter mini-domain (Table 5 above) had no global trigger for compfirst, but only the local triggers OVS and SOV (contrasting with VOS and SVO for comp-final). Comparing these, it is easy to see that OV word order is a globally available trigger for compfirst in that domain—as long as triggers are not required to be complete specific sentences. This also seems linguistically more appropriate, in that it focuses on the property that is actually relevant to the parameter setting: the reason *why* the sentence OVS is a trigger for comp-first is that it has OV word order. This is a trivially simple example, of course, but the CoLAG domain offers less obvious cases. For instance what makes a sentence such as (26) a trigger for +NullTopic is just that it contains a preposition *P* without its object *O3* (which could have been deleted only from topic position).

(26) S never Verb O1[+WH] P Adv

Thus, defining triggers as sentence patterns yields more abundant global E-triggers and more linguistically relevant E-triggers of a kind that a syntactician might point to in discussion of the acquirability of a newly proposed parameter. In this respect our revised definitions bring the formal theory of triggers into closer line with ways of thinking in linguistics and psycholinguistics, and to that extent they are worthwhile. Nevertheless, it is still an open question whether trigger characterizations at this level play a part in the working machinery for syntactic parameter setting by children. Does a learner need to be given these characterizations as a guide to what to look for in input sentences? In section 3.2 we noted that there can be alternative answers to this question.