ELSEVIER

Cognition 61 (1996) 161–193

COGNITION

A language learning model for finite parameter spaces

Partha Niyogi*, Robert C. Berwick

Center for Biological and Computational Learning, Massachusetts Institute of Technology E25-201, Cambridge, MA 02142, USA

Abstract

This paper shows how to formally characterize language learning in a finite parameter space, for instance, in the principles-and-parameters approach to language, as a Markov structure. New language learning results follow directly; we can explicitly calculate how many positive examples on average ("sample complexity") it will take for a learner to correctly identify a target language with high probability. We show how sample complexity varies with input distributions and learning regimes. In particular we find that the average time to converge under reasonable language input distributions for a simple three-parameter system first described by Gibson and Wexler (1994) is psychologically plausible, in the range of 100–150 positive examples. We further find that a simple random step algorithm – that is, simply jumping from one language hypothesis to another rather than changing one parameter at a time – works faster and always converges to the right target language, in contrast to the single-step, local parameter setting method advocated in some recent work.

1. Introduction: language acquisition in finite parameter spaces

With the advent of the "principles-and-parameters" approach to language (Chomsky, 1981), the question of language learnability can again be raised in a new context. We take "principle-and-parameters" approaches to encompass such diverse linguistic theories as government and binding theory (including its current minimalist incarnations); head-driven phrase structure grammar (HPSG); and current lexical-functional grammar (LFG). In each of these frameworks it seems to be possible to characterize the class of possible target (learnable) grammars (or languages) as fixed by the parametric variation of a finite number of discontinuous variables. Learning a grammar (language) involves fixing the values of these parameters. The notion of a finite parameterization for grammars and learning extends even to phonological systems, such as stress, described by Dresher and

* Corresponding author. Fax: 1 617 253 5060; e-mail: pn@ai.mit.edu, berwick@ai.mit.edu.

PII S0010-0277(96)00718-4

Kaye (1990), and perhaps even lexical knowledge, if parameterized along the lines discussed by Hale and Keyser (1993) and others. The classic example of a finite parameterization, common to GB, HPSG, and LFG, is X-bar theory: each of the theories above assumes a basic phrase structure determined by an unordered template of the form {*Head, Complement*}, where *Head* is one of the lexical categories Noun, Verb, Preposition, ..., and *Complement* is some list of (possibly argument) phrases. (Under some recent accounts, Heads may be extended to non-lexical or functional categories such as Inf(lection) or Tense, but we shall not depend on such details in the sequel.) By fixing the order *Head first*, we get languages like English, French, and so forth; the other possibility, *Head final*, applies to languages like Japanese, German, and the like.

However, as emphasized particularly by Wexler in a series of works (Hamburger and Wexler, 1977; Wexler and Culicover, 1980; and Gibson and Wexler, 1994), the finite character of these hypothesis spaces does *not* solve the language acquisition problem. As Chomsky noted in *Aspects of the Theory of Syntax* (Chomsky, 1965), the key point is how the space of possible grammars – even if finite – is "scattered" with respect to the primary language input data. It is logically possible for just two grammars (or languages) to be so near each other that they are not separable by psychologically realistic input data. This was the thrust of Wexler and Hamburger and Wexler and Culicover's earlier work on the learnability of transformational grammars from simple data (with at most two embeddings). Note that this is essentially a question of how many examples it will take to identify a target language – in other words, a sample complexity problem. More recently, Gibson and Wexler (1994) show that more subtle difficulties can arise in the principles-and-parameters framework: given a linguistically plausible three-parameter space, some target languages are not learnable from some initial grammatical hypotheses, given plausible positive-only input.

This article provides a complete mathematical model for analyzing Chomsky's informal notion of "scattering" and particular learnability results like those of Gibson and Wexler in a more general and precise framework. Our central goal is to focus on the question of *convergence time*: how many positive examples will it take to reach a target grammar, under varying assumptions about learning procedures, input data, possible grammars, and so forth. In this sense we can now exactly quantify the amount of data needed to learn (natural) languages.

As a way of organizing our analysis, we may consider the language learning problem to vary along five familiar dimensions: (1) the type of learning algorithm involved; (2) the distribution of the input data; (3) the presence or absence of noise or extraneous examples; (4) the use of memory; (5) the parameterization of the language space itself (the class of possible grammars/languages).

Our central observation is that one can model learning in a finite grammar (language) space by memoryless algorithms (like the Triggering Learning Algorithm and others discussed below) *completely* and *mathematically precisely* by a Markov process. The states in the Markov process denote possible languages or grammars (henceforth, we shall use the terms "language" and "grammar" interchangeably where no confusion would arise). We can then apply standard Markov theory to exactly compute the number of examples required to attain a

target language with high probability, given a learner's initial state. In some cases, this number is unbounded: that is, previously established non-learnability results fall out as a specific case of this so-called *sample complexity* question. For our purposes, we define the sample complexity of the language learning problem to be the number of positive examples the learner needs in order to identify the target grammar with high (greater than $1 - \delta$) probability. We regard this analysis as the next step in refining the general learnability questions posed by Chomsky, Wexler, and others. Previous research has usually addressed only the question of convergence in the limit without probing the equally important question of sample complexity. However, plainly it is of not much use that a learner can acquire a language if sample complexity is extraordinarily high, hence psychologically implausible. Sample complexity is also closely related to Chomsky's question of grammar "scattering," as we shall see.

For our analysis we choose as a concrete starting point the Gibson and Wexler (1994) Triggering Learning Algorithm (TLA). In our five-dimensional taxonomy of language learning systems, this one corresponds to (1) a local hill climbing[1] search algorithm; (2) a uniform sentence distribution over unembedded (degree-0) sentences; (3) no noise; (4) a three-way parameterization using mostly X-bar theory; and (5) memoryless (non-batch) learning. Following our analysis of this learning system, we consider variations in learning algorithms, sentence distribution, and noise.

2. Formal analysis of the triggering learning algorithm

Let us start with the TLA. We first show that this algorithm and others like it are completely modeled by a Markov chain. We explore the basic computational consequences of this fundamental fact, including some surprising results about sample complexity and convergence time, the dominance of random walk over hill climbing, and the applicability of these results to actual child language acquisition and possibly language change.

2.1. Background

Following Gold (1967) and Gibson and Wexler (1994) the basic framework is that of *identification in the limit*. We assume some familiarity with Gold's assumptions. The learner receives an (infinite) sequence of (positive) example sentences from some target language. After each example presentation, the learner either (i) stays in the same state, or (ii) moves to a new state (changes its

[1] The TLA is an online algorithm. After every example, one can imagine constructing a hill over the hypothesis space. For every hypothesis h in this space, the height of the hill is 1 if the example is analyzable by that hypothesis, otherwise, the height is 0. The TLA finds the highest point on this hill *around* its current hypothesis and moves to that highest point. In this sense, it does local hill climbing. Also, note that the hill changes after every example. The 'hills' after every example could be alternatively views as stochastic samples of a single global objective function that is being optimized but an elaboration of this point would divert us further afield.

parameter settings). If after some finite number of examples the learner converges to the correct target language and never changes its guess, then it has correctly identified the target language in the limit; otherwise, it fails.

In the Gibson and Wexler model (and others) the learner obeys two additional fundamental constraints: (1) the *single-value constraint* – the learner can change only one parameter value each step; and (2) the *greediness constraint* – if the learner is given a positive example it cannot recognize and changes one parameter value, finding that it can accept the example, then the learner retains that new value. The TLA can then be precisely stated as follows. See Gibson and Wexler (1994) for further details.

- [Initialize] Step 1. Start at some random point in the (finite) space of possible parameter settings, specifying a single hypothesized grammar with its resulting extension as a language.
- [Process input sentence) Step 2. Receive a positive example sentence s_i at time t_i (examples drawn from the language of a single target grammar, $L(G_t)$), from a uniform distribution on the degree-0 sentences of the language (we relax this distributional constraint later on).
- [Learnability on error detection] Step 3. If the current grammar parses (generates) s_i, then go to Step 2; otherwise, continue.
- [Single-step hill climbing] Step 4. Select a single parameter uniformly at random, to flip from its current setting, and change it (0 mapped to 1, 1 to 0) *iff that change allows the current sentence to be analyzed.*

Of course, this algorithm never halts in the usual sense. Gibson and Wexler aim to show under what conditions this algorithm converges "in the limit" – that is, after some number, m, of steps, where m is unknown, the correct target parameter settings will be selected and never changed. They investigate the behavior of the TLA on a linguistically natural, three-parameter subspace (of the complete linguistic parametric space which involves many more parameters). We review their subspace immediately below. Note that a *grammar* in this space is simply a particular n-length array of 0's and 1's; hence there are 2^n possible grammars (languages). Gibson and Wexler's surprising result is that the simple three-parameter space they consider is unlearnable in the sense that positive-only examples can lead to *local maxima* – incorrect hypotheses from which a learner can never escape. More broadly, they show that learnability in such spaces is still an interesting problem, in that there is a substantive learning theory concerning feasibility, convergence time, and the like, that must be addressed beyond traditional linguistic theory and that might even choose between otherwise adequate linguistic theories.

2.1.1. Remark

Various researchers (Clark and Roberts, 1993; Frank and Kapur, 1992; Gibson and Wexler, 1994; Lightfoot, 1991) have explored the notion of *triggers* as a way to model parameter space language learning. For these researchers, triggers are essentially sentences from the target that cannot be analyzed by the learner's

current grammatical hypothesis and thereby indirectly inform it about the correct hypothesis. Gibson and Wexler suggest that the existence of triggers for every (hypothesis, target) pair in the space suffices for TLA learnability to hold. As we shall see later, one important corollary of our stochastic formulation shows that this condition does *not* suffice. In other words, even if a *triggered* path exists from the learner's hypothesis language to the target, the learner might, with high probability, not take this path, resulting in non-learnability. A further consequence is that many of Gibson and Wexler's proposed cures for non-learnability in their example system, such as a "maturational" ordering imposed on parameter settings, simply do not apply. On the other hand, this result reinforces Gibson and Wexler's basic point that apparently simple parameter-based language learning models can be quite subtle – so subtle that even a seemingly complete computer simulation can fail to uncover learnability problems.

2.2. The Markov formulation

Given this background, we turn directly to the formalization of parameter space learning in terms of Markov chains. This formalization is in fact suggested but left unpursued in a footnote of Gibson and Wexler (1994).

2.2.1. Parameterized grammars and their corresponding Markov chains

Consider a parameterized grammar (language) family with n parameters. We picture the 2^n-size hypothesis space as a set of points; see Fig. 1 for the three-parameter case. Each point corresponds to one particular vector of parameter settings (languages, grammars). Call each point a *hypothesis state* or simply *state* of this space. As is conventional, we define these languages over some alphabet[2]. One state is the target language (grammar). Without loss of generality, we may place the (single) target language at the center of this space. Since by the TLA the learner is restricted to moving at most 1 binary value in a single step, the theoretically possible transitions between states can be drawn as (directed) lines connecting parameter arrays (hypotheses) that differ by at most one binary digit (a 0 or a 1 in some corresponding position in their arrays). (Recall that the distance between the grammars in parameter space is the so-called *Hamming distance*.)

We may further place *weights*, b, on the transitions from state i to state j. These correspond to the probabilities that the learner will move from hypothesis state i to state j. In fact, given a distribution over the target languages $L(G)$, we can carry out an exact calculation of these transition probabilities themselves. Thus, we can picture the TLA learning space as a directed, labeled graph V with 2^n vertices[3].

As mentioned, not all these transitions will be possible in general. For example,

[2] Following standard notation, Σ denotes a finite alphabet and Σ^* denotes the set of all finite strings (sentences) obtained by concatenating elements of Σ.

[3] Gibson and Wexler construct an identical transition diagram in the description of their computer program for calculating local maxima. However, this diagram is not explicitly presented as a Markov structure; it does not include transition probabilities, which we shall see lead to crucial differences in learnability results. Of course, topologically, both structures must be identical.

by the single value hypothesis, the system can only move 1 bit at a time. Also, by assumption, only differences in surface strings can force the learner from one hypothesis state to another. For instance, if state i corresponds to a grammar that generates a language that is a proper subset of another grammar hypothesis j, there can never be a transition from j to i, and there might be one from i to j. Further, it is clear that once we reach the target grammar there is nothing that can move the learner from this state, since no positive evidence will cause the learner to change its hypothesis. Thus, there must be a loop from the target state to itself, and no exit arcs. In the Markov chain literature, this is known as an *Absorbing State* (*A*). Obviously, a state that leads only to an absorbing state will also drive the learner to that absorbing state. If a state corresponds to a grammar that generates some sentences of the target there is always a loop from that state to itself, with some non-zero probability. Finally, let us introduce the notion of a closed set of states C to be any proper subset of states in the Markov chain such that there is no are from any of the states in C to any state outside C in the Markov chain (see Isaacson and Madsen, 1976; Resnick, 1992 and later in this paper for further details). In other words, it is a set of states from which there is no way out to other states lying outside this set. Clearly, a closed set with only one element (state) is an absorbing state.

Note that in the absence of noise, the target state is always an Absorbing State in the systems under discussion. This is because once the learner is at the target grammar, all examples it receives are analyzable and it will never exit this state. Consequently, the Markov chains we will consider always have at least one *A*. Given this formulation, one can immediately give a very simple learnability theorem stated in terms of the Markov chains corresponding to finite parameter spaces and learning algorithms[4]. We do this below.

2.2.2. Markov chain criteria for learnability

We argued how the behavior of the Triggering Learning Algorithm can be formalized by a Markov chain. This argument will be formally completed by providing details of the transition probabilities in a little while. While the formalization is provided for the TLA, every memoryless learning algorithm $\mathcal{A}$ for identifying a target grammar g_f from a family of grammars $\mathcal{G}$ via positive examples can be formalized as a Markov chain M. In particular, M has as many states as there are grammars in $\mathcal{G}$ with the states in M being in 1–1 correspondence with grammars $g \in \mathcal{G}$. The target grammar g_f corresponds to a target state s_f of M. We call M the Markov chain *associated with* the triple ($\mathcal{A}$, $\mathcal{G}$, g_f), and the triple itself a *memoryless learning system*, or *learning system* for short. The triple decides completely the topology of the chain. The transition probabilities of the chain are related to the probability P with which sentences are presented to the learner.

An important question of interest is whether or not the learning algorithm A

[4] Note that learnability requires that the learner converge to the target state from *any* initial state in the system.

identifies the target grammar in the limit. The following theorem shows how to translate this conventional Gold-learnability criterion for identifiability in the limit into a corresponding Markov chain criterion for such memoryless learning systems.

We first recall the familiar definition for Gold-learnability:

Definition 1 *Consider a family of grammars $\mathcal{G}$, a target grammar $g_t \in G$, and a learning algorithm $\mathcal{A}$ that is exposed to sentences from the target according to some arbitrary distribution P. Then g_t is said to be* ***Gold-learnable*** *by $\mathcal{A}$ for the distribution P if and only if $\mathcal{A}$ identifies g_t in the limit with probability 1.*

A family of grammars $\mathcal{G}$ is Gold-learnable if and only if each member of G is Gold-learnable.

The learnability theorem below says that if a target grammar $g_t \in \mathcal{G}$ is to be Gold-learnable by $\mathcal{A}$, then the Markov chain associated with the particular learning system must be restricted in a certain way. To understand the statement of the theorem, we first recall the related notions of *absorbing state* and *closed set of states*. Intuitively, these terms refer to Markov chain connectivity and associated probabilities: an absorbing state has no exit link to any other state, while a *closed set of states* is the extension of the absorbing state notion to a *set* of states. They have already introduced informally in the earlier section for pedagogical reasons, They are reproduced here again for completeness of the current formal account.

Definition 2 *Given a Markov chain M, an* ***absorbing state*** *of M is a state $s \in M$ that has no exit arcs to any other states of M.*

Since by the definition of a Markov chain the sum of the transition probabilities exiting a state must equal one, it follows that an absorbing state must have a self-loop with transition probability 1. In a learning system that makes transitions based on error detection, the target grammar will be an absorbing state, because once the learner reaches the target state, all examples are analyzable and the learner will never exit that state.

Definition 3 *Given a Markov chain M, a* ***closed set of states*** *(C) is any proper subset of states in M such that there is no arc from any of the states in C to any state not in C.*

If two states belong to the same closed set C then there may be transitions from one to the other. Further, there can be transitions from states *outside* C to states *within* C. However, there cannot be transitions from states within C to states outside C. Clearly, an absorbing state represents the special case of a closed set of states consisting of exactly one element, namely, the absorbing state itself.

We can now state the learnability theorem.

Theorem 1 *Let $\langle A, \mathcal{G}, g_t \in \mathcal{G} \rangle$ be a memoryless learning system. Let sentences*

from the target be presented to the learner according to the distribution P and let M be the Markov chain associated with this learning system. Then the target g_f is Gold-learnable by $\mathcal{A}$ for the distribution P if and only if M is such that every closed set of states in it includes the target state corresponding to g_f.

Proof This has been relegated to the appendix for continuity of reading.

Thus, if we are interested in the Gold-learnability of a memoryless learning system, one could first construct the Markov chain corresponding to such a system and then cheek to see if the closed sets of the chain satisfy the conditions of the above theorem. If and only if they do, the system is Gold-learnable.

We now provide an informal example of how to construct a Markov chain for a parametric family of languages. This is followed by a formal account of how to compute the transition probabilities of the Markov chain. Finally, we note some additional properties of the learning system that fall out as a consequence of our analysis. For example, our analysis is consistent with the subset principle, it can handle a variety of algorithms, and even noise.

2.2.3. Example

Consider the following three-parameter system studied by Gibson and Wexler (1994). Its binary parameters are: (1) Spec(ifier) first (0) or last (1); (2) Comp(lement) first (0) or last (1); and Verb Second constraint (V2) does not exist (0) or does exist (1). Following standard linguistic convention, by *Specifier* we mean the part of a phrase that "specifies" that phrase, roughly, like *the old* in *the old book*; by *Complement* we mean roughly a phrase's arguments, like *an ice-cream* in *John ate an ice-cream* or *with envy* in *green with envy*. There are also seven possible "words" in this language: S, V, O, O1, O2, Adv, and Aux, corresponding to Subject, Verb (Main), Object, Direct Object, Indirect Object, Adverb, and Auxiliary Verb. There are 12 possible surface strings for each (−V2) grammar and 18 possible surface strings for each (+V2) grammar if we restrict ourselves to unembedded or "degree-0" examples for reasons of psychological plausibility (see Wexler and Culicover, 1980; Lightfoot, 1991; and Gibson and Wexler, 1994 for discussion). Note that the "surface strings" of these languages are actually *phrases* such as [Subject, Verb, Object] as in *John ate an ice-cream.* Fig. 3 of Gibson and Wexler summarizes the possible binary parameter settings in this system. For instance, parameter setting #5 corresponds to the array [0 1 0] = Specifier first, Comp last, and −V2, which works out to the possible basic English surface phrase order of Subject–Verb–Object (SVO). As shown in Gibson and Wexler's Fig. 3, the other possible arrangements of surface strings corresponding to this parameter setting include S V; S V O1 O2 (two objects, as in *give John an ice-cream*); S Aux V (as in *John will eat*); S Aux V O; S Aux V O1 O2; Adv S V (where Adv is an Adverb, like *quickly*); Adv S V O; Adv S V O1 O2; Adv S Aux V; Adv S Aux V O; and Adv S Aux V O1 O2.

2.2.4. The Markov chain for the three-parameter example

Suppose the target language is SVO (Subject Verb Object, or "English" setting #5 = [0 1 0]). Within the Gibson and Wexler three-parameter system, there are $2^3 = 8$ possible hypotheses, so we can draw this as an 8-point Markov configuration space, as shown in Fig. 1. The shaded rings represent increasing distance in parameter space (Hamming distances) from the target. Each labeled circle is a Markov state, a possible array of parameter settings or grammar, hence specifies a possible target language. Each state is exactly 1 binary digit away from its possible transition neighbours. Each labeled, directed are between the points is a possible

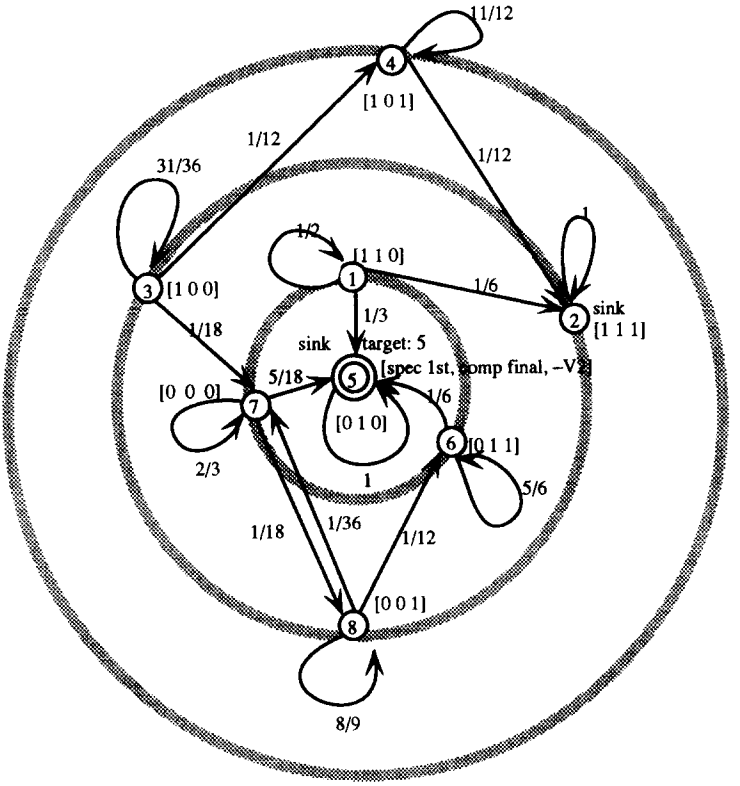

Fig. 1. The eight parameter settings in the GW example, shown as a Markov structure. Directed arrows between circles (states, parameter settings, grammars) represent possible non-zero (possible learner) transitions. The target grammar (in this case, number 5, setting [0 1 0]), lies at dead center. Around it are the three settings that differ from the target by exactly one binary digit; surrounding those are the three hypotheses two binary digits away from the target; the third ring out contains the single hypothesis that differs from the target by three binary digits. Note that the learner can either stay in the same state or step in or out one ring (binary digit) at a time, according to the single-step learning hypothesis; but some transitions are not possible because there is no data to drive the learner from one state to the other under the TLA. Numbers on the arcs denote transition probabilities between grammar states; these values are not computed by the original GW algorithm. The next section shows how to compute these values, essentially by taking language set intersections.

transition from state i to state j, where the labels are the transition probabilities; note that the probabilities from all arcs exiting a state sum to 1. We shall show how to compute these probabilities immediately below. The target grammar, a double circle, lies at the center. This corresponds to the (English) SVO language. Surrounding the bull's-eye target are the three other parameter arrays that differ from [0 1 0] by one binary digit each; we picture these as a ring 1 Hamming distance away from the target: [0, 1, 1], corresponding to Gibson and Wexler's parameter setting #6 in their Fig. 3 (Spec-first, Comp-final, +V2, basic order SVO+V2); [0 0 0], corresponding to Gibson and Wexler's setting #7 (Spec-first, Comp-first, −V2), basic order SOV; and [1 1 0], Gibson and Wexler's setting #1 (Spec-final, Comp-final, −V2), basic order VOS.

Around this inner ring lie three parameter setting hypotheses, all 2 binary digits away from the target: [0 0 1], [1 0 0], and [1 1 1] (grammars #2, 3, and 8 in Gibson and Wexler's Fig. 3). Finally, one more ring out, three binary digits different from the target, is the hypothesis [1 0 1], corresponding to target grammar 4.

It is easy to see from inspection of the figure that there are exactly two Absorbing States in this Markov chain, that is, states that have no exit arcs with non-zero probability. One absorbing state is the target grammar (by definition). The other absorbing state is state 2 (corresponding to language VOS + V2, i.e., [1 1 1]). Finally, state 4 (parameter setting [1 0 1]), while not an absorbing state in itself, has no path to the target. It has arcs that lead only to itself or to state 2 (an absorbing state which is not the target). These two states correspond to the local maxima at the head of Gibson and Wexler's Fig. 4. Hence this target language is *not* learnable. In addition to these local maxima, the next section below shows that there are in fact other states from which the learner will, with high probability, never reach the correct target.

2.3. Derivation of the transition probabilities for the Markov TLA structure

We have discussed in the previous section how the behavior of the TLA can be modeled as a Markov chain. The argument is incomplete without a characterization of the transition probabilities of the associated Markov chain. We first provide an example and follow it with a formal exposition.

2.3.1. Example

Consider again the three-parameter system in Fig. 1 with target language 5. What is the probability that the learner will move from state 8 to state 6? The learner will make such a transition if it receives a sentence that is analyzable according to the parameter settings of state 6, but not according to the parameter settings of state 8. For example, a sentence of the form (S V O1 O2) as in *Peter gave John an ice-cream* could drive the learner to change its parameter settings from 8 to 6. If one assumes a probability distribution with which sentences from the target are presented to the learner, one could find the total probability measure of all such sentences and use it to calculate the appropriate transition probability.

2.3.2. Formalization

The computation of the transition probabilities from the language family can be done by a direct extension of the procedure given in Gibson and Wexler (1994). Let the target language L_t consist of the strings s_1, s_2, ..., that is,

$$L_t = \{s_1, s_2, s_3, \ldots\}$$

Let there be a probability distribution P on these strings. Suppose the learner is in a state s corresponding to the language L_s. Consider some other state k corresponding to the language L_k. What is the probability that the TLA will update its hypothesis from L_s to L_k after receiving the next example sentence? First, observe that due to the single valued constraint, if k and s differ by more than one parameter setting, then the probability of this transition is zero. In fact, the TLA will move from s to k *only if* the following two conditions are met: (1) the next sentence it receives (say, ω occurring with probability $P(\omega)$) is analyzable by the parameter settings corresponding to k and not by the parameter setting corresponding to s; and (2) the TLA has a choice of n parameters to flip on not being able to analyze ω and it happens to pick the one which would move it to state k.

Event 1 occurs with probability $\Sigma_{\omega \in (Lk \setminus Ls) \cap Lt} P(\omega)$. This is simply the probability measure associated with all strings ω that are both in the target L_t and L_k but not in the language L_s (the learner's currently hypothesized language). Event 2 occurs with probability $1/n$, since the parameter to flip is chosen uniformly at random out of the n possible choices. Thus the co-occurrence of both these events yields the following expression for the total probability of transition from s to k after one step:

$$P[s \rightarrow k] = \sum_{s_j \in (L_k \setminus L_s) \cap L_t} (1/n) P(s_j)$$

Since the total probability over all the arcs out of s (including the self-loop) must be 1, we obtain the probability of remaining in state s after one step as:

$$P[s \rightarrow s] = 1 - \sum_{k \text{ is a neighbour state of } s} P[s \rightarrow k]$$

In other words, the probability of remaining in state s is 1 minus the probability of moving to any of the other (neighboring) states.

Finally, given any parameter space with n parameters, we have 2^n languages. Fixing one of them as the target language L_t we obtain the following procedure for constructing the corresponding Markov chain. Note that this is simply the Gibson and Wexler procedure for finding local maxima, with the addition of a probability measure on the language family.

- [Assign distribution) Fix a probability measure P on the strings of the target language L_t.
- [Enumerate states] Assign a state to each language, that is, each L_i.
- [Normalize by the target language] Intersect all languages with the target

language to obtain for each i, the language $L'_i = L_i \cap L_t$. Thus with state i associated with language L_i, we now associate the language L'_i.

- [Take set differences] For any two states i and k, $i \neq k$, if they are more than 1 Hamming distance apart, then the transition $P[i \rightarrow k] = 0$. If they are 1 Hamming distance apart then $P[i \rightarrow k] = (1/n)P(L'_k \backslash L'_i)$. For $i = k$, we have $P[i \rightarrow i] = 1 - \sum_{j \neq i} P[i \rightarrow j]$.

Remark

This model captures the dynamics of the TLA completely. We note that the learner's movement from one language hypothesis to another is driven by purely extensional considerations – that is, it is determined by set differences between language pairs. A detailed investigation of this point is beyond the scope of this paper. We simply note here that if this extensional calculation is the basis of the learning algorithm, then it is unclear what the notion "trigger" means, because the calculation simply refers to string–language set differences. We shall therefore henceforth place the term "trigger" in quotes. (The same point has been made by Frank and Kapur, 1992 and Dresher, 1994, unpublished.)

2.3.3. Example (continued)

For our three-parameter system, we can follow the above procedure to calculate set differences and build the Markov figure straightforwardly. For example, consider $P[8 \rightarrow 6]$; we compute $(L_6 \backslash L_8) \cap L_5 = \{$S V O1 O2, S Aux V O, S Aux V O1 O2$\}$. This set has three degree-0 sentences. Assuming a uniform distribution on the 12 degree-0 strings of the target L_5, we obtain the value of the transition from state 8 to state 6 to be $1/3\ (3/12) = 1/12$. Further, since the normalized language L'_1 for state 1 is the empty set, the set difference between states 1 and 5 $(L'_5 \backslash L'_1)$ yields the entire target language, so there is a (high) transition probability from state 1 to state 5. Similarly, since states 7 and 8 share some target language strings in common, such as S V, and do not share others, such as Adv S and S V O, the learner can move from state 7 to 8 and back again.

2.3.4. Additional properties of the learning system

Once the mathematical formalization has been given many additional properties of this particular learning system now become evident. For example, an issue that is amenable to analysis in the current formalization has to do with the existence of subset/superset pairs of languages. The existence of such pairs does not alter the procedure by which the Markov chain is computed, nor does it alter the validity of our main learnability theorem. However, it is clear by our analysis, that if the target happens to be a subset language, the superset language will correspond to an absorbing state. This is because all target sentences are analyzable by the superset language and if the error-driven learner happens to be at the state corresponding to it, it will never exit. This additional absorbing state automatically implies non-learnability by our theorem. Consequently the classic results on subset/superset non-learnability all fall out as special cases of our framework. However, following

Gibson and Wexler, we will assume that such complications do not arise in the parametric systems under discussion in the current paper.

It is now easy to imagine other alternatives to the TLA that will avoid the local maxima problem: we can vary any of the five aspects of the language learning models we described at the beginning of this paper. To take just one example, as it stands the learner is allowed to change only one parameter setting at a time. If we relax this condition so that in this situation the learner can change more than one parameter at a time, that is, the learner can conjecture hypotheses far from its current one (in parameter space), then the problem with local maxima disappears. It is easy to see that in this case, there can be only one Absorbing State, namely the target grammar. All other states have exit arcs (under the previous assumption of no subset/superset relations). Thus, by our main theorem, such a system *is* learnable.

As another variant, consider the possibility of noise – that is, occasionally the learner gets strings that are not in the target language. Gibson and Wexler state (footnote 4) that this is not a problem: the learner need only pay attention to frequent data. But this is of course a serious problem for the model; *how* is the learner to "pay attention" to frequent data? Unless some kind of memory or frequency-counting device is added, the learner cannot know whether the examples it receives are noise or not. If the learner is memoryless, then there is always some finite probability, however small, of escaping a local maximum. Clearly, the memory window has to be large enough to ensure that sufficient statistics are computable to distinguish noise from relevant data. A serious investigation of this issue is beyond the scope of this paper.

To explore these and other possible variations systematically, let us return to the 5-way classification scheme for learning models introduced at the beginning of this paper. We consider first details about sample complexity. Next, we turn to questions about the distribution of the input data, and ask how this changes the sample complexity results. We also consider realistic input distributions, namely, some drawn from the CHILDES corpus (MacWhinney, 1990). Finally, we briefly consider issues pertaining to the effective modeling of noise.

3. Convergence times for the markov model

We return first to a more detailed look at convergence time – the sample complexity question. The Markov chain formulation gives us some distinct advantages in theoretically characterizing the language acquisition problem. We have already seen how given a Markov chain one could investigate whether or not every closed set includes the absorbing state corresponding to the target grammar. This is akin to the question of whether any local maxima exist. One could also look at other issues (like stationarity or ergodicity assumptions) that might potentially affect convergence.

Perhaps the most significant advantage of the Markov chain formulation is that it allows us to analyze convergence times. Recall that learnability requires the

learner to converge to the target grammar in the limit. The number of examples it would take to do so is our informal notion of convergence time. This is the same as the notion of sample complexity as far as this paper is concerned. In the next sections we provide some formal ways of characterizing convergence times.

Given the transition matrix of a Markov chain, the problem of how long it takes to converge has been well studied. This question is of crucial importance in learnability. Following Gibson and Wexler, we believe that it is not enough to show that the learning problem is *consistent*, that is, that the learner will converge to the target in the limit. We also need to show that the learning problem is *feasible*, that is, the learner will converge in "reasonable" time. This is particularly true in the case of finite parameter spaces where consistency might not be as much of a problem as feasibility. The Markov formulation allows us to attack the feasibility question. It also allows us to clarify the assumptions about the behavior of data and learner inherent in such an approach. For example, if it turns out that a particular parametric theory requires 30 million sentences to be learnable (as analyzed by the Markov approach), it would almost certainly render the theory inadequate on grounds of feasibility. We have not used the convergence criteria to falsify certain kinds of parametric theories yet, but would like to point out the possibility of doing so.

3.1. Some transition matrices and their convergence curves

Let us consider the example that we looked at informally in the previous section. Here the target grammar was grammar 5 and the L' languages were obtained by taking appropriate set differences as discussed. For simplicity, let us first assume a uniform distribution on the strings in L_5, that is, the probability the learner sees a particular string s_j in L_5 is $1/12$ because there are 12 (degree-0) strings in L_5. We can now compute the transition matrix (shown in Fig. 2), where O's occupy matrix entries if not otherwise specified:

Notice that both states 2 and 5 correspond to Absorbing States. Therefore this

		To							
		L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8
From	L_1	$\frac{1}{2}$	$\frac{1}{6}$			$\frac{1}{3}$			
	L_2		1						
	L_3			$\frac{3}{4}$	$\frac{1}{12}$			$\frac{1}{6}$	
	L_4		$\frac{1}{12}$		$\frac{11}{12}$				
	L_5					1			
	L_6					$\frac{1}{6}$	$\frac{5}{6}$		
	L_7					$\frac{5}{18}$		$\frac{2}{3}$	$\frac{1}{18}$
	L_8						$\frac{1}{12}$	$\frac{1}{36}$	$\frac{8}{9}$

Fig. 2. Transition matrix for the Markov chain when the target is L_5. The element occupying the ith row and jth column indicates the probability of moving *from* L_i *to* L_j in one step. Recall that the chain has eight states in this case. Each state corresponds to a particular language (grammar), L_i in the parametric system.

Markov process suffers from the local maxima problem. Note also (following the previous figure as well) that state 4 only exits to either itself or to state 2. For a given transition matrix T, it might be possible to compute[5]

$$T_\infty = \lim_{m \to \infty} T^m$$

The (i, j) element of $T\infty$ denotes the probability that in the limit, the TLA will be at state j if it started out in state i at time $t=0$. For learnability to hold, irrespective of which state the learner starts in, the probability that the learner is in state 5 should tend to 1 as m goes to infinity. This means that column 5 of $T\infty$ should contain all 1's, and the matrix should contain 0's everywhere else. Actually we find that T^m converges to the matrix shown in Fig. 3 as m goes to infinity:

Examining this matrix, we see that if the learner starts out in states 2 or 4, it will certainly end up in state 2 in the limit. These two states correspond to local maxima grammars in the Gibson and Wexler results; if the learner starts in either of these two states, it will never reach the target. From the matrix we also see that if the learner starts in states 5 through 8, it will certainly converge in the limit to the target grammar.

The situation regarding states 1 and 3 is more interesting, and discernible only with the probabilistic formulation developed here. If the learner starts in either of these states, it will reach the target grammar with probability 3/5 and reach state 2, the other absorbing state with probability 2/5. Thus we see that local maxima are *not* the only problem for learnability. As we see from the transition matrix,

$$T_\infty = \begin{array}{c} \\ \textbf{From} \end{array} \begin{array}{c} \textbf{To} \\ \begin{array}{cccccccc} L_1 & L_2 & L_3 & L_4 & L_5 & L_6 & L_7 & L_8 \end{array} \\ \begin{bmatrix} 0 & \frac{2}{5} & 0 & 0 & \frac{3}{5} & 0 & 0 & 0 \\ & 1 & & & & & & \\ & \frac{2}{5} & & & \frac{3}{5} & & & \\ & 1 & & & & & & \\ & & & & 1 & & & \\ & & & & 1 & & & \\ & & & & 1 & & & \\ & & & & 1 & & & \end{bmatrix} \end{array} \begin{array}{l} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \end{array}$$

Fig. 3. The limiting matrix T_∞ has the following structure, Again, the element occupying the ith row and jth column indicates the probability of moving from L_i to L_j. In this case, the transition probabilities are 'in the limit'. Thus, if the learner started in state 1 (L_1), it will be at state 2 (L_2) with probability 2/5 after infinite example sentences. The target language in this particular case was L_5.

[5] This limiting matrix is guaranteed to exist for learnable parametric systems due to the fact that for such systems the target is an absorbing state and all other states are transient ones. One can determine the conditions for the existence of the limiting matrix in general, though it is much beyond the scope of this article. Most of the matrices discussed here do have limits. At the very least, one can certainly conclude that if the limit does not exist, the system is not learnable. These conclusions can be derived from a direct application of the limiting behavior of Markov chains. For further details, see Resnick (1992).

while it is true that states 2 and 4 will, with probability 1, not converge to the target grammar, it is *also* true that states 1 and 3 will not *necessarily* converge to the target. Thus, the number of "bad" initial hypotheses is significantly larger than the five presented in Fig. 4 of Gibson and Wexler (1994). More precisely, out of the 56 possible initial-target language pairs, 12 result in high-probability non-convergence. This is a remarkably high proportion of non-learnable initial-target pairs, considering that the parameter space defines only eight languages. This new finding is again due entirely to the stochastic framework introduced in the current paper.

The importance of these alternative bad initial hypotheses should not be underestimated. As a result, it is not sufficient to consider just the topological structure of the Markov chain to understand its learnability properties. Pure "reachability" of the target from the initial language hypothesis is *not* enough for learnability. Rather, one must consider the Markov transition matrix in the limit and its actual numerical entries. Consequently, the existence of a chain of "triggers" from a source to target language (grammar) does *not* suffice to guarantee learnability. Grammars 1 and 3 represent initial hypotheses from which a triggered sequence of examples to the target exist, yet from which the target is not learnable in the Gold sense.

Gibson and Wexler rely on the "reachability" property to drive a wedge between Verb Second initial state–target state situations and non-Verb Second initial states. For instance, they note that all the cases of non-reachability occur when the initial state is +Verb Second. They then go on to devise various cures for this situation, some involving parameter acquisition ordering or "maturation": for instance, one can imagine that the learner starts out with just −V2 settings. However, the stochastic formulation in the current paper casts doubt on this analysis and its potential cures, because, as we have just seen, some non-learnable initial states are in fact −V2. Thus their proposals for solving the +V2 local maxima problem can only address part of the problem.

To conclude this section, we consider a transition matrix when the target language is L_1. This has no V2 movement, no local maxima problems, and is actually learnable under our assumptions. Again we assume a uniform distribution on degree-0 strings of the target. The transition matrix for the corresponding Markov chain is shown in Fig. 4.

Here we find that T^m does indeed converge to a matrix with 1's in the first column and 0's elsewhere. Consider the first column of T^m. It is of the form: $(p_1(m), p_2(m), p_3(m), p_4(m), p_5(m), p_6(m), p_7(m), p_8(m))'$. Here p_i denotes the probability of being in state 1 at the end of m examples in the case where the learner started in state i. Naturally we want

$$\lim_{m \to \infty} p_i(m) = 1$$

and for this example this is indeed the case. The next figure (Fig. 5; large dashed curve) shows a plot of the following quantity as a function of m, the number of examples.

From \ To	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8
L_1	1							
L_2	$\frac{1}{6}$	$\frac{5}{6}$						
L_3	$\frac{5}{18}$		$\frac{2}{3}$	$\frac{1}{18}$				
L_4		$\frac{3}{36}$	$\frac{1}{36}$	$\frac{8}{9}$				
L_5	$\frac{1}{3}$				$\frac{23}{36}$	$\frac{1}{36}$		
L_6		$\frac{5}{36}$				$\frac{31}{36}$		
L_7			$\frac{1}{18}$				$\frac{11}{12}$	$\frac{1}{36}$
L_8				$\frac{1}{18}$				$\frac{17}{18}$

Fig. 4. Transition matrix for the Markov chain when the target language is L_1. Again, the element in the ith row and jth column denotes the probability of moving from L_i to L_j in one step. The chain has eight states in all corresponding to the eight grammars (languages) in the parametric system under discussion.

$$p(m) = \min_{1 \le i \le 8} \{p_i(m)\}$$

The quantity $p(m)$ is easy to interpret: $p(m) = 0.95$ means that the probability of converging to the target is at least 0.95 for *any* starting state. Moreover, there is at

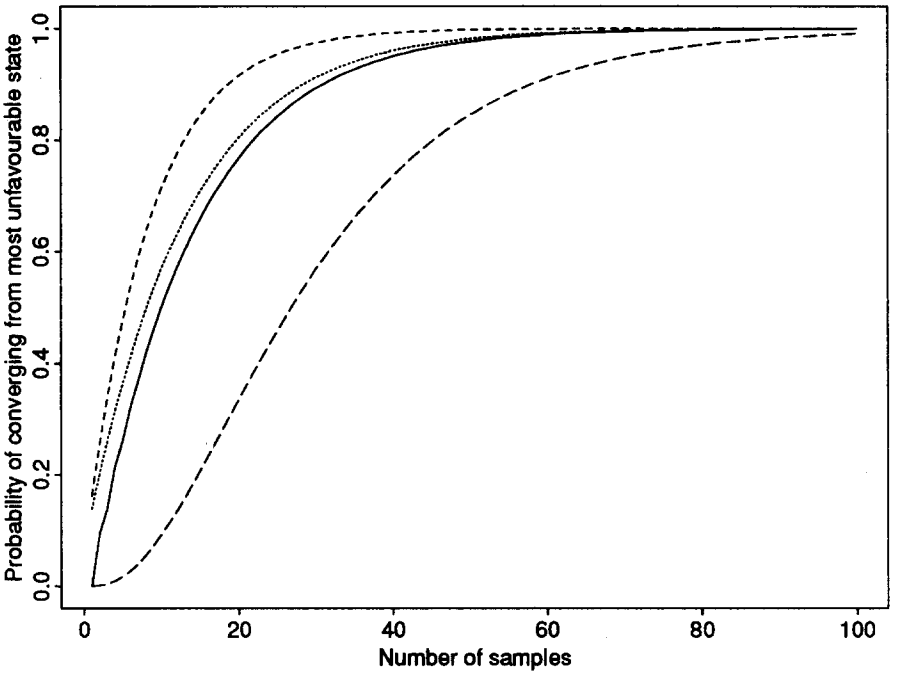

Fig. 5. Convergence rates for different learning algorithms when L_1 is the target language. The curve with the slowest rate (large dashes) represents the Triggering Learning Algorithm. The curve with the fastest rate (small dashes) is a Random Step Algorithm (RSA) with no greediness or single value constraints. Random steps with exactly one of the greediness and single value constraints have performances in between these two and are very close to each other.

least one initial starting state for the learner from which the probability of converging to the target is *exactly* 0.95. Examining the curve, the learner converges with high probability within 100 to 200 (degree-0) example sentences, a psychologically plausible number.

3.2. Changing the algorithm

Having investigated one point in our learning system space, we now proceed to probe variations in the search algorithm. Previously, we considered one change: simply have the learning procedure consider the possibility of moving to an arbitrary state (i.e., not necessarily one that is close to it in parameter space) if it cannot process the current example sentence. We also saw that this change eliminates the local maxima problem. Other simple variations in the algorithm also do better than local hill climbing approach of the TLA. Perhaps the simplest variation is random step: start the learner at a random point in the three-parameter space, and then, if an input sentence cannot be analyzed, pick a state uniformly at random[6] and move there awaiting the next sentence. Note that this regime cannot suffer from the local maxima problem, since there is always some finite probability of exiting a non-target state[7]. We can get two other simple variants of the TLA by dropping in turn the single valued and the greediness constraint. Dropping both yields random step. We exhibit the convergence curves for each of these four algorithms on the three-parameter state space in Fig. 5. Surprisingly, we find that the convergence times for the variants are actually faster[8] than for the TLA. Since the RSA is also superior in that it does not suffer from the same local maxima problem as TLA, the computational support for the TLA is by no means clear. Of course, it may be that future work will yield empirical support for the TLA, in the sense of independent evidence that children do use this procedure (given by the pattern of their errors, etc.), but this evidence is currently lacking, as far as we know.

Now that we have made a first attempt to quantify the convergence time, we examine how this convergence time depends upon the distribution of the data the learner encounters.

[6] Specifically, out of all the 2^n possible states, the learner picks any state to go to with equal $(1/2^n)$ probability.

[7] As discussed previously, we are of course implicitly assuming that there are no subset–superset pairs of languages in the system. Note that if this were not the case, there would always be local maxima, unless the learner observed some kind of subset principle. This is easy to see. Suppose the target was the subset language. Then, the superset language would correspond to a state where every target sentence was analyzable. Hence, the learner would never leave the state once it had reached it. This state would correspond to an additional absorbing state and system is not learnable by Theorem 1.

[8] Janet Fodor and an anonymous reviewer have noted that it is conceivable that the TLA might take more examples but make fewer mind changes than the RSA. This is possibly because the TLA can't move arbitrarily far in one step, and might spend a lot of time staying in one place without moving while the RSA might have wildly fluctuating hypotheses from step to step while converging to the target in a smaller number of steps. This conjecture can of course be tested using the Markov tools developed here but we do not report any systematic investigations of the issue here.

3.3. Changing the distributional assumptions

In an earlier section we assumed that the data was uniformly distributed. We computed the transition matrix for a particular target language and showed that convergence times were of the order of 100–200 samples. In this section we show that the convergence times depend crucially upon the distribution – a point that might seem obvious, but that can be quantified as we show below. In particular, we can choose a "malicious" distribution that will make the convergence time as large as we want. To meet the requirements of a psychologically plausible sample size, one therefore needs to put constraints on the distributions with which sentence-types from the target are presented to the learner. This implies that a realistic study of the Markov/TLA type models will depend on a more careful analysis of language input to children since this class of models is crucially sensitive to the extensional properties of set differences between languages (sets of sentences).

As before, we consider the situation where the target language is L_1. There are no local maxima problems for this choice. To illustrate the dependence of convergence times on the distribution, it is convenient to let the distribution be parameterized by the variables a, b, c, d where

$$a = P(A = \{\text{Adv} - V - S\})$$

$$b = P(B = \{\text{Adv} - \text{V} - \text{O} - \text{S}, \text{Adv} - \text{Aux} - \text{V} - \text{S}\})$$

$$c = P(C = \{\text{Adv} - \text{V} - \text{O1} - \text{O2} - \text{S}, \text{Adv} - \text{Aux} - \text{V} - 0 - \text{S}, \text{Adv} - \text{Aux} - \text{V} - \text{O1} - \text{O2} - \text{S}\})$$

$$d = P(D = \{\text{V} - \text{S}\})$$

Thus each of the sets A, B, C and D contain different degree-0 sentences of L_1. Clearly the probability of the set $L_1 \setminus \{A \cup B \cup C \cup D\}$ is $1-(a+b+c+d)$. The elements of each subset of L_1 are equally likely with respect to each other. Setting positive values for a, b, c, d such that $a+b+c+d<1$ now defines a unique probability for each degree(0) sentence in L_1. For example, the probability of Adv V O S is $b/2$, the probability of Adv Aux V O S is $c/3$, that of V O S is $(1-(a+b+c+d))/5$ and so on. We can thus obtain the transition matrix corresponding to this distribution as shown in Table 1.

Let us compare this matrix with that obtained with a uniform distribution on the sentences of L_1 in the earlier section. This matrix has non-zero elements (transition probabilities) exactly where the earlier matrix had non-zero elements. In other words, the topology of the chain remains the same; its learnability properties are not affected. However, the value of each transition probability now depends upon a, b, c, and d. In particular if we choose $a=1/12$, $b=2/12$, $c=3/12$, $d=1/12$

Table 1
Transition matrix corresponding to a parameterized choice for the distribution on the target strings. In this case the target is L_1 and the distribution is parameterized according to section 3.2.

	L_1	L_2	L_3	L_4	L_5	L_6	L_7	L_8
L_1	1							
L_2	$\frac{1-a-b-c}{3}$	$\frac{2+a+b+c}{3}$						
L_3	$\frac{1-a-d}{3}$		$\frac{2+a+d-b}{3}$	$\frac{b}{3}$				
L_4			$\frac{c}{3}$	$\frac{d}{3}$	$\frac{3-c-d}{3}$			
L_5	$\frac{1}{3}$				$\frac{2-a}{3}$	$\frac{a}{3}$		
L_6		$\frac{b+c}{3}$				$\frac{3-b-c}{3}$		
L_7			$\frac{a+d}{3}$				$\frac{3-2a-d}{sm}3$	$\frac{a}{3}$
L_8				$\frac{b}{3}$				$\frac{3-b}{3}$

(this is equivalent to assuming a uniform distribution) we obtain the transition matrix for a uniform distribution. Looking more closely at the general transition matrix, we see that the transition probability from state 2 to state 1 is $(1-(a+b+c))/3$. Clearly if we make a arbitrarily close to 1, then this transition probability is arbitrarily close to 0 so that the number of samples needed to converge can be made arbitrarily large. Thus choosing large values for a and small values for b, c, and d will result in exceptionally large convergence times.

What does this mean? Briefly, sample complexity depends crucially upon the distribution and by choosing a highly unfavorable distribution the sample complexity can be increased without limit. Furthermore, the exact nature of this dependence can be quantified in our stochastic formulation. For example, we give the convergence curves calculated for different choices of a, b, c, d in Fig. 6. We see that for a uniform distribution the convergence occurs within 200 samples. By choosing a distribution with $a=0.9999$, the convergence time can be pushed up to as much as 50 million samples. (Of course, this distribution is presumably not psychologically realistic.) For $a=0.99$, the sample complexity is on the order of 100,000 positive examples.

3.4. Natural distributions: the CHILDES corpus

Turning from artificially constructed distributions, it is of some interest to examine the utility of the Markov model using real language distributions, namely, those from the CHILDES database (MacWhinney, 1990). We have carried out preliminary direct experiments using the CHILDES caretaker input to "Nina" (Suppes' journal) and German input to "Katrin." These consist of 43,612 and 632 sentences, respectively. We note, following well-known results, that both corpuses

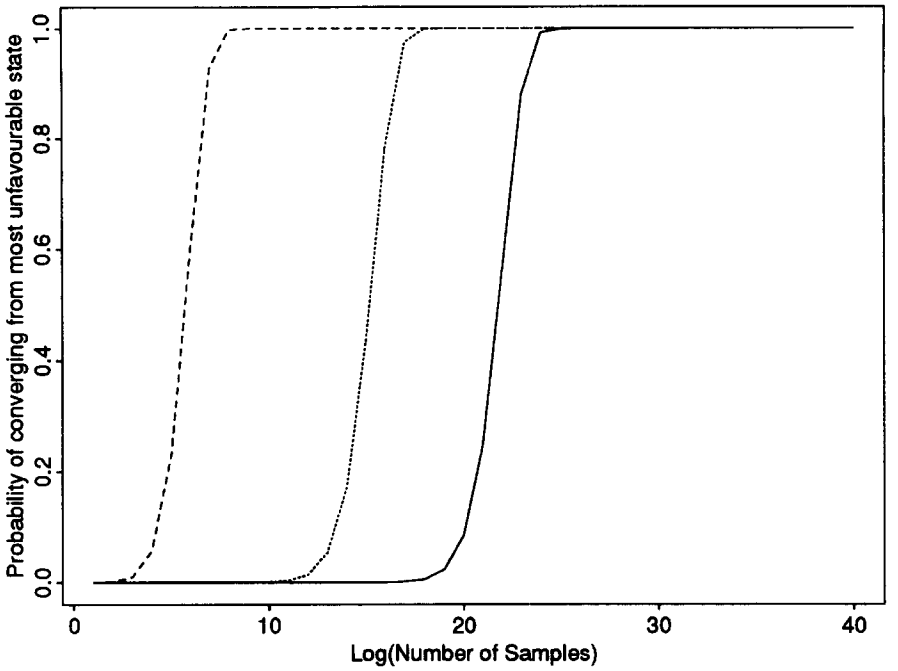

Fig. 6. Rates of convergence for TLA with L_1 as the target language for different distributions. The y-axis plots the probability of converging to the target after m samples and the x-axis is on a log scale, that is, it shows $\log_2(m)$ as m varies. The solid line denotes the choice of an 'unfavorable' distribution characterized by $a = 0.9999$. The dotted line denotes the choice of $a = 0.99$ and the dashed line is the convergence curve for a uniform distribution. For all choices of a, the values of b, c, and d are chosen to make all sentences not in A equally likely.

contain a much higher percentage of auxiliary inversion and *wh*-questions than "ordinary" text (such as the Lund–Oslo–Bergen or Brown corpuses): the "Nina" database has 25,890 auxiliary inversion questions (59.3%) and 11,755 *wh*-questions (26.9%) in caretaker input, as compared to 2506 *wh*-questions or 3.7% of the 53,495 sentences in the LOB corpus; the "Katrin" corpus contains 201 (31.8%) auxiliary inversion questions and 99 (15.7%) *wh*-questions.

To test the convergence of the model, an implemented system was developed using a newer version of the deMarcken partial parser (deMarcken, 1990). Each degree-0 and degree-1 sentence was analyzed as falling into one of the target patterns SVO, S Aux V, etc., as appropriate for the target language. Word part of speech was assumed known. Sentences not parsable into these patterns were discarded (following a presumption that they are "too complex" in some sense, as in Wexler and Culicover (1980); this assumption could of course be relaxed). Some examples of caretaker inputs in both languages follow:

this is a book? what do you see in the book?
how many rabbits?

what is the rabbit doing? (...)
is he hopping? oh. and what is he playing with?
red mir doch nicht alles nach!
ja, die schäwatzen auch immer alles nach (...)

When the system analyzes these sentences and then uses the TLA, we discover that convergence time falls roughly along the TLA rates displayed in Fig. 1: roughly 100 positive (structured) examples to attain asymptotic convergence. Thus, the feasibility of the basic model is confirmed by this simple direct simulation, for both English and German, starting from feasible initial states. Trapping in local maxima is of course observed, if the initial state is unfavorable.

We are continuing to investigate this computer model with additional examples, distributions on the CHILDES inputs, and other languages. One important complication that must be taken into account is the preponderance of auxiliary inversion and *wh*-questions; the parser must be able to detect this pattern, but this largely begs the question of setting the Verb Second parameter. In brief, as far as our tentative simulations indicate, we have not yet arrived at a completely satisfactory account for setting the Verb Second parameter.

3.5. Formal computation of rates of convergence

We have shown how to characterize learnability by Markov chains. Recall that Markov chains corresponding to memoryless learning algorithms have an associated transition matrix T. We saw that T^k was the transition matrix after k examples, and in the limiting case,

$$\lim_{k\to\infty} T^k = T_\infty$$

In general, the structure of $T\infty$, as discussed earlier, determined whether the target grammar was Gold-learnable. The rate at which T converges to $T\infty$ determines the rate at which the learner converges to the target "in the limit". This rate allows us to bound the sample complexity in a formal sense, that is, it allows us to bound the number of examples needed before the learner will be at the target with high confidence. In this section, we develop some formal machinery borrowed from classical Markov chain theory that is useful to bound the rate of convergence of the learner to the target grammar for learnable target grammars. We first develop the notion of an eigenvalue of a transition matrix and show how this can be used to construct an alternative representation of T^k.

We then discuss the limiting distributions of Markov chains from various initial conditions, and finally combine all these notions to formally state some results for the rate at which the learner converges to the target.

3.5.1. Eigenvalues and eigenvectors

Many properties of a transition matrix can be characterized by its eigenvalues and eigenvectors.

Definition 4 *A number* λ *is said to be an eigenvalue of a matrix* T *if there exists some non-zero vector* $\mathbf{x}$ *satisfying*

$$\mathbf{x}T = \lambda\mathbf{x}$$

Such a row vector $\mathbf{x}$ *is called a left eigenvector of* T *corresponding to the eigenvalue* λ. *Similarly, a non-zero column vector* $\mathbf{y}$ *satisfying* $T\mathbf{y} = \lambda\mathbf{y}$ *is called a right eigenvector of* T.

It can be shown that the eigenvalues of a matrix T can be obtained by solving

$$|\lambda I - T| = 0 \tag{1}$$

where I is the identity matrix and $|M|$ denotes the determinant of the matrix M.

3.5.2. Example

Consider the matrix

$$T = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix}$$

Such a matrix could, for example, be the transition matrix for a learner in a parametric space with two grammars, that is, a space defined by *one* boolean valued parameter. In order to solve for the eigenvalues of the matrix, we need to solve

$$|\lambda I - T| = \left| \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} - \begin{bmatrix} \frac{2}{3} & \frac{2}{3} \\ \frac{2}{3} & \frac{2}{3} \end{bmatrix} \right| = 0$$

This reduces to the quadratic equation

$$(\lambda - \frac{2}{3})^2 = \frac{1}{9}$$

which can be solved to yield $\lambda = 1$ and $\lambda = 1/3$ as its two solutions. It can be easily seen that the row vector, $\mathbf{x} = (1, 1)$ is a left eigenvector corresponding to the eigenvalue $\lambda = 1$. As a matter of fact, all multiples of (1, 1) are eigenvectors for this particular eigenvalue. Similarly, it can also be seen that $\mathbf{x} = (1, -1)$ is a left eigenvector for the eigenvalue $\lambda = 1/3$.

In general, for an $m \times m$ matrix T, eq. (1) is an mth order equation and can be solved to yield m solutions (complex-valued) for λ. Two other facts about eigenvalue solutions of such transition matrices are worth noting here.

1. For transition matrices corresponding to finite Markov chains, it is possible to

show that $\lambda = 1$ is always an eigenvalue. Further, it is the largest eigenvalue in that any other eigenvalue, λ, is less than one in absolute value, that is, $|\lambda| < 1$.
2. For transition matrices corresponding to finite Markov chains, the multiplicity of the eigenvalue $\lambda = 1$ is equal to the number of closed classes (see Appendix) in the chain.

In our example above, we do see that $\lambda = 1$ is an eigenvalue. It has multiplicity of 1, indicating that there is only one closed class in the chain; in the example, the class consists of the two states of the chain.

3.5.3. Representation of T^k

The eigenvalues and associated eigenvectors can be used to represent T^k in a form that is convenient for bounding the rate of its convergence to T_∞. This representation is only true for matrices that are of full rank, that is, $m \times m$ matrices that have m linearly independent left eigenvectors.

Let T be an $m \times m$ transition matrix. Let it have m linearly independent left eigenvectors $\mathbf{x}_1, ..., \mathbf{x}_m$ corresponding to eigenvalues $\lambda_1, \lambda_2, ..., \lambda_m$. One could then define the matrix L whose rows are the left eigenvectors of the matrix T. Thus

$$L = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \cdot \\ \cdot \\ \mathbf{x}_m \end{bmatrix}$$

Clearly, since the rows of L are linearly independent, its inverse, L^{-1} exists. It turns out that the columns of L^{-1} are the right eigenvectors of T. Let the ith column of L^{-1} be $\mathbf{y}_i$; that is,

$$L^{-1} = [\mathbf{y}_1 \quad \mathbf{y}_2 \quad \cdot \quad \cdot \quad \cdot \quad \mathbf{y}_m]$$

Now we can represent T^k in a convenient form stated in the following lemma:

Lemma 1 *Let T be an $m \times m$ transition matrix having m linearly independent left eigenvectors, $\mathbf{x}_1, ..., \mathbf{x}_m$ corresponding to eigenvalues $\lambda_1, ..., \lambda_m$. Further let L be the matrix whose rows are the left eigenvectors and let the columns of L^{-1} be $\mathbf{y}_i$'s. Then*

$$T^k = \sum_{i=1}^{m} \lambda_i^k \mathbf{y}_i \mathbf{x}_i$$

Thus, according to the lemma above, T^k can be represented as the linear combination of m fixed matrices $(\mathbf{y}_i \mathbf{x}_i)$. The coefficients of this linear combination are λ_i^k. Clearly, we see that the rate of convergence of T^k is now bounded by the rate of convergence of terms like λ_i^k.

3.5.4. Example (contd.)

Continuing our previous example, we can construct the matrices, L and L^{-1} out of the left eigenvectors. In fact using our solutions from before, we see that

$$L = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \text{ and } L^{-1} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

The rows of L are the $\mathbf{x}_i$'s and the columns of L^{-1} are the $\mathbf{y}_i$'s.

3.5.5. Initial conditions and limiting distributions

Recall that the learner could start in any initial state. One could quantify the initial condition of the learner by putting a distribution on the states of the Markov chain according to which the learner picks its initial state. Let this be denoted by $\prod_0 = (\pi_1(0), \pi_2(0), \ldots, \pi_m(0))$. Thus, $\pi_i(0)$ is the probability with which the learner picks the ith state as the initial state. For example, if the learner were equally likely to start in any state, then $\pi_i(0) = 1/m$ for all i.

The above characterizes the probability with which the learner is in each of the states before having seen any examples. The learner would then move from state to state according to the transition matrix T. After k examples, the probability with which the learner would be in each of the states is given by:

$$\prod_k = \prod_0 T^k$$

Finally, one could characterize the limiting distribution as

$$\prod = \lim_{k \to \infty} \prod_k = \prod_0 T_\infty \tag{2}$$

Clearly, $\prod$ characterizes the probability with which the learner is in each of the states "in the limit". Suppose the target were L_1, and it were Gold-learnable; then the first element of the vector $\prod$ would be 1 and all other elements would be 0. In other words, the probability that the learner is at the target in the limit is 1 and the probability that the learner is at some other state (non-target) in the limit is correspondingly 0.

3.5.6. Rate of convergence

We are interested in bounding the rate at which $\prod_k$ converges to $\prod$. We see that this rate depends on the rate at which T^k converges to T_∞. (eq. 2) which in turn depends upon the rate at which the λ_i^k converges to 0 by Lemma 1 (for $i > 1$). As we have discussed, $\lambda_1 = 1$. Consequently, we can bound the rate of convergence by the rate at which the second largest eigenvalue converges to 0. Thus we can state the following theorem.

Theorem 2 *Let the transition matrix characterizing the behavior of the memoryless learner be T. Further, let T have the eigenvalues, $\lambda_1, \ldots, \lambda_m$, and m*

linearly independent left eigenvectors, $\mathbf{x}_1, \ldots, \mathbf{x}_m$ *and m right eigenvectors* $\mathbf{y}_1, \ldots, \mathbf{y}_m; \lambda_1 = 1$. *Then, the distance between the learner's state after k examples and its state in the limit is given by:*

$$\|\prod_k - \prod\| = \|\sum_{i=2}^{n} \lambda_i^k \prod_0 y_i x_i\| \leq \max_{2 \leq i \leq n} \{|\lambda_i|^k\} \sum_{j=2}^{n} \|\Pi_0 y_j x_j\|$$

Let us first apply this theorem to the illustrative example of this section.

3.5.7. Example (contd.)

We have already solved for the eigenvalues of T and constructed the matrices L and L^{-1}. The rows of L are the row vectors $\mathbf{x}_i$ and the columns of L^{-1} are the column vectors $\mathbf{y}_i$. Assuming that the learner is three times as likely to start in state 1 as compared to state 2, that is, $\prod_0 = (3/4, 1/4)$, we can show that

$$\|\prod_k - \prod\| \leq \left(\frac{1}{3}\right)^k \left(\frac{1}{2}\right)$$

Thus the rate at which the learner converges to the limiting distribution over the state space is of the order of $(1/3)^k$. Note that 1/3 is the second largest eigenvalue of the transition matrix.

3.5.8. Transition matrix recipes

The above discussion allows us to see how one could extract useful learnability properties of the memoryless learner from the transition matrix characterizing the behavior of that learner on the finite parameter space. As a matter of fact, we can now outline a procedure whereby one could check for the learnability and sample complexity of learning in such parameter spaces.

1. Construct the transition matrix T for the memoryless learner according to the arguments developed earlier. Such a matrix has 2^n states if there are n boolean valued parameters in the grammatical theory.
2. Compute the eigenvalues of the matrix T.
3. If the multiplicity of the eigenvalue $\lambda = 1$ is more than one, then there are additional closed classes and by the learnability theorem, the target grammar is not Gold-learnable.
4. If the target is Gold-learnable, and the eigenvectors are linearly independent, then use Theorem 2 to bound the rate of convergence.

Using such a procedure, we can bound the rate of convergence of each of the following learning scenarios for the three parameter syntactic subsystem we have examined in some detail in previous examples. In each case, the target is the language L_1. The learning algorithm is the TLA with different sentence distributions (parameterized by a with b, c, d chosen to make sentences outside of A

equally likely). We also considered the Random Step Algorithm with a uniform sentence distribution. The rate of convergence is denoted as a function of the number of examples.

Learning scenario	Rate of Convergence
TLA (uniform)	$O(0.94^k)$
TLA($a=0.99$)	$O((1-10^{-4})^k)$
TLA($a=0.9999$)	$O((1-10^{-6})^k)$
Random Step	$O(0.89^k)$

4. Modeling noise in the Markov analysis

So far, we have analyzed the learnability and sample complexity of TLA and its variants when the learner was exposed only to positive strings from the target (albeit under varying distributional assumptions). In most human learning scenarios, this presents a problem; children surely are exposed to noise, possibly due to disfluencies in speech, the presence of foreign speakers, or a variety of other reasons. This results in the child hearing sentences that have not been generated by the target grammar. What effect does this sort of noise have on the learnability of these spaces by TLA-like algorithms? For our purpose, we can effectively model noise by dispensing with the idea of a single state that is the source of sentences in our Markov chain. There might now be multiple sources: consider the case of a child brought up in a linguistic community where most people speak the target language but there are also foreign speakers (or speakers who have internalized different grammars) who also contribute to the sum total of the child's linguistic input on the basis of which he or she forms hypotheses about the target grammar. Mathematically, one could express this idea by associating probability p_i with the ith source which then produces sentences according to a distribution P_i on the sentences of this language L_i. The probability of hearing a sentence ω then is given by

$$P(\omega) = \sum_{i=1}^{2^n} p_i P_i(\omega)$$

One can reasonably imagine that p_t which corresponds to the weight given to the target source (or represents the proportion of the population speaking the "target" grammar) dominates all the other p_i's. Note that this framework of modeling noise essentially captures "grammatical noise", that is, sentences inconsistent with the target grammar but analyzable by some other grammar in the parametric system. Noise of this form is particularly important to study as it might systematically drive the learner to incorrect solutions. Other forms of noise

("ungrammatical") corresponding to sentence forms that are not analyzable under any grammatical hypothesis within the parametric system are not considered here.

Certain computational consequences of this framework are worth highlighting. First, the behavior of TLA-like algorithms can again be analyzed within this Markov framework with exactly the same expressions for the transition probabilities as before. Obviously, this time, the probability of the sentence ω is computed by the above expression. The Markov structure now has no Absorbing States. This is because even if the learner is at the target state, some sentence from some other source which is not analyzable by the current target parameter settings will cause the learner to move out; the self loop probability at the target is not one, and certainly this is so for all the other states. By our main theorem, the system is not learnable; consequently it no longer makes sense to talk of convergence to a unique grammar in the limit. However, there is a sense in which the long-term behavior of the TLA (as the data goes to infinity) assumes importance. It might be possible to characterize the probability that the learner will be in a state s in the limit; specifically after m examples it might be possible to compute the probability vector $\bar{p}(m) = (p_1(m), ..., p_2n(m))$ where $\bar{p}(m)$ converges to a limiting distribution $\bar{p}_\infty$

Now, imagine that the target language was L_i. The probability of being in the target state at some distant time (i.e., after an arbitrary number of examples) is given by $\bar{P}_\infty[i]$. Although this probability will not be equal to 1, as discussed earlier, one would like it to be as high as possible. Since noise can be effectively modeled by our framework, the transition matrices and the limiting distribution $\bar{p}_\infty$ can often be computed. These would clearly depend upon the noise. In fact, as the noise levels increase, the probability $\bar{p}_\infty[i]$ will decrease from 1. In principle, one can formally study this decrease as a function of noise level to characterize how the learning power of the TLA degenerates with increase in noise.

This general characterization allows us to formulate an explicit computational model of language change[9]. Imagine for a moment that the human learner follows a TLA like scheme to search the parameter space for the target grammar. A child then on being exposed to sentences according to the distribution P would with probability $\bar{p}_\infty[i]$ internalize the grammar corresponding to L_i. Taking a demographic perspective on the system, we would expect $\bar{p}_\infty[i]$ of the population of children to have internalized the grammar L_i. This generation of children on maturing to adulthood would now serve as a source of sentences for the following generation according to the distribution $\bar{p}_\infty$ on the different states in the Markov chain. In this fashion, over generations the linguistic composition of the population would evolve. Thus this model makes concrete the suggestion of Lightfoot (1991) regarding a dynamical system model for diachronic syntax change.

[9] Space prohibits a thorough consideration of this subject. See Niyogi and Berwick (1995) for details.

5. Conclusions, open questions, and future directions

The problem of learning parameterized families of grammars has several different dimensions. One needs to investigate the learnability of various language families for algorithms under a variety of distributional assumptions, for different parameterization schemes, for various levels and kinds of noise, and so on. In this paper we have emphasized that it is not enough to merely check for learnability in the limit; one also needs to quantify the sample complexity of the learning problem, that is, how many examples does the learning algorithm need to see in order to be able to identify the target grammar with high confidence. We have presented a Markov formalization of the problem in order to investigate both learnability and sample complexity issues precisely. This model provides us with a useful research tool to explore the issues involved in learning natural languages. While examples were primarily given for the TLA (and some other variants) on a particular three-parameter linguistic subspace, it should be reiterated that any parameterization scheme with a finite number of parameters and several other kinds of algorithms can be usefully studied by such an approach. For example, genetic algorithms, which have been proposed as a possible computational paradigm for language learning and language change (see Clark and Roberts, 1993) can also be easily studied within this same Markov framework.

On studying the performance of the TLA on the three-parameter linguistic subspace previously investigated by Gibson and Wexler, not only were we able to characterize the learnability and sample complexity, but we also found some surprising new results. We found the existence of new problem states: states from which the TLA-based learner would with high probability not converge to the target. The existence of a path from a particular state (hypothesis) to the target, equivalent to the existence of local triggers, is not enough to guarantee learnability. The stochastic formulation suggests one has to go further. Further, our Markov analysis showed that the TLA was suboptimal (for the three-parameter task considered here); for example, the random step algorithm on this space had no local maxima and converged faster.

Several directions for further research naturally arise. As the number of parameters n increases, the size of the corresponding Markov matrix grows as 2^n. Thus in the case of a 10-parameter system as found in models of stress learning (Dresher and Kaye, 1990) the corresponding Markov structure will be a 1024×1024 matrix. We are currently conducting an analysis of this larger system to find its local maxima, analyze its convergence times, and see if its convergence times correspond to what one might find in practice with real stress systems. How the sample complexity scales with the number of parameters is an important question that needs to be addressed. Another interesting direction involves the derivation of a model for language change from the model of language learning. The Markov model for language learning focuses on the individual child and how she/he updates her/his grammatical hypotheses from sentence to sentence. If one analyzes the behavior of a population of such child learners and attempts to

characterize the linguistic composition of the population from generation to generation, a model of language change would emerge. Such a diachronic model has been derived in Niyogi and Berwick (1995). The details of the model rest on the learnability analyses presented in this article.

Acknowledgments

We would like to thank Ken Wexler and Ted Gibson for valuable discussions that led to this work, as well as Noam Chomsky for additional discussion. We also benefited greatly from the comments of issue editor, Michael Brent, and anonymous reviewers of drafts of this article. Leonardo Topa provided essential programming support. All residual errors are ours. This research is supported by NSF grant 9217041-ASC and ARPA under the HPCC program.

Appendix 1

Proof of learnability theorem

To establish the theorem, we recall three additional standard terms associated with the Markov chain states: (1) equivalent states; (2) recurrent states; and (3) transient states. We then present another standard result about the *form* of any Markov chain: its *canonical decomposition* in terms of closed, equivalent, recurrent, and transient states.

Markov state terminology

Definition 5 *Given a Markov chain M, and any pair of states s, t* ϵ *M, we say that s is* **equivalent** *to t if and only if s is reachable from t and t is reachable from s, where by reachable we mean that there is a path from one state to another.*

Two states s and t are equivalent if and only if there is a path from s to t and a path from t to s. Using the equivalence relation defined above, we can divide any M into equivalence classes of states. All the states in one class are reachable (from and to) the states in that class.

Definition 6 *Given a Markov chain M, a state s* ϵ *M is* **recurrent** *if the chain returns to s in a finite number of steps with probability 1.*

Definition 7 *Given a Markov chain M, and a state s* ϵ *M, if s is not recurrent, then s is* **transient.**

We will need later the following simple property about transient states:

Lemma 2 *Given a Markov chain M, if t is a transient state of M, then, for any state s ∈ M*

$$\lim_{n\to\infty} p_{st}(n) = 0$$

where $p_{st}(n)$ denotes the probability of going from state s to state t in exactly n steps.

Proof sketch

Proposition 2.6.3 (p. 88 of Resnick (1992) states that

$$\sum_{n=1}^{\infty} p_{st}(n) < \infty$$

Therefore, $\Sigma p_{st}(n)$ is a convergent series. Thus $p_{st}(n)_{n\to\infty} \to 0$.

Canonical decomposition

A particular Markov chain might have many closed states (Definition 3 of text), and these need not be disjoint; they might also be subsets of each other. However, even though there can be many closed states in a particular Markov chain, the following standard result shows that there is a canonical decomposition of the chain (Lemma 3) that will be useful to us in proving the learnability theorem.

Lemma 3 *Given a Markov chain M, we may decompose M into disjoint sets of states as follows:*

$$M = T \cup C_1 \cup C_2 \ldots$$

where (i) T is a collection of transient states and (ii) the C_i's are closed, equivalence classes of recurrent stages.

Proof sketch

This is a standard Markov chain result; see Corollary 2.10.2 of p. 99 of Resnick (1992).

We can now proceed to a proof of the main learnability theorem.

Formal proof

⇒ We need to show that if the target grammar is learnable, then every closed set in the chain must contain the target state. By assumption, target grammar g_f is learnable. Now assume for sake of contradiction that there is some closed set C that does not include the target state associated with the target grammar. If the learner starts in some $s \in C$, by the definition of a closed set of states, it can never reach the target state. This contradicts the assumption that g_f was learnable.

⇐ Assume that every closed set of the Markov chain associated with the

learning system includes the target state. We now need to show that the target grammar is Gold-learnable. First, we make use of some properties of the target state in conjunction with the canonical decomposition of Lemma 3 to show that every non-target state must be transient. Then we make use of Lemma 2 about transient states to show that the learner must converge to the target grammar in the limit with probability 1.

First, note the following properties of the target state:

1. By construction, the target state is an absorbing state, that is, no other state is reachable from the target state.
2. Therefore, no other state can be in an equivalence relation with the target state and the target state is in an equivalence class by itself.
3. The target state is recurrent since the chain returns to it with probability 1 in one step (the target state is an absorbing state).

These facts about the target state show that the target state constitutes a closed class (say C_i) in the canonical decomposition of M. However, there cannot be any *other* closed class $C_j, j \neq i$ in the canonical decomposition of M. This is because by the definition of the canonical decomposition any other such C_j must be disjoint from C_i, and by the hypothesis of the theorem, such C_j must contain the target state, leading to a contradiction. Therefore, by the canonical decomposition lemma, every *other* state in M must belong to T, and must therefore be a transient state.

Now denote the target state by s_f. The canonical decomposition of M must therefore be in the form:

$$T \cup \{s_f\}$$

Without loss of generality, let the learner start at some arbitrary state s. After any integer number n of positive examples, we know that

$$\sum_{t \in M} p_{st}(n) = 1$$

because the learner has to be in *one* of the states of the chain M after n examples with probability 1. But by the decomposition lemma and our previous arguments $M = T \cup s_f$. Therefore we can rewrite this sum as two parts, one corresponding to the transient states and the other corresponding to the final state:

$$\sum_{t \in T} p_{st}(n) + p_{ssf}(n) = 1$$

Now take the limit as n goes to infinity. By the transient state lemma, every $p_{st}(n)$ goes to zero for $t \in T$. There are only a finite (known) number of states in T. Therefore, $\Sigma_{t \in T} p_{st}(n)$ goes to zero. Consequently, p_{ssf} goes to 1. But that means that the learner converges to the target state in the limit (with probability 1). Since this is true irrespective of the starting state of the learner, the learner converges to

the target with probability 1, and the associated target grammar g_f is Gold-learnable.

References

Chomsky, N. (1965). *Aspects of the theory of syntax*. Cambridge, MA: MIT Press.

Chomsky, N. (1981). *Lectures on government and binding*. Dordrecht: Foris.

Clark, R. & Roberts, I. (1993). A computational model of language learnability and language change. *Linguistic Inquiry, 24*(2), 299–345.

deMarcken, C. (1990). Parsing the LOB corpus. *Proceedings of the 25th Annual Meeting of the Association for Computational Linguistics* (pp. 243–251). Pittsburgh, PA: Association for Computational Linguistics.

Dresher, E. (1994). *Some current issues in learnability*. Unpublished paper. Talk given at MIT Linguistics Colloquium, Cambridge, MA, May 1994.

Dresher, E. & Kaye, J. (1990). A computational learning model for metrical phonology. *Cognition, 34*, 137–195.

Frank, R. & Kapur, S. (1992). *On the use of triggers in parameter setting*. Philadelphia, PA: Institute for Research in Cognitive Science, University of Pennsylvania, Report no. 92–52. In *Linguistic Inquiry*, in press.

Gibson, T. & Wexler, K. (1994). Triggers. *Linguistic Inquiry, 25*(4), 407–454.

Gold, E.M. (1967). Language identification in the limit. *Information and Control, 10*(4), 447–474.

Hale, K. & Keyser, J. (1993). On argument structure and the lexical expression of syntactic relations. In K. Hale and J. Keyser (Eds.), *The view from building 20* (pp. 53–110). Cambridge, MA: MIT Press.

Hamburger, H. & Wexler, K. (1977). A mathematical theory of learning transformational grammar. *Journal of Mathematical Psychology, 12*(1), 137–177.

Isaacson, D. & Madsen, R. (1976). *Markov chains theory and applications*. New York: Wiley.

Lightfoot, D. (1991). *How to set parameters*. Cambridge, MA: MIT Press.

MacWhinney, B. (1990). *CHILDES: A tool for studying language input*. Hillsdale, NJ: Erlbaum.

Niyogi, P. & Berwick, R.C. (1995). *The logical problem of language change*. AI-Memo 1516, Artificial Intelligence Laboratory, MIT.

Resnick, S. (1992). *Adventures in stochastic processes*. Boston: Birkhauser.

Wexler, K. & Culicover, P. (1980). *Formal principles of language acquisition*. Cambridge, MA: MIT Press.