ON TRIGGERED LEARNING

By

WILLIAM JOSEPH TURKEL

B.Sc., The University of British Columbia, 1990

Dipl. Appl. Ling., The University of British Columbia, 1991

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ARTS

in

THE FACULTY OF GRADUATE STUDIES

(Department of Linguistics)

We accept this thesis as conforming

to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

March 1997

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of LINGUISTICS

The University of British Columbia
Vancouver, Canada

Date 10 APRIL 1997

Abstract

In current linguistic theory, natural languages are thought to depend on extensive interaction between systems of principles or violable constraints. The principles or constraints are considered to be innate, and subject to a small degree of variation. This variation is expressed in *parameters* or, alternately, in the *ranking* of constraints. Under such accounts, the child acquires the language of his or her community, in part, by establishing the settings of parameters or the ordering of constraints. The forms which the learner uses in this task are known as *triggers*.

The study of triggered learning emphasises a number of dimensions of language acquisition. Here I consider three such dimensions: the space of possible languages, the nature of the input the child receives, and the nature of the learning algorithm. Chapter 1 provides an overview of the phenomena to be explained, the linguistic theory and the specific theory of learning which I assume. Chapter 2 investigates the extent to which small changes in the internal representation of grammars correspond to small changes in the resulting language. Chapter 3 explores the role that forms not accounted for by the target grammar may have on parameter setting. Chapter 4 shows that aspects of the learning scenario which lead to a failure of one kind of learning algorithm may not be as severe for another. In conclusion, I try to clarify the role that computational studies of language acquisition can play in the construction of linguistic theory, and the generation of testable hypotheses.

Table of Contents

List of Tables

List of Figures

Preface

This thesis is based on three papers which were previously written as separate entities. They have been substantially rewritten for inclusion here.

An earlier version of Chapter 4, was previously published under the same title in *Linguistic Inquiry* as Turkel (1996a). At the time of writing, earlier versions of Chapters 2 and 3 had been submitted to *Journal of Linguistics* and *Linguistic Inquiry*, respectively.

Acknowledgements

I thank my co-supervisors, Henry Davis and Doug Pulleyblank. Henry first showed me that learning lies at the core of cognition. To the extent that I have managed to be precise, concise and explicit, it is due to his inspiration. Doug taught me a way of thinking and a respect for the complexity of natural languages. I learned a lot during our collaboration, and look forward to future work together. Both have been exemplary teachers and good friends. The third member of my committee, Joe Pater, was pressed into duty at the last possible moment. He responded with a seemingly instant and easy command of the material and some perspicacious questions. I aspire to such cool.

Many of my ideas about learnability and acquisition were refined in discussions with Carl Alphonce, Dare Baldwin, Stefano Bertolo, Bob Berwick, Paul Bloom, Kevin Broihier, Robin Clark, Renee Desjardins, Elan Dresher, Jason Eisner, Bob Frank, Ted Gibson, Steven Gillis, Geoff Hall, Bill Idsardi, Partha Niyogi, Edwin Pulleyblank, Rushen Shi, Christine Stager, Bruce Tesar, James Thomas, William K. Thompson, Janet Werker, Ken Wexler, Chris Whincop and Andi Wu. Will Thompson deserves special mention, as I spent many enjoyable hours with him, drinking coffee and hashing over this stuff.

Many other people in the linguistic and psycholinguistic communities have provided friendship, feedback and encouragement. Although I cannot thank them individually, I am grateful to each.

Finally I thank my wife Juliet Armstrong, for love and support. This thesis (a small thing to be sure) is dedicated to her.

Chapter 1—Introduction

First Language Acquisition and Linguistic Theory

We begin with a phenomenon in the natural world: children somehow come to speak the language or languages spoken in their community. Given such a phenomenon, we may take a scientific stance toward it, and attempt to explain what is happening and why. This necessarily requires abstraction. We have to decide which details are relevant to our explanation, and which can be set aside. In order to make the abstraction, we have certain tools at our disposal, such as observation and experiment.

The scientific stance which I adopt here is that of Chomsky (e.g., 1965, 1981, 1986, 1988, 1995).

> A person who speaks a language has developed a certain system of knowledge, represented somehow in the mind, and ultimately, in the brain in some physical configuration. In pursuing an inquiry into these topics, then, we face a series of questions, among them:
>
> 1. What is the system of knowledge? What is in the mind/brain of the speaker of English or Spanish or Japanese?
> 2. How does this system of knowledge arise in the mind/brain?
> 3. How is this knowledge put to use in speech (or secondary systems such as writing)?
> 4. What are the physical mechanisms that serve as the material basis for this system of knowledge and for the use of this knowledge?
>
> (Chomsky 1988:3)

In attempting to provide an answer to Chomksy's second question, we have to presuppose an answer to the first—the nature of our developmental account will be largely shaped by our assumptions about the nature of linguistic knowledge.

Triggered Learning

The system of knowledge that underlies natural language is very rich and complex. One of Chomsky's standard examples of this complexity (1986:7-8) is the knowledge that native speakers possess about the possible referents of pronouns. For example, in sentence (1) below, *'them'* may not refer to *'the men'*. In the very similar sentence (2), it may.

(1) The men expected to see them.

(2) I wonder who the men expected to see them.

Chomsky uses many examples of this sort to argue that the system of knowledge is vastly underdetermined by the evidence available to children. This problem is known variously as *Plato's problem* or *the argument from the poverty of the stimulus* (Chomsky 1986, Wexler 1991).

Accounting for the richness of linguistic knowledge would not be as difficult if children received what is known as *negative evidence* (Gold 1967, Brown & Hanlon 1970, Marcus 1993, Bloom 1994). For example, if a child were told "*Them* cannot mean *the men* in the sentence *The men expected to see them*, but it can mean *the*

men in the sentence *I wonder who the men expected to see them*," then explaining first language acquisition would be a bit easier. In place of such metalinguistic information, the child might receive corrections when he or she made a mistake. It appears to be the case, however, that children do not have systematic access to correction, and do not pay that much attention to the correction they do get. So we still have something to explain.

One explanation is that language is structured by innate principles, and subject to a small degree of surface variation. The innate principles are believed to be uniform across the human species, and are known as *Universal Grammar.* The surface variation is limited to a finite number of finitely-valued dimensions known as *parameters.* The resulting theory is commonly known as Principles & Parameters (P&P) theory.

We think of a given language as having a setting for each of the parameters. Part of the task of the child acquiring a language is to determine the pattern of variation that characterises his or her language. This process is known as *parameter setting.* The linguistic forms which enable the child to determine the direction and kind of variation are known as *triggers.*

One particularly abstract way of thinking about parameters is in terms of a space of possible languages. Each of the parameters forms one of the dimensions of the space. Each language has a value for each dimension, and thus can be thought of as a point in the multidimensional *parameter space.* Those languages which have most

parameters set the same way will be closer to one another in the parameter space than those which differ in many parameter settings. Although this way of thinking about parameters is fairly abstract, it is also quite useful. I will refer to it often, and make extensive use of it in Chapter 2.

To summarise the discussion so far, the evidence available to the child acquiring a language seems to underdetermine the structure of the resulting knowledge. This suggests that much of the structure must be innate, and that variation is limited. What variation we do observe must be established by the learner on the basis of positive evidence from the ambient linguistic environment. I will refer to this process as *triggered learning.*

Although I concentrate here on Principles & Parameters theory, triggered learning is also of interest in other linguistic theories based on the idea of a Universal Grammar, such as Optimality Theory (OT) (Prince & Smolensky 1993, 1997). In OT, knowledge of language consists of innate, violable constraints in a particular ranking. The ranking determines grammatical forms in the language, in that satisfaction of more highly ranked constraints may involve violations of the constraints they dominate. Determination of the ranking in OT is analogous to determination of the settings of parameters in P&P theories. The similarities and differences of triggered learning in P&P and OT are explored in work by Pulleyblank & Turkel (1996, to appear a, to appear b).[1]

[1]It is also possible that triggered learning could play a role in theories where knowledge of

I make two further idealisations, the hypothesis of *continuity* (Pinker 1984, Bloom 1994) and the hypothesis of *stationarity* (Pinker 1981). Under the continuity hypothesis, the child begins with a setting for each parameter that is consistent with *some* natural language, although probably not the target language. Learning proceeds via a resetting of each parameter. Thus, each hypothesis entertained by the learner is a possible adult language, and learning can be viewed in terms of a trajectory through the parameter space. Under the stationarity hypothesis, there are no changes to the mechanism underlying the system, although there will be changes to the structures of knowledge the system supports.

The idea of stationarity has come under attack in recent work. Elman (1993) notes that it is a striking fact that the greatest learning occurs precisely at a point in time when the most dramatic maturational changes are also occuring. Furthermore, use-dependent addition of structure is a well-documented form of neural plasticity. It seems clear that such processes must be occuring during childhood, at least at the level of the brain. This may have important ramifications at the cognitive level, as non-stationary mechanisms have different learnability properties (Quartz 1993, Quartz & Sejnowski 1995). If stationarity turns out to be an untenable assumption, then continuity will probably also be untenable. Pending further work, I assume

---

language is not considered to be domain-specific. One can envision a theory where some aspect of language is subserved by a domain-general mechanism that also handles some other aspect of cognition. Upon exposure to the relevant triggering datum (which could be completely non-linguistic) the learner would reconfigure the mechanism, affecting performance in all domains handled by the mechanism. I set aside such possibilities here.

both idealisations here.

Computational Learning Scenario

We start with a phenomenon in the natural world, and by a process of abstraction, we attempt to explain what is happening, and why. Following Brent (1996), I will refer to this as a *what/why* theory.[2] Theories of linguistic structure are what/why theories.

In order to explain *how* something happens, we need to move to a further level of abstraction. A theory of parameter setting is a *how* theory... it suggests the way which parameters might be set. Note that there are interpretive problems with attempting to relate *how* theories directly to the phenomenal level. It is not clear what kinds of behavioural evidence would count for or against a particular algorithm for setting parameters. This is not to say that *how* theories are unrelated to phenomena, simply that their relation is mediated by *what/why* theories.

In this thesis, I will consider what I call *computational learning scenarios* (cf. Niyogi & Berwick 1996:162). Each such scenario consists of a parametric space, input to the learner and a learning algorithm. In general, I will not attempt to argue for the sufficiency or parsimony of the set of parameters. That is a job for *what/why* theorists. Rather, I assume standard sets of parameters that have been studied in the learnability literature. Similarly, the input which I assume is available

[2]Brent's work is based on the well-known distinction made by David Marr (1982) between computational and algorithmic levels of explanation.

to the learning algorithm is not meant to be representative of the input available to the child during language acquisition, but rather consists of a set of abstract forms generated by the grammars of interest. Finally, the learning algorithms are not meant to be models of the child, but rather to illustrate ways in which parameters might be set effectively.[3]

In the next two sections, I outline the sample parameter spaces which we will use.

V2-X-bar Parameter Space

The first sample parameter space is the *V2-X-bar* space of Gibson & Wexler (1994). The V2-X-bar space has three binary parameters: two control X-bar schemata, and one controls movement of a finite verb. The parameters and possible settings are given in Table 1.

Table 1: *V2-X-bar space: parameters and settings*

Parameter	*Effect [Possible Values]*
P1	Specifier-head direction [Spec-final/Spec-first]
P2	Complement-head direction [Comp-final/Comp-first]
P3	Verb-second [+V2/-V2]

There are eight possible grammars, each of which licenses a number of triggers.

[3]As an aid to the reader, I will use the pronoun 'it' to refer to abstract learners or learning algorithms, and 'he' or 'she' to refer to human children. This usage underscores the fact that abstract learning algorithms are not meant to be models of human children.

The grammars and triggers are shown in Table 2 (adapted from Gibson & Wexler 1994 Table 3 and Berwick & Niyogi 1996 Table 2). There are 72 distinct triggers, many of them licensed by more than one of the grammars.

In more recent work, the V2-X-bar space has been expanded to include four more parameters: a parameter that specifies complementizer direction, one for verb raising to Agr, one for verb raising to Tense, and one for embedded V2 (Bertolo, Broihier, Gibson & Wexler 1997b). I do not make use of the larger space here.

Quantity Insensitive Stress Parameter Space

The second sample parameter space which I use is taken from Dresher & Kaye (1990). Since I have only used part of the total parameter space, and since I have made minor changes, I describe the space in some detail.

The parameter space is meant to account for a portion of metrical phonology, and is a standard space for studies in phonological learnability (e.g., Dresher 1994, Gillis & Durieux 1996, Gillis, Durieux & Daelemans 1995, Gupta & Touretzky 1991, 1992). The full space has 11 parameters and generates 216 distinct stress systems. In order to have a smaller and more manageable space, I only consider the subspace which generates quantity insensitive (QI) stress systems. This space is still reasonably large (48 grammars), seems somewhat independent of quantity sensitive (QS) stress, and has the advantage that all forms generated by each of the languages in the space can be easily enumerated. Whether the results presented in

Table 2: *V2-X-bar space: languages and triggers*

Language	**P1**	**P2**	**P3**	*Triggers*
L1 (VOS-V2)	1	1	0	V S, V O S, V O1 O2 S, Aux V S, Aux V O S, Aux V O1 O2 S, Adv V S, Adv V O S, Adv V O1 O2 S, Adv Aux V S, Adv Aux V O S, Adv Aux V O1 O2 S
L2 (VOS+V2)	1	1	1	S V, S V O, O V S, S V O1 O2, O1 V O2 S, S Aux V, S Aux V O, O Aux V S, O2 V O1 S, S Aux V O1 O2, O1 Aux V O2 S, O2 Aux V O1 S, Adv V S, Adv V O S, Adv V O1 O2 S, Adv Aux V S, Adv Aux V O S, Adv Aux V O1 O2 S
L3 (OVS-V2)	1	0	0	V S, O V S, O2 O1 V S, V Aux S, O V Aux S, O2 O1 V Aux S, Adv V S, Adv O V S, Adv O2 O1 V S, Adv V Aux S, Adv O V Aux S, Adv O2 O1 V Aux S
L4 (OVS+V2)	1	0	1	S V, O V S, S V O, S V O2 O1, O1 V O2 S, O2 V O1 S, S Aux V, S Aux O V, O Aux V S, S Aux O2 O1 V, O1 Aux O2 V S, O2 Aux O1 V S, Adv V S, Adv V O S, Adv V O2 O1 S, Adv Aux V S, Adv Aux O V S, Adv Aux O2 O1 V S
L5 (SVO-V2)	0	1	0	S V, S V O, S V O1 O2 S Aux V, S Aux V O, S Aux V O1 O2, Adv S V, Adv S V O, Adv S V O1 O2, Adv S Aux V, Adv S Aux V O, Adv S Aux V O1 O2
L6 (SVO+V2)	0	1	1	S V, S V O, O V S, S V O1 O2, O1 V S O2, O2 V S O1, S Aux V, S Aux V O, O Aux S V, S Aux V O1 O2, O1 Aux S V O2, O2 Aux S V O1, Adv V S, Adv V S O, Adv V S O1 O2, Adv Aux S V, Adv Aux S V O, Adv Aux S V O1 O2
L7 (SOV-V2)	0	0	0	S V, S O V, S O2 O1 V, S V Aux, S O V Aux, S O2 O1 V Aux, Adv S V, Adv S O V, Adv S O2 O1 V, Adv S V Aux, Adv S O V Aux, Adv S O2 O1 V Aux
L8 (SOV+V2)	0	0	1	S V, S V O, O V S, S V O2 O1, O1 V S O2, O2 V S O1, S Aux V, S Aux O V, O Aux S V, S Aux O2 O1 V, O1 Aux S O2 V, O2 Aux S O1 V, Adv V S, Adv V S O, Adv V S O2 O1, Adv Aux S V, Adv Aux S O V, Adv Aux S O2 O1 V

this thesis scale up to the full Dresher & Kaye (1990) space, and whether they apply to other theories of metrical phonology, are important open questions.

In the Dresher & Kaye account, metrical structures are in the form of labelled trees built on rime projections. Since I am concerned here with quantity insensitive stress, I will not consider any articulation of the word tree below the level of the syllable. In any group of sister nodes in the tree, one is designated *strong*, and the rest *weak*. Stress patterns are controlled by these metrical structures, with the main stress of the word falling on the syllable dominated entirely by strong nodes.

There are six metrical parameters which can vary in QI stress systems.[4] These are shown in Table 3.

Table 3: *QI stress space: parameters and settings*

Parameter	*Effect [Possible Values]*
P1	The word tree is strong on the [Left/Right]
P3	Feet are built from the [Left/Right]
P4	Feet are strong on the [Left/Right]
P8A	There is an extrametrical syllable [No/Yes]
P8	It is extrametrical on the [Left/Right]
P10	Feet are iterative [No/Yes]

I assume fixed settings for the remaining metrical parameters: **P2** feet are [Bi-

[4]With the exception of **P10**, all parameters are taken directly from Dresher & Kaye (1990). I changed the sense of **P10** from noniterative to iterative. Elan Dresher (personal communication 7 June 1996) suggests that a better name for the parameter would have been 'Conflation'. Iterative [Yes] corresponds to Conflation [No] and vice versa. I work through an example of stress computation below.

nary], **P5** feet are Quantity Sensitive [No], **P6** feet are QS to the Rime/Nucleus [Don't care] and **P7** a strong branch of a foot must itself branch [Don't care]. Following Dresher & Kaye, I remove defooting (**P9**) from consideration.

Ordinarily, a system of 6 binary parameters would yield a space of 64 (i.e., 2^6) possible grammars. Parameters **P8A** and **P8**, have a built-in dependency, however, with **P8** being suspended unless **P8A** is set to the value [Yes]. If **P8A** is set to [No], then we don't care what the value of **P8** is. I will represent the don't care value with a diamond [$\Diamond$]. Given the dependency between **P8A** and **P8**, there are 48 possible grammars, each represented by a vector of settings for the six parameters.

The grammars are listed in Table 4. Parameter vectors for each grammar are represented compactly. For example, Grammar 1 has the vector $LLLN\Diamond N$, which is to be understood as: **P1** The word tree is strong on the [Left], **P3** Feet are built from the [Left], **P4** Feet are strong on the [Left], **P8A** There is an extrametrical syllable [No], **P8** It is extrametrical on the [Don't care] side, and **P10** Feet are iterative [No]. Along with each grammar, I have listed the grammars that are near to it in the parameter space (its *neighbours*). I clarify the neighbour relation, and make use of it, in Chapter 2.

Stress can be computed for a five syllable form in Grammar 1 as follows. We start with five syllables. If there were an extrametrical syllable, we would bracket it at this point. There are no extrametrical syllables, however, so we build binary feet from the left. Note that this leaves a unary foot on the right edge. Feet are

Table 4: *QI stress space: languages and neighboring grammars*

	System	Neighbours		System	Neighbours
1	$LLLN\Diamond N$	2,3,5,9,10,25	25	$RLLN\Diamond N$	1,26,27,29,33,34
2	$LLRN\Diamond N$	1,4,6,11,12,26	26	$RLRN\Diamond N$	2,25,28,30,35,36
3	$LRLN\Diamond N$	1,4,7,13,14,27	27	$RRLN\Diamond N$	3,25,28,31,37,38
4	$LRRN\Diamond N$	2,3,8,15,16,28	28	$RRRN\Diamond N$	4,26,27,32,39,40
5	$LLLN\Diamond Y$	1,6,7,17,18,29	29	$RLLN\Diamond Y$	5,25,30,31,41,42
6	$LLRN\Diamond Y$	2,5,8,19,20,30	30	$RLRN\Diamond Y$	6,26,29,32,43,44
7	$LRLN\Diamond Y$	3,5,8,21,22,31	31	$RRLN\Diamond Y$	7,27,29,32,45,46
8	$LRRN\Diamond Y$	4,6,7,23,24,32	32	$RRRN\Diamond Y$	8,28,30,31,47,48
9	$LLLYLN$	1,10,11,13,17,33	33	$RLLYLN$	9,25,34,35,37,41
10	$LLLYRN$	1,9,12,14,18,34	34	$RLLYRN$	10,25,33,36,38,42
11	$LLRYLN$	2,9,12,15,19,35	35	$RLRYLN$	11,26,33,36,39,43
12	$LLRYRN$	2,10,11,16,20,36	36	$RLRYRN$	12,26,34,35,40,44
13	$LRLYLN$	3,9,14,15,21,37	37	$RRLYLN$	13,27,33,38,39,45
14	$LRLYRN$	3,10,13,16,22,38	38	$RRLYRN$	14,27,34,37,40,46
15	$LRRYLN$	4,11,13,16,23,39	39	$RRRYLN$	15,28,35,37,40,47
16	$LRRYRN$	4,12,14,15,24,40	40	$RRRYRN$	16,28,36,38,39,48
17	$LLLYLY$	5,9,18,19,21,41	41	$RLLYLY$	17,29,33,42,43,45
18	$LLLYRY$	5,10,17,20,22,42	42	$RLLYRY$	18,29,34,41,44,46
19	$LLRYLY$	6,11,17,20,23,43	43	$RLRYLY$	19,30,35,41,44,47
20	$LLRYRY$	6,12,18,19,24,44	44	$RLRYRY$	20,30,36,42,43,48
21	$LRLYLY$	7,13,17,22,23,45	45	$RRLYLY$	21,31,37,41,46,47
22	$LRLYRY$	7,14,18,21,24,46	46	$RRLYRY$	22,31,38,42,45,48
23	$LRRYLY$	8,15,19,21,24,47	47	$RRRYLY$	23,32,39,43,45,48
24	$LRRYRY$	8,16,20,22,23,48	48	$RRRYRY$	24,32,40,44,46,47

left-headed, so we mark this in, as well as making the unary foot strong. We build a left-headed word tree. Finally, since Grammar 1 is noniterative, we remove all secondary stresses by changing any strong node dominated by a weak node to weak. The computation is shown in Figure 1.

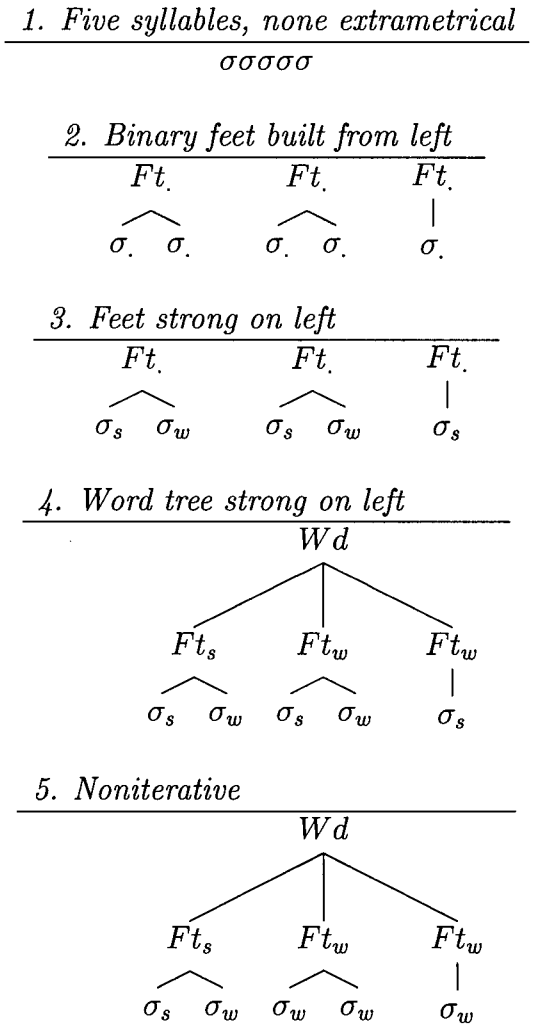
There are 114 distinct triggers generated by the grammars in this parametric space. Comparison with the V2-X-bar space shows that there are six times as many grammars (48 vs. 8) but less than twice as many triggers (114 vs. 72). This may be indicative of an interesting difference between syntactic and phonological parameter sets, or it may be accidental. I don't have any evidence one way or the other, so I merely note the phenomenon.

Input to the Learner

The input to the learner consists of sequences of triggers that are licensed by the target grammar. The linguistic nature of these triggers is not important for our purposes, so we use integers to denote them. For example, in the V2-X-bar space, a sentence consisting of a verb followed by a subject (V S) is the first trigger, and is licensed by Languages 1 and 3. For us, it is sufficient to say that the learner receives trigger 1.[5]

[5]Berwick & Niyogi (1996:609) point out that this notion of triggers is purely extensional, in the sense that the triggers do not bear any logical relation to the grammars themselves. Such a relationship may be stipulated. For example, Dresher & Kaye (1990:157) postulate an *appropriateness condition* which states that cues must be appropriate to their parameters with regard to their scope and operation. I set aside conditions of appropriateness here, as they are orthogonal to our concerns.

Figure 1: *Example of computation of stress in grammar 1*



In the next two sections, I outline a pair of parameter setting algorithms. The algorithm specifies the way in which the learner makes use of the triggers to determine its trajectory through a parameter space to the target grammar.

Triggering Learning Algorithm

The first learning algorithm that I will make use of is the Triggering Learning Algorithm (TLA) of Gibson & Wexler (1994). The TLA is specified as follows.

> Given an initial set of values for n binary-valued parameters, the learner attempts to syntactically analyze an incoming sentence S. If S can be successfully analyzed, then the learner's hypothesis regarding the target grammar is left unchanged. If, however, the learner cannot analyze S, then the learner uniformly selects a parameter P (with probability $1/n$ for each parameter), changes the value associated with P and tries to reprocess S using the new parameter value. If analysis is now possible, then the parameter value change is adopted. Otherwise, the original parameter value is retained.

The TLA has a number of important properties, some of which we explore here. It is *error driven*—it will not make a change to its current hypothesis, unless it is unable to account for some input. It is *memoryless*—since it does not store former hypotheses or data, it needs only to account for the current datum. This means that it is extremely susceptible to forms that are in the input but which are not generated by the target grammar. It is *greedy*—it will not adopt a change that does not lead to an immediate improvement. It is *conservative*—it only makes small adjustments to its current hypothesis. In terms of the parameter space, this means

that the learner's trajectory is characterised by small steps between neighbouring grammars, rather than large jumps from one region of the space to another.

When the learning algorithm stops adjusting its hypothesis, we say that it has *converged.* Gibson & Wexler (1994) used the TLA to demonstrate that a small, plausible parameter space contained a number of *local maxima*, non-target grammars that were better than any nearby grammar in the space. Given the properties of the TLA, these grammars prevented the learner from converging to the target grammar.

There are a number of modifications to the TLA which address the problem of local maxima. One that Gibson & Wexler proposed was that the learner might start in some distinguished initial grammar, from which convergence to all possible targets was possible in principle. Work by Broihier (1995) and Pulleyblank & Turkel (to appear a, to appear b) suggests that such initial states are not a property of all P&P and OT spaces, however. Another possibility is that the learner may be constrained by maturation, and thus not able to entertain problematic grammars. Bertolo (1995) explores the putative role of maturation in the TLA in some depth. Other possibilities include adding a small amount of memory for data or previous hypotheses, relaxing greediness to allow the learner to take a neutral step on occasion, relaxing conservativism so the learner can jump out of local maxima, and so on. I will not discuss such modified algorithms here (cf. Frank & Kapur 1996, Berwick & Niyogi 1996, and others).

Work in the TLA framework is greatly aided by an exact model of the (unmod-

ified) TLA as a Markov chain (Niyogi & Berwick 1993, 1995a, 1995b, Berwick & Niyogi 1996). The Markov chain model (which I use in Chapter 3) allows one to predict convergence times for the TLA. When embedded in a grammatical dynamical system, it also allows one to make predictions about the diachronic evolution of a population of speakers, each modelled by a TLA (e.g., Niyogi & Berwick 1995a, 1995b, Pulleyblank & Turkel 1996a, to appear c).

Genetic Algorithm

The other learning algorithm that I consider is a powerful nonlinear optimisation algorithm called the *genetic algorithm.* To motivate such an algorithm, consider the following problem. We are trying to tune a special television set which has four dials, each of which has 4 settings. There are $4^4 = 256$ possible different settings that the TV set could have. Unfortunately, only one of those provides us with nice, clear reception of our favourite channel. The problem is to find the correct setting. Let us consider two television sets. On the first, the action of each of the dials is independent in the following sense. When we switch the first dial around, it adjusts the colour. The first setting gives us bright red people with a yellow sky. The second setting gives us very sickly looking people and a greenish-blue sky. The third gives us a blue sky and normal-coloured people. We are happy with that, and we move to the second dial, and begin trying settings until the sound comes out correctly. An algorithm for finding the correct setting of this TV set could adjust each dial until

the correct setting for that dial was established and then move on to the next. This is a problem in linear optimisation. Say that on our second TV set, the dials are dependent on one another. We adjust the first dial so that the people are the right colour, but when we start adjusting the second dial (to adjust the sound) the people start changing colours. The only way to find the right setting is to simultaneously adjust all four dials at once. This is a more difficult problem, a problem in nonlinear optimisation.

Instead of adjusting the dials separately, a genetic algorithm solution to the problem would try a setting of all four of the dials at once, then move to another setting. So the genetic algorithm evaluates a whole solution to the problem, rather than attempting to solve it in a piecewise fashion.

In brief, a genetic algorithm is a form of evolution that is typically implemented on a computer. A number of randomly generated solutions (dial settings, in our TV example) are each evaluated. In general, none of them is going to be particularly good. For the genetic algorithm to work, however, some of those solutions must be measurably better than others. To go back to our TV example, it might be the case that the colour setting next to the correct one is fairly close: the people are sickly looking but not bright red, and the sky is greenish-blue rather than yellow. If it is the case that there is some way of telling when answers are getting better, then the genetic algorithm has something to go on.

We take the best of our randomly generated answers, and use those to create a

set of new answers. These new answers are similar to the previous ones, but not exactly the same. The analogy is to reproduction: babies resemble their parents, but are not clones.[6] If the answers that were better were used preferentially to generate the next set of answers, then eventually, the algorithm should find the right answer. The operation of the genetic algorithm is described below (taken from Turkel 1996b).

1. *Individuals encode possible solutions to a problem.* The genetic algorithm operates on a population of individuals, where each individual is a possible solution to a problem of interest.
2. *Individuals can be mated.* Two individuals can be combined in such a way as to result in a third, which is similar to the parents but not identical. In addition, individuals are sometimes changed slightly at random, a process known as *mutation.*
3. *Some individuals are better than others.* The *fitness function* of the genetic algorithm determines how good the solutions represented by each individual are.
4. *The better individuals are preferentially selected to mate.* Individuals which represent better solutions than the others in a given population are more likely to be parents. The basic idea is *survival of the fittest.*
5. *The population adapts to its environment.* Over time, the average fitness of members of the population increases; eventually, the solutions represented by the individuals are good enough. At that point, we say that the algorithm has converged.

Summary and Overview

To recap, we consider computational learning scenarios with the following components.

[6] Around the time of writing (March 1997) sheep and monkeys were cloned. I am talking about natural reproduction here.

1. A parametric space

2. Input to the learner

3. A learning algorithm

I investigate the property of *smoothness* of parametric spaces in Chapter 2. Following Chomsky (1986), we can distinguish between *I-language*, the knowledge that the speaker has of his or her language, and *E-language*, the set of strings generated by the I-language. A smooth relation between I-language and E-language would entail that small changes in I-language were associated with small changes in E-language. Such a relation is an empirical question, and will have implications for parametric theories of language acquisition, variation and change. I show that there is a smooth relation between I-language and E-language for a sample parametric space. Since the V2-X-bar space is very small—3 parameters, $2^3 = 8$ grammars—I use the QI stress space.

In Chapter 3, I consider the effect that noise in the input has on the behaviour of the TLA. Recall that the TLA is memoryless, and thus susceptible to input forms that are not generated by the target language, but that *are* generated by some other language in the space. Using the Markov chain model, I show that (contrary, perhaps, to expectation) a small amount of noise actually improves the convergence of the TLA. I relate this finding to a phenomenon known as *stochastic resonance.* This chapter uses the V2-X-bar space so that the results can be compared with the

performance of the TLA given in Gibson & Wexler 1994.

In Chapter 4, I study the performance of a genetic algorithm-based learner. I show that, unlike the TLA, the convergence of the genetic algorithm is not disrupted by the presence of local maxima. To facilitate comparison of the genetic algorithm and the TLA, I use the V2-X-bar space.

Chapter 2—Smoothness in a Parametric Subspace

Introduction

We begin our study of triggered learning by considering the first part of the learning scenario, the parametric space. Characteristics of the space will have an effect on the trajectory that the learner takes, from the initial grammar to (hopefully) the target grammar. At each stage of parameter setting, the learner is entertaining a particular grammar as its current hypothesis. Each of these grammars generates a certain set of surface strings. The approach that I take here is to determine the degree to which a change in hypothesised grammar affects the set of surface strings currently generated. To make this notion precise, I make use of the distinction between I-language and E-language, and the notion of smoothness.

In Principles & Parameters theory, the object of study is what Chomsky (1986:22) referred to as *I-language*, "some element of the mind of the person who knows the language, acquired by the learner, and used by the speaker-hearer." The process of language acquisition is construed in part as the process of fixing language-specific values for a finite set of finitely varying parameters. Given such an emphasis, the nature of the relation between parameter settings and the resulting surface strings (the *E-language*) is not clear. It may be the case that the relation between I-language and E-language is *smooth* (Niyogi & Berwick 1993), and that small changes in parameter settings correspond to small changes in the resulting sets of surface strings.

Alternately, parameters may be interdependent to such an extent that a change in a single parameter has ramifactions throughout the grammar.

Smoothness (or the lack thereof) has implications for theories of language acquisition, variation and change. For example, Clark (1989) proposed the Single Value Constraint, a requirement that successive hypotheses of a learner differ in the value of at most one parameter. This constraint was adopted by Gibson & Wexler (1994) for the Triggering Learning Algorithm (TLA), since there is no evidence that children make large-scale changes to their grammars in a single step.[7] If the relation between parameter settings and surface strings is nonsmooth, however, then making the learning algorithm conservative may result in a series of wildly fluctuating E-languages anyway, an undesirable result.

Language variation and language change can also be cast in terms of parameter settings (Lightfoot 1991, Clark & Roberts 1993, Niyogi & Berwick 1995a, 1995b). Because of differences in ambient language input and other factors, different speakers of the 'same' language may have slightly different settings for some parameters. If the I-language/E-language relation is nonsmooth, then this account ceases to convince—it becomes a recipe for Babel, and not for idiolectal or dialectal variation. Likewise, if diachronically successive I-languages differ in the setting of a single parameter, then we expect the resulting E-languages to be similar, rather than

[7]Both Paul Bloom and Doug Pulleyblank have separately pointed out to me the fact that parameter setting must be mostly complete by a fairly early age, so it is not clear there is much empirical evidence pertaining to the question one way or another.

apparently unrelated.

In this chapter, I provide two smoothness measures for the relation between Quantity Insensitive (QI) stress systems, and the stress patterns generated by those systems. The QI stress systems are each characterized by settings for six metrical parameters, and comprise a subspace of the parameter space for metrical phonology (Dresher & Kaye 1990). I show that QI stress is smooth in the sense that every I-language in the subspace has an E-language which is more similar on average to the E-languages of its neighbours than to the E-languages of its non-neighbours.

The Problem of Abrupt Change

Principles and Parameters theories are highly deductive; a change in the setting of a single parameter may have far-reaching consequences (Chomsky 1981). As an example, consider the location of main stress in a word. In the parametric system used here, there is a single parameter which sets either the leftmost or rightmost node of an unbounded word-tree to *strong*. In such a system, the setting of this parameter will affect the stress of almost every form in the language, resulting in either initial or final stress (cf. Dresher & Kaye 1990:155 on the difficulty of measuring closeness of fit).

To a large extent, the power and usefulness of parameters depends on their widespread influence. Formulating a parameter such as the one above amounts to hypothesizing that the learner is able to determine that the target language has

initial (or final stress), and adjust his or her behaviour accordingly. Simultaneously, we hypothesize that two languages might differ only in that respect, whether they are diachronic stages of one language or synchronic neighbours.

If the setting of a single parameter can have such an influence on the resulting E-language, we are justified in asking whether there is any point in making learning algorithms conservative. For example, it might be the case that the E-languages of any pair of I-languages are dissimilar to roughly the same degree, that every change in I-language, no matter how large or small, is accompanied by a uniform change in E-language. Alternately, the degree of the change in I-language might not be correlated with the degree of the change in E-language. In either case, conservativism in the learning algorithm would be pointless.

Ideally, a measure of the smoothness (or lack of smoothness) for a sample parametric space can be used to make predictions about the kinds of changes that language learners will go through during acquisition. These predictions can be used to both suggest empirical research, and to refine the design of learning algorithms.

Measuring Smoothness

To study smoothness, I use the Quantity Insensitive stress space described in Chapter 1. Recall that the space has six parameters, 48 possible grammars and 114 distinct triggers.

In order to measure smoothness, we need to quantify the degree to which a pair

of I-languages differ, and also the degree to which the corresponding E-languages differ. We start with the difference between a pair of I-languages, denoted d_I.

> Given a pair of I-languages I, J the difference between them $d_I(I, J)$ is equal to the number of parameter values for which corresponding parameters in I and J are *distinct*. The don't care value [$\Diamond$] is considered to be *nondistinct* from any other value for the same parameter.

As an example, let I be the parameter vector for Grammar 1, $I = LLLN\Diamond N$. I is *nondistinct* from two other parameter vectors $I\prime = LLLNLN$ and $I\prime\prime = LLLNRN$. That is, $d_I(I, I\prime) = 0$ and $d_I(I, I\prime\prime) = 0$. Note that the nondistinct relation is not transitive: I and $I\prime$ are nondistinct, as are I and $I\prime\prime$, but $I\prime$ and $I\prime\prime$ are distinct from one another. The grammars $I\prime$ and $I\prime\prime$ are not included in the 48 considered in this study.

There are six parameter vectors at a distance (d_I) of one from I, known as the *neighbours* of I. They are $LLRN\Diamond N$ (Grammar 2), $LRLN\Diamond N$ (Grammar 3), $LLLN\Diamond Y$ (Grammar 5), $LLLYLN$ (Grammar 9), $LLLYRN$ (Grammar 10), and $RLLN\Diamond N$ (Grammar 25). The neighbours for each of the grammars in the QI subspace are listed in Chapter 1.

Any pair of I-languages that have a d_I greater than one are said to be *non-neighbours*. Each language in the QI subspace has 41 non-neighbours. One non-neighbour of I is $LRRN\Diamond N$ (Grammar 4), since $d_I(LLLN\Diamond N, LRRN\Diamond N) = 2$.

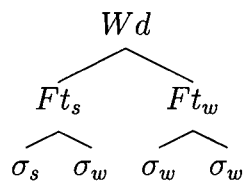
Each I-language generates a pattern of stresses in the corresponding E-language. Here I consider forms that have between two and nine syllables. As Dresher & Kaye

(1990:185) note, all stress systems converge at monosyllabic forms, and thus these forms will not convey any information to the learner about the stress patterns of the language. I assume that forms of ten or more syllables will be relatively infrequent in speech to language learners.[8] I also abstract away from possible complications due to different morpheme classes and so on. That is, I follow Dresher & Kaye's (1990:153) *transparency assumption*: words with identical syllable structure are accented in the same way.

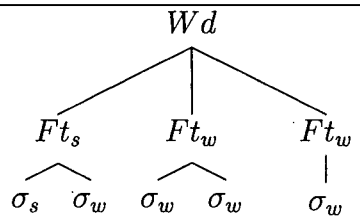
The parameter vector for a given I-language can be used to determine the stress pattern for each of the forms. For example, Grammar 1 has a left-headed word tree and left-headed binary feet built from the left. There is no extrametrical syllable and no secondary stress. The net result is that all forms have initial stress. The metrical trees for forms with four and five syllables will look as follows.

Grammar 1

Four syllables



Five syllables

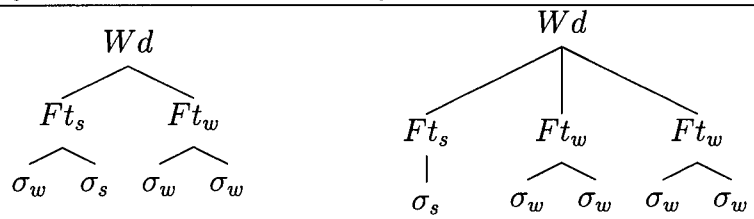


Grammar 4 has a left-headed word tree, but has right-headed binary feet built from the right. Again, there is no extrametrical syllable and no secondary stress. In Grammar 4, metrical trees for forms with four and five syllables will look as follows.

[8]There are two possible two-syllable forms; each is a trigger in twenty-four of the languages. The mean number of grammars which generate a given trigger are 5.33 for three-syllable forms, 3.43 for four-syllable forms, 2.82 for five-syllable, and 2.67 for six-syllable and longer forms. A smoothness measure based exclusively on forms less than six syllables would be artificially smooth.

Grammar 4

Four syllables *Five syllables*



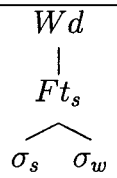
A plausible learner will not have access to metrical trees, however, but to phonetic stress patterns. The information which is available in the input and which is relevant to learning stress consists of syllables marked as bearing main stress (2), secondary stress (1) or no stress (0).

To determine the stress pattern from the metrical tree, we assign main stress to the syllable dominated entirely by strong nodes, secondary stress (if applicable) to any other strong syllables, and no stress to remaining syllables. Grammar 1 generates the stress pattern 2000 for words of four syllables, and 20000 for words of five syllables. Grammar 4 generates the stress patterns 0200 and 20000 for words of four and five syllables, respectively. The important thing to note is that surface stress patterns underdetermine metrical trees. A learner confronted with the stress pattern 20000 does not know if it should be parsed as in Grammar 1, or Grammar 4, or in some other way (the stress pattern 20000 is in fact generated by Grammars 1, 3, 4, 10 and 14). This is an example of a form which is *weakly equivalent* in two or more grammars; it has the same phonetic form but different abstract representations.

It is also possible for a given stress pattern to have the same metrical tree in two different languages. For example, the stress pattern 20 has the same metrical

tree in Grammar 1, Grammar 3, and many others in this space. In this case, the forms (although not the grammars) are said to be *strongly equivalent.* The learning algorithm will only be able to tell if two forms are *equivalent*, and will not be able to distinguish between weak and strong equivalence.

Metrical tree for 20 *in Grammars 1,3,...*



The phonetic stress patterns generated by a given I-language comprise the E-language. Since we are considering forms with two to nine syllables only, there are eight such forms in each E-language. Potentially, there could be as many as 384 different phonetic stress patterns (i.e., $8*48$) generated by all of the languages in the subspace. Strong and weak equivalence reduces the number of distinct forms to 114. Some of these are shared by many of the grammars, and some are unique to a single grammar. For example, the stress pattern 20 is common to half of the grammars in the space. The stress pattern 2101010 can only be generated by Grammar 7. I will refer to the phonetic stress patterns of a particular grammar as the *triggers* for that grammar, in keeping with work such as Gibson & Wexler 1994.

Given E-languages for each of the I-languages in our space, we can now calculate the difference of a pair of E-languages (denoted d_E) by considering the forms in each. Assume for a moment that we have a d_E measure. I define *smoothness* as follows.

Let I be an I-language and IE its corresponding E-Language. I has n neighbours $N_1, \ldots, N_n$ (with corresponding E-languages $NE_1, \ldots, NE_n$), and x non-neighbours $X_1, \ldots, X_x$ (with corresponding E-languages $XE_1, \ldots, XE_x$). The region of parametric space surrounding I is *smooth* iff

$$\frac{1}{n}\sum_{i=1}^{n} d_E(IE, NE_i) < \frac{1}{x}\sum_{i=1}^{x} d_E(IE, XE_i)$$

There are many different ways that we could measure d_E, the difference between E-languages. I will concentrate on methods that involve comparing each form in one E-language with one or more forms in the other, and then averaging the results to get an overall difference measure.

The simplest such comparison involves equivalence of forms. We start with a function that compares two forms and returns a one if they are not equivalent and a zero if they are. Since two forms of unequal length cannot be equivalent, we need only apply this function to the two syllable forms generated by each language, the three syllable forms, and so on.

Let $Diff(a, b)$ be a function which returns 1 if a and b are non-equivalent and 0 if they are equivalent. Given two E-languages A (with forms $a_2, a_3, \ldots, a_k$) and B (with forms $b_2, b_3, \ldots, b_k$), the difference between them is

$$d_E(A, B) = \frac{1}{k-1}\sum_{i=2}^{k} Diff(a_i, b_i)$$

I will refer to this measure as the *Mean Difference in Triggers* (MDT) and express it as a percentage for convenience. Given this measure, we can now calculate the MDT for every pair of grammars in the space. Comparing the MDT for the

neighbours of a given grammar with the MDT for the non-neighbours will allow us to determine whether the parametric space is smooth in the region of a given grammar. MDT measures are given in Table 5. *By the MDT measure, the QI subspace is smooth everywhere.*

Many other measures of E-language difference (d_E) are possible. Since the stress patterns consist of a sequence of stresses, determining the extent to which a pair of forms differs is a problem of *sequence comparison* (Kruskal 1983). Typical sequence comparison methods measure the difference of a pair of sequences in terms of the number of substitutions required to convert one to the other, or in terms of deletions and insertions, swaps or other *local* operations.

As a second comparison measure, I computed the distance between main stress positions for two forms of a given length, and incremented the distance for every remaining position that differed in secondary stress. I refer to this measure as *Gradient Difference in Stress* (GDS).

> Given two E-languages A (with forms $a_2, a_3, \ldots, a_k$) and B (with forms $b_2, b_3, \ldots, b_k$), compute the $GDS(A, B)$ as follows. Let *mainloc* be a function which returns the syllable position of main stress. For example, $mainloc(00201011) = 3$.
>
> $$MainDiff(a_i, b_i) = |mainloc(a_i) - mainloc(b_i)|$$
>
> $SecDiff$ is a function which returns a 1 for each syllable position where a pair of forms differ and neither has main stress.

$$d_E(A, B) = \frac{1}{k-1} \sum_{i=2}^{k} MainDiff(a_i, b_i) + SecDiff(a_i, b_i)$$

Table 5: *MDT and GDS measures*

System	MDT Neighbours	MDT Non-neighbours	GDS Neighbours	GDS Non-neighbours
1	61	86	17	41
2	69	80	17	34
3	53	87	15	41
4	69	83	19	37
5	78	90	37	53
6	83	91	33	47
7	83	91	35	53
8	78	90	38	50
9	69	80	15	34
10	53	87	13	42
11	72	87	15	33
12	61	82	14	34
13	64	81	14	34
14	47	88	12	42
15	61	86	15	33
16	69	82	15	37
17	75	83	37	42
18	67	88	28	50
19	72	90	30	42
20	75	85	27	42
21	75	85	29	42
22	72	90	34	49
23	67	88	29	43
24	75	83	37	45

Continued

Table 5 continued: *MDT and GDS measures*

System	MDT Neighbours	MDT Non-neighbours	GDS Neighbours	GDS Non-neighbours
25	69	83	19	37
26	53	87	15	41
27	69	80	17	34
28	61	86	17	41
29	78	90	38	50
30	83	91	35	53
31	83	91	33	47
32	78	90	37	53
33	69	82	15	37
34	61	86	15	33
35	47	88	12	42
36	64	81	14	34
37	61	82	14	34
38	72	87	15	33
39	53	87	13	42
40	69	80	15	34
41	75	83	37	45
42	67	88	29	43
43	72	90	34	49
44	75	85	29	42
45	75	85	27	42
46	72	90	30	42
47	67	88	28	50
48	75	83	37	42
Mean	69	86	24	42

Example. As an example, we compute the difference between the two forms 02101010 and 10101020. We start with the difference in the positions of the main stresses.

$$MainDiff(02101010, 10101020) = |2 - 7| = 5$$

The two forms differ in secondary stress only in the first position, where the first has a zero and the second a one. Positions 3,4,5,6 and 8 share the same value for secondary stress, and positions 2 and 7 are not included because one of the forms has main stress in that position. Thus, the difference between 02101010 and 10101020 is $5 + 1 = 6$.

The new d_E (GDS) uses a different function to compare forms, but is otherwise calculated the same as the MDT measure above. GDS measures (also expressed as percentages) are given in Table 5. *By the GDS measure, the QI subspace is smooth everywhere.*

Note that smoothness as I have defined it is an average measure. On an individual basis, a pair of neighbouring grammars may have no triggers in common, giving rise to the kinds of abrupt changes studied by Lightfoot (1991) and others. For example, Grammar 1 generates the triggers 20, 200, 2000, 20000, and so on. The neighbouring Grammar 25 (which differs only in the setting of the parameter for the headedness of the word tree) generates triggers 20, 002, 0020, 00002, and so on. The E-languages of Grammars 1 and 25 have only one equivalent trigger, 20, and are very different by the GDS measure as well. On the other hand, Grammar 14 is not a neighbour of Grammar 1, and yet generates exactly the same triggers as Grammar 1. This is because Grammar 14 has initial stress, but also has feet built from the right and right-edge extrametricality. Since there are no secondary stresses in either

Grammar 1 or Grammar 14, the fact that each assigns very different metrical trees to the surface forms is not apparent (cf. the *bogus parameters* of Frank & Kapur 1996).

Conclusion

In this chapter, I have shown that the relationship between parameter settings for QI stress systems and the resulting stress patterns is smooth. Every grammar in the subspace has an E-language which is more similar on average to the E-languages of neighbouring grammars than to the E-languages of non-neighbouring grammars.

It is important to note that this result is specific to this particular subspace, and is not a general result for an arbitrary parameter space. The smoothness measure can be used to compare two sets of parameters that explain the same phenomena—for example, to compare the Dresher & Kaye account with a different version of QI stress. The measure may also be used to assess the effects of adding more parameters to the system. Does the addition of parameters for quantity sensitive stress result in a smooth space? The measure may even be used to compare a parameter-based account of stress with an Optimality Theoretic account, if we take the difference in neighbouring I-languages in Optimality Theory to be some local permutation of the constraint hierarchy, such as swapping adjacent constraints. I leave these questions for future work.

For QI stress systems then, parameter-setting learning algorithms should be con-

servative, variation can be studied in terms of nearness of I-languages in parameter space, and language change can be framed in terms of change in parameter settings. The fact that smoothness is an average measure means that a change in a single parameter value may be associated with abrupt change in E-language, with more gradual change, or with no change at all.

Chapter 3—Noise-induced enhancement of Parameter Setting

Noise and the Triggering Learning Algorithm

In this chapter, we consider the second component of the learning scenario, the input to the learner from the ambient linguistic environment. Simple learning algorithms that rely on triggers are particularly sensitive to the distribution of such triggers in the input data. For example, Niyogi & Berwick (1993:7) show that by altering the distribution of triggers available to a learner, convergence time can be pushed up to as much as 50 million samples. Another possible problem is the presence of forms in the input which are not generated by the target, but are generated by some non-target grammar. Such forms will act as triggers for the non-target grammars. Here I assess the impact that such forms have on the convergence of the Triggering Learning Algorithm.

Recall that the TLA is a simple, memoryless learner that Gibson & Wexler (1994) used to demonstrate that parameter setting is non-trivial, even in very small spaces. Using the familiar learnability paradigm of identification in the limit (Gold 1967), they showed that the TLA can become entrapped in local maxima, preventing convergence to the target grammar.

In analysing the performance of the TLA, Gibson & Wexler abstracted away from the problem forms in the input which are not generated by the target grammar. Such forms constitute a kind of noise. The TLA is constructed such that it will

effectively ignore forms that are not generated by any grammar in the parameter space. Although such forms are noise, they do not have any deleterious effects on parameter setting, and we can safely set them aside. Of more concern, as pointed out by Frank & Kapur (1996) and Niyogi & Berwick (1993), are forms that *are* generated by some non-target grammar. Such forms can arise in a number of ways. They may be peripheral constructions (Fodor 1989), dialectal variations or performance errors.

It is possible that noise in the input to the TLA will alter the problem of local maxima. In the limit, noisy input means that there will always be some probability of escaping from a local maximum (Niyogi & Berwick 1993), and this may be reflected in the performance of the learner when given a small number of data (the *finite sample case*). It is not clear in advance what the effect of noise will be, however. It may be that noisy data tend to deflect the learner into local maxima as easily as they deflect it out.

Stochastic Resonance

A phenomenon known as *stochastic resonance* provides a suggestive analogy. In a variety of physical and biological systems, noise has been shown to enhance the detection of weak signals (reviewed in Wiesenfeld & Moss 1995). The hallmark of stochastic resonance is that a detector is optimally sensitive at some non-zero level of noise. This means that if we plot the performance of the detector against the level of noise, we should see that performance increases as noise is added, up to

some point, then decreases after that.

An example may serve to convey the essence of a stochastic resonance account. Wiesenfeld & Moss describe a series of experiments with the crayfish *Procambarus clarkii.* The crayfish normally lives in an environment in which it is buffeted by water currents and eaten by fish. In order to solve the problem of avoiding predators, the crayfish needs to be able to detect the weak, coherent water motions of swimming predators in an environment of strong, random water motion. To this end, it has a system of mechanoreceptor hair cells which detect the coherent motion in a noisy environment. The weak, periodic stimulus entrains the random fluctuation of the environment, greatly enhancing the periodic component. Because the sensory system of the crayfish evolved in noisy environments, it is optimally sensitive at non-zero levels of noise.

Obviously, the parameter setting case differs from the example in a number of ways. The key point however, is that noise may enhance the learning algorithm's ability to detect the target language. In the following, I show that this is indeed the case for the TLA under some conditions: the maximum probability of convergence is found at non-zero levels of noise.

Noise-induced Enhancement of Parameter Setting

We can study the effects of noise on parameter setting by using the Markov chain model of the TLA (Niyogi & Berwick 1993). Given a distribution of triggers,

the Markov chain model allows us to derive a transition matrix for the TLA in the parametric space under consideration. The transition matrix gives the probability that the TLA moves from any grammar to any other in one step. Raising the matrix to a power m gives the probabilities that the TLA moves from any grammar to any other when given m data (the finite sample case).

The transition matrix is computed from the distribution of triggers that are given to the learner during acquisition. In order to study the effects of noise, we need only give our learner a certain number of forms that are triggers for grammars other than the target grammar. Note that the learner may also receive forms that are not generated by any possible grammar; since these do not cause movement in the parametric space, I do not consider them here.

I used the Markov chain model to predict performance of the TLA in the parametric space described in Gibson & Wexler 1994 (the V2-X-bar space). Rather than assessing behaviour in the limit, I chose to assess the probability of convergence after a finite sample of 300 forms. This number was chosen in accordance with two desiderata. First, it is low enough to be plausible. Second, adding more data does not result in any noticeable change in the convergence probabilities.

There are eight grammars in the V2-X-bar space, which I will designate Language 1, Language 2, etc. There are a total of 72 distinct triggers generated by all of the languages in the space. Assume that the child is learning Language 1 in a monolingual community of Language 1 speakers. The child has a 100% chance of

receiving forms generated by Language 1. This corresponds to a learning scenario with zero noise. Since Language 1 licenses 12 triggers, the child receives each form as input 1/12th of the time. Given 300 data generated by Language 1, the chance that the learner will converge to the target grammar is very close to 1. This is true even if the learner starts with all three parameters set incorrectly (i.e., starts in Language 8 in the V2-X-bar space). I will refer to the case where the learner starts with all parameters set incorrectly as the *worst initial grammar.* For Language 1 it is Language 8, for Language 2 it is Language 7, and so on. Assuming that there is no noise, that the learner always starts in the worst initial grammar, and that the learner only receives 300 data, Languages 1, 2, 4, 6 and 8 are learnable with a probability very close to 1. Languages 3, 5 and 7, on the other hand, are unlearnable under the same conditions. The reader is encouraged to consult Gibson & Wexler 1994, Niyogi & Berwick 1993, and Berwick & Niyogi 1996 for details.

We are interested in cases where the input to the learner is noisy. Consider a case where exactly 1% of the adult speakers in the child's ambient environment are not speaking Language 1. Then the child will have a 99% chance of hearing a form generated by Language 1. I assume that the speakers of languages other than Language 1 are uniformly distributed. In other words, the child has a 1/7th of 1% chance of hearing a form from Language 2, and the same chance of hearing a form from Languages 3 to 8. The total chance of hearing a form that is not generated by

the target language is 1%, hence I will refer to this scenario as a case of 1% noise.[9] I altered the Markov chain model to reflect increasingly multilingual communities by increasing the percentage of non-target language speakers from zero to 50% in steps of one percent. For each target language, at each level of noise, I calculated the learner's probability of convergence to the target grammar from the worst initial grammar.

The addition of noise causes a more-or-less linear decrease in the probability of convergence for grammars that are learnable under noiseless conditions. This is plotted in Figure 2.

For those grammars that are not learnable under noiseless conditions, we see a stochastic resonance effect: the optimal performance of the TLA under these conditions is at a noise level of 11% (Languages 3 and 5) or 12% (Language 7). At this noise level, the probability of converging to the target grammar is 0.6658 for Language 3, 0.5468 for Language 5 and 0.6438 for Language 7. This is plotted in

[9]Since some of the triggers in the V2-X-bar space are generated by more than one grammar, it is possible that a 'noisy' trigger will actually be a trigger for the target grammar. It would be possible to control for this using a slightly different noise model, but I decided not to. The model that I use is designed to reflect multilingual communities, and the extent to which a pair of E-languages overlap is an empirical fact about the languages of the community. For example, an OVS datum is a trigger for languages with OVS word order, but it can also arise in VOS, SVO and SOV languages as a result of verb-second phenomena. One could imagine a collection of formal languages where the intersection between the sets of strings generated by those languages was null; such a case bears little resemblance to natural languages, however. The whole idea of distinctive features, parameters, and so on suggests that individual languages are built from elements common to natural language in general. The situation is muddied somewhat by the fact that different languages use different lexical items—I envision a situation where (say) the syntactic structure of one language is used with the lexical items of another, as might occur in cases of language contact. Obviously, there is much room for refinement here.

Figure 2: *Finite sample convergence from worst initial grammar deteriorates for Languages 1, 2, 4, 6 and 8 with increasing noise*

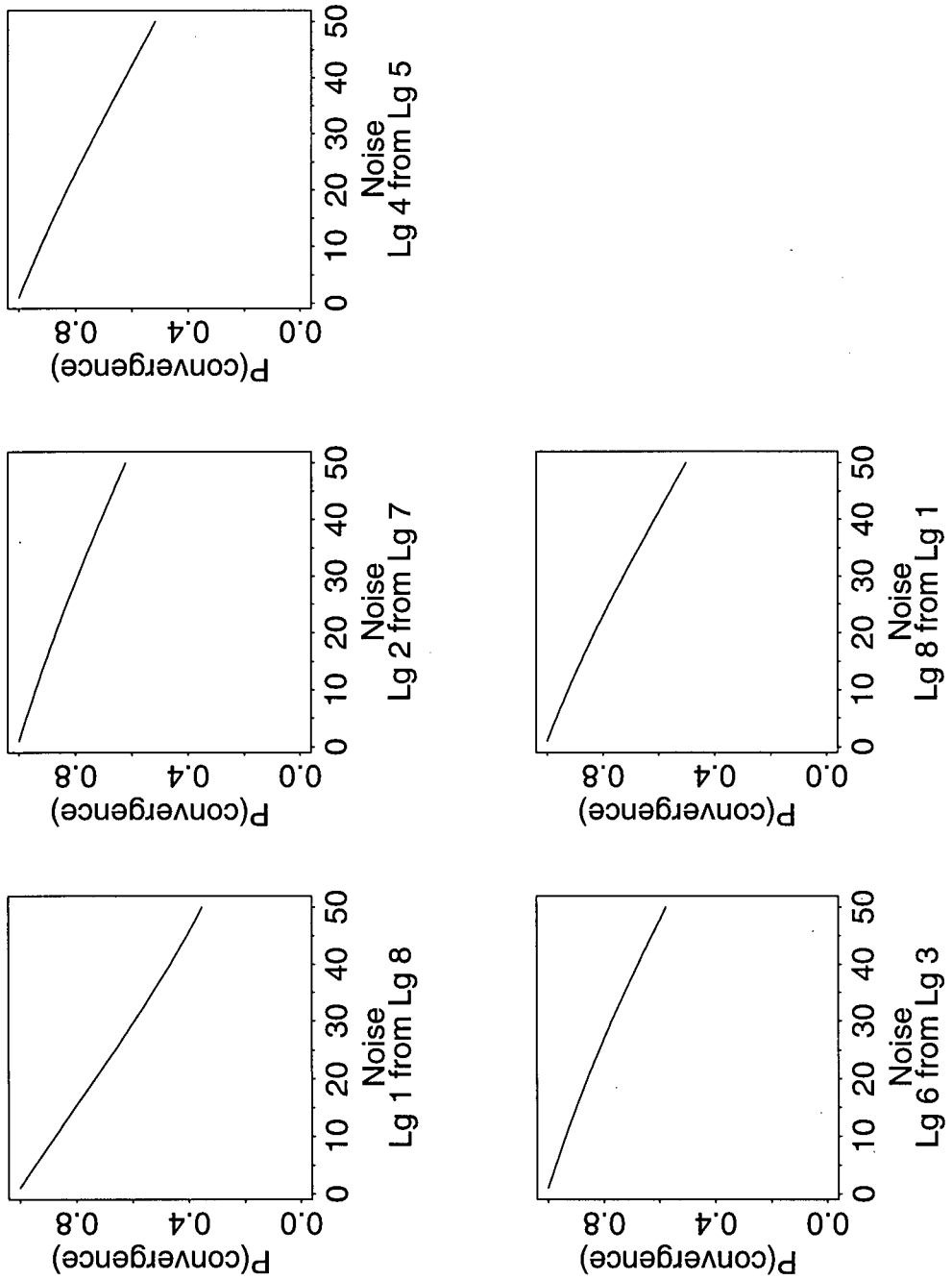


Figure 3.

Averaging across all eight grammars shows that the best overall performance of the TLA (when started in the worst initial grammar) occurs at a noise level of 8%, as shown in Figure 4.

Conclusion

In conclusion, noise in the input to language learners arises from a variety of sources. Performance errors, peripheral constructions and dialectal variation can all result in forms that the core target grammar does not generate. Presumably this problem is more acute in multilingual environments, where the child may receive forms from more than one target grammar. As Frank & Kapur (1996) argue, the problem of noisy input is inherent to the problem of grammatical acquisition and should be addressed in plausible parameter-setting models.

Rather than being a drawback, it may be that low levels of noise actually improve the performance of learning algorithms under some conditions. Where the target grammar is learnable without noise, the noise causes a degradation of performance. When local maxima prevent convergence to the target grammar, however, the noise can 'jiggle' the learner out of the trap, increasing performance. I proposed that this is analogous to stochastic resonance, a phenomenon which has been demonstrated in a variety of other biological systems. Whether noise can be shown to have a beneficial effect for other parameter-setting algorithms, or for the TLA in other

Figure 3: *Finite sample convergence from worst initial grammar for Languages 3, 5 and 7 shows stochastic resonance effect*

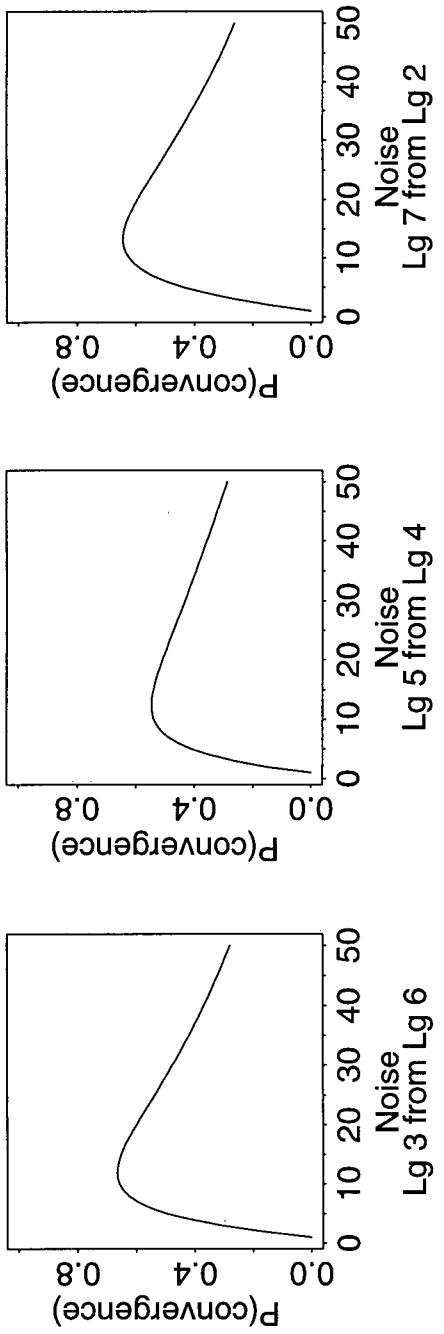
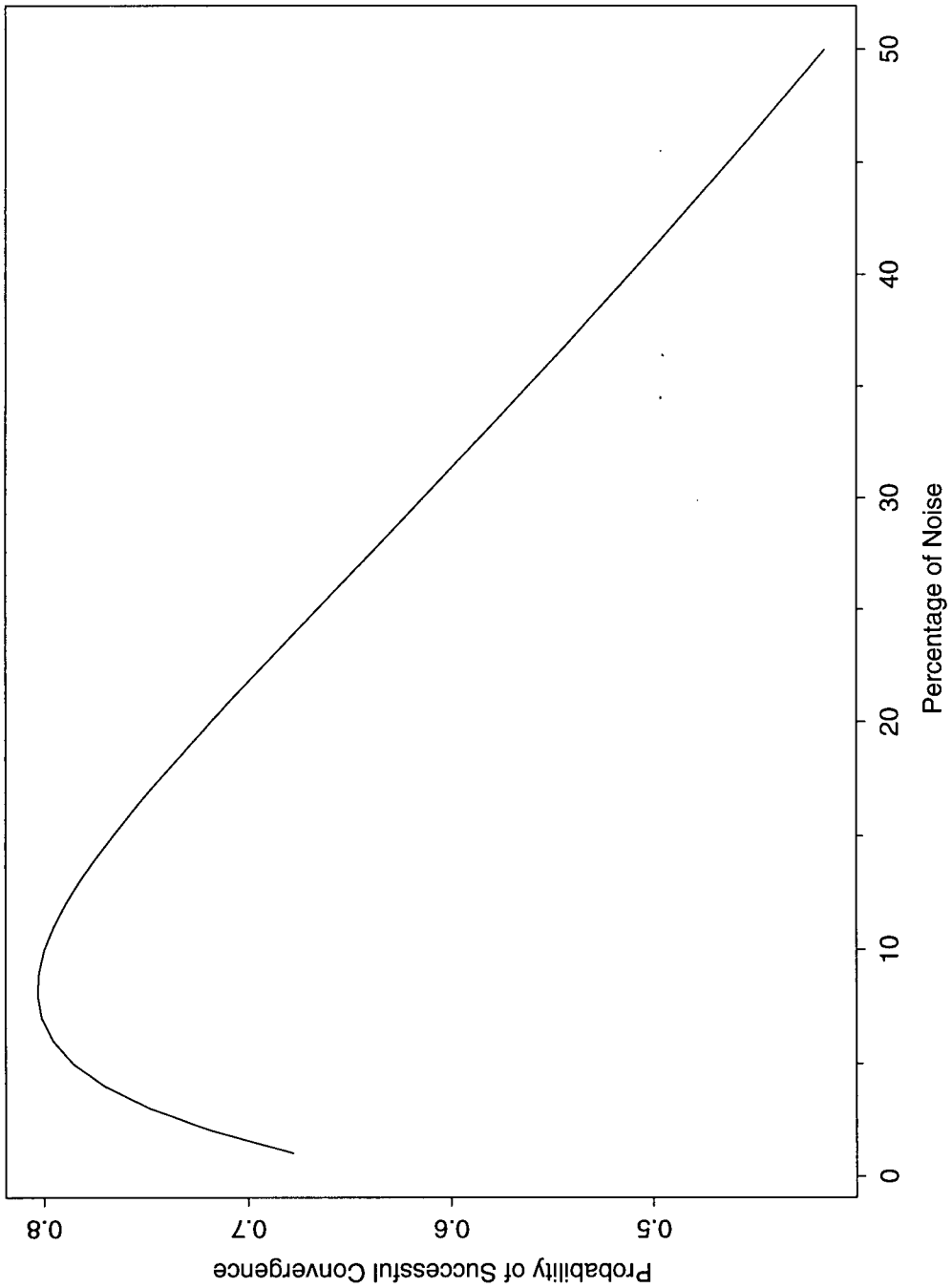


Figure 4: *Average finite sample convergence from worst initial grammar is optimal at a low level of input noise (8%)*



parametric spaces, remains to be investigated.

Chapter 4—Acquisition by a Genetic Algorithm-Based Model in Spaces with Local Maxima

Local Maxima

In this chapter, we consider the third component of the learning scenario, the learning algorithm itself. One of the main results of Gibson & Wexler (1994) was a demonstration that a sample parametric space contains *local maxima*, regions where a learning algorithm may become trapped on its way to the target grammar. Such traps are not limited to local maxima, of course. Frank & Kapur (1996:639) describe the *archipelago*—a set of grammars among which movement is possible, but from which there is no escape. The more traditional *subset problem* (Berwick 1985) is also faced by trigger-based learning algorithms. Here I concentrate on local maxima.

The finding that there are local maxima in a small parametric space suggests that any theory of parameter setting will have deal with them. The issue is how. The three places where we might attempt to overcome the problem correspond to the three parts of a typical learning scenario: the parametric space, the data available to the learner, or the behaviour of the learning algorithm.

Pursuing the first option, one might claim that the parameter space associated with Universal Grammar contains no local maxima, and that parameter spaces which do are not good characterizations of UG. I follow Gibson & Wexler in rejecting this option. Local maxima are partially due to parameter interdependence. To

the extent that we wish to have a highly deductive theory (Chomsky 1981) we must expect a reasonable degree of parameter interdependence. Consideration of linguistically well-motivated parameters (and constraints) suggests their interaction will tend to lead to local maxima. Although the question is ultimately empirical, I suspect the answer lies elsewhere.

Alternately, one might argue that there are data that have not yet been considered, which would allow the learning algorithm to move out of the local maximum. We must tentatively reject this option too. The only evidence likely to be of use to a learner trapped in a local maximum would be negative; for example, a learner might use corrections to determine that it was overgenerating, and so retreat to a grammar that generated a subset of the triggers generated by the local maximum.[10] In the absence of negative evidence, it is best not to base a theory of parameter setting on the possible discovery of enriched data.

The third option involves adjusting the learning algorithm. Local maxima are traps relative to particular classes of learners. By changing the properties of the learning algorithm, it is possible to create a learner which can escape or avoid the local maxima.

---

[10]Pulleyblank & Turkel (to appear b) make use of a mechanism they call *gradient retreat*, which allows principled retreat from subsets and local maxima without negative evidence. The mechanism relies on Optimality Theory, however, and cannot be directly applied in the parameter-setting case discussed here.

Genetic Algorithm-based Model of Parameter Setting

In this chapter, I explore the behaviour of a learning algorithm which is neither greedy nor conservative. I address four questions. First, is the algorithm able to converge on the target grammar without making any assumptions about the initial state? Second, is the behaviour of the algorithm *feasible*? Is it a realistic model of a child with limited memory and computational resources? Third, does the existence of local maxima affect the convergence of the algorithm? Fourth, how is the behaviour of the algorithm likely to scale as the size of the space is increased?

Clark (1992a) and Clark & Roberts (1993) describe a model of language acquisition based upon *genetic algorithms* (GA) (see Forrest 1993, Goldberg 1989). In a GA-based theory, the learning algorithm creates a number of hypotheses at one time. These are assessed against the data, and each is given a *fitness*. The hypotheses are then *mutated*, and those with the highest fitness are preferentially *recombined* to give rise to a new set of hypotheses. These are again tested against the data and the whole process repeated. GA-based models are not conservative, because the mechanism of recombination can give rise to a hypothesis which is arbitrarily far away from either "parent" hypothesis. Neither are they greedy; change in hypotheses is independent of the utility of the new hypothesis.

I implemented a GA-based model to test in Gibson & Wexler's (1994) V2-X-bar parameter space. The space consists of three binary parameters. The complement-head parameter allows complements to be initial or final in X-bar structures. The

specifier-head parameter allows specifiers to be initial or final, and the verb-second parameter states that in +V2 languages Comp must be marked for finiteness. Although this set of parameters is controversial with respect to syntactic analysis, I retain it because my intention is to compare the performance of learning algorithms under constant conditions. The reader is encouraged to consult Frank & Kapur (1996, Section 5) for a discussion of the parametric analysis of verb second phenomena.

There are eight possible ways to set the three parameters, each corresponding to a target grammar. Each of the target grammars is consistent with a set of degree-0 data, specified as patterns. As an example, setting the parameters to Spec-Final, Comp-Final, -V2 results in a VOS language. This language will include VS, VOS, and AuxVS among its data, but not SOV. A complete listing of the data is given in Chapter 1.

My implementation operated on a space of two hypotheses at a time, each hypothesis represented as a vector of three bits. The bit vector corresponded to a possible array of parameter settings, extensionally specifying a possible target language. Each hypothesis was tested against four example sentence patterns randomly drawn from the set admitted by the target grammar. I assumed that the examples were distributed uniformly. Fitness for each hypothesis was determined by the number of examples the hypothesis was consistent with, expressed as a fraction of the number of examples it was tested against, resulting in fitness values drawn from

the set {0, 0.25, 0.5, 0.75, 1}. The fitness values were scaled, so that the sum of the fitnesses for the population was equal to 1. Two hypotheses were then drawn from the population with a probability equal to their scaled fitness value. The two new hypotheses were subject to mutation and recombination, yielding a final pair of hypotheses for the next iteration of the acquisition process. Mutation consisted of flipping a single parameter value, and was applied with a probability of 0.25. Recombination was *one-point crossover* (as described in Clark 1992a, Clark & Roberts 1993, Forrest 1993 and elsewhere) applied with a probability of 0.5. Convergence was defined as the first time at which both hypotheses were equal to the target parameter setting.[11] During each iteration of the system, the learner was exposed to eight example sentences drawn from the target language. The soonest the learner could converge was in one iteration.

Convergence to each of the eight possible target grammars was tested under two assumptions about the initial configuration of the system. In the first case, both of the starting hypotheses were randomly generated. In the second case, both of the starting hypotheses were the same, and were set to one of the other seven possible grammars. Each of the 64 tests was run 100 times. Table 3 shows the minimum, median, and maximum number of (positive) examples required to converge for each target grammar and each type of initial configuration.[12] Unlike the

[11]The probability of this happening by chance is $\frac{1}{8^2} = \frac{1}{64}$.

[12]Recall that the GA receives eight data per iteration. This means that minimum and maximum number of examples required to converge will always be a multiple of 8. Tables in this chapter

Table 6: *Examples required for convergence, all source-target pairs*

Target	Random Start			Specified Start		
	Minimum	Median	Maximum	Minimum	Median	Maximum
VOS	8	64	480	8	120	896
VOS +V2	8	92	640	8	112	696
OVS	8	76	688	16	152	1392
OVS +V2	8	120	952	8	128	1176
SVO	8	128	944	8	160	1248
SVO +V2	8	84	552	8	112	1232
SOV	8	80	1176	16	128	1088
SOV +V2	8	120	872	8	136	1032

TLA, the GA-based learner does not need to start in a particular configuration for successful acquisition in the V2-X-bar parameter space. In every case, the algorithm converged, making stipulations about the initial state unneccessary for a GA-based learner. There is another feature of interest in Table 3. The GA-based learner does better when given a random start. There are two reasons why this might be the case. First of all, the specified start involves having a pair of identical hypotheses, thus disrupting the GA's ability to efficiently search the parameter space via recombination. The second reason is that the fitness measure works most effectively when the two hypotheses are different. In the specified start case, they will be the same until mutation changes one, at which point fitness and recombination can come into play.

are expressed in number of examples (rather than number of iterations) to facilitate comparison of the two algorithms, and to allow comparison of results with the convergence curves in Chapter 3, which are the predicted results for the TLA given 300 examples.

The second question I addressed was the feasibility of the model. Niyogi & Berwick (1993) found that the TLA converged with high probability within 100 to 200 example sentences, a number that they argue is psychologically plausible. The median performance of the GA-based algorithm is comparable, particularly under the assumption of randomly generated initial hypotheses. The worst case performance requires about an order of magnitude more positive examples. So the GA-based model is feasible in terms of the number of examples the learner needs to be exposed to.

One concern might be the amount of storage that the GA-based learner requires, given a general interest in restricting the amount of memory available to the learning algorithm. Under one interpretation, hypotheses are created and tested in parallel, requiring the explicit storage of eight example sentences at one time. Since the evaluation of fitness is independent for different hypotheses, there is also a sequential interpretation of the GA-model. In this case, explicit storage is limited to two example sentences at one time.

A possible objection is that if the TLA were modified to use a number of simultaneous hypotheses its performance in the V2-X-bar space would probably be improved. For example, the TLA might be revised as follows.

> Given a set of vectors, each of which contains settings for n binary-valued parameters, the learner attempts to syntactically analyze an incoming sentence S with the grammar specified by the first vector. If S can be successfully analyzed, then the learner's hypothesis regarding the target grammar is left unchanged. If, however, the learner cannot analyze S,

> then the learner uniformly selects a parameter P (with probability $1/n$ for each parameter), changes the value associated with P and tries to reprocess S using the new parameter value. If analysis is now possible, then the parameter value change is adopted. Otherwise, the original parameter value is retained, and the learner goes on to try the grammar specified by the next vector. The learner is said to converge when one of the vectors matches the target grammar.

In effect, this allows the TLA to maintain a number of trajectories at once. This modification is not likely to scale well with an increase in the size of the space, however, because the modified TLA would have to sacrifice one of its hypotheses for each local maximum it encountered. Hypotheses in the GA-based learner which become trapped in local maxima are soon eliminated (via selection) in favour of better hypotheses which do not. This is because the fitness of a hypothesis in a local maximum can only be so good. Once some other hypothesis has a greater fitness, that hypothesis has a selective advantage, and the hypothesis which is trapped in the local maximum soon dies out. Thus, unlike the modified TLA, the GA-based learner does not lose the capacity to entertain multiple hypotheses as it encounters local maxima.

The third question I addressed was the degree to which the local maxima affected the operation of the learning algorithm. In addition to the six unlearnable source-target grammar pairs presented in Gibson & Wexler 1994, Niyogi & Berwick found another six pairs that probabilistically lead to an unlearnable situation. These are grammars that are not local maxima themselves, but which are on a trajectory to local maxima. Minimum, median, and maximum numbers of examples required for

convergence for each of the twelve unlearnable source-target pairs are presented in Tables 4 and 5. Convergence is slowed appreciably for these source-target pairs, despite the fact that the algorithm is not greedy or conservative. For example, four of the seven source grammars are local maxima with respect to SVO. The GA-based learner entertained local maxima as hypotheses more frequently (about 56% of total hypotheses) while acquiring SVO from a local maximum than it did while acquiring from grammars that are not local maxima with respect to SVO (about 41% of total hypotheses). The genetic algorithm can operate by populating local maxima rather than by avoiding them, as the TLA must (Winston 1992). This suggests that the problem of local maxima extends beyond variants of the TLA, and will need to be considered in any parameter setting account. This stands to reason; any algorithm that moves towards the target via improvement steps (rather than moving randomly) is liable to be occasionally led into local maxima. This is because local maxima are an improvement relative to neighbouring hypotheses.

The fourth question I addressed was the likelihood that the genetic algorithm would continue to converge feasibly as the size of the space was increased. An extensive literature (Goldberg 1989 cites 83 studies) has shown that GA-based methods can be applied robustly in a wide variety of global optimization tasks. (Global optimization is the determination of the global maximum of a function of an arbitrary number of independent variables). Cvijović & Klinowski (1995) note that the genetic algorithm is able to avoid entrapment in local maxima and continue the

Table 7: *Examples required for convergence, unfavorable source-target pairs from Gibson & Wexler 1994*

Source	Target	Minimum	Median	Maximum
SVO +V2	OVS	40	176	1016
SOV +V2	OVS	24	188	1344
VOS +V2	SVO	32	212	1192
OVS +V2	SVO	40	200	1248
VOS +V2	SOV	48	196	1088
OVS +V2	SOV	32	188	1000

Table 8: *Examples required for convergence, unfavorable source-target pairs from Niyogi & Berwick 1993*

Source	Target	Minimum	Median	Maximum
SVO	OVS	24	144	632
SOV	OVS	32	184	696
VOS	SVO	16	144	1120
OVS	SVO	24	216	1048
VOS	SOV	24	136	752
OVS	SOV	16	72	736

search to give a near-optimal solution whatever the initial conditions, the principle requirement of any global optimization method. Problems that are difficult for genetic algorithms tend to have isolated maxima, and are thus difficult for many optimization methods (Goldberg 1989). I conclude that a GA-based learner is likely to continue to perform well as the size of the space increases (cf. performance reported by Clark & Roberts (1993) for a space of $2^5 = 32$ grammars, and by Pulleyblank & Turkel (to appear a) for a GA-based learner in an 11 constraint OT space with $11! = 39916800$ grammars).

Conclusion

In conclusion, the GA-based learner shows feasible acquisition in the V2-X-bar parametric space, and is likely to do so in other parametric spaces, although the sufficiency of the algorithm in general is an important open question. Unlike the TLA, the GA-based learner can start with any initial configuration of parameters, eliminating the need for (or the desirability of) the stipulation of distinguished initial states. Such an account also suggests that parametric markedness may be the result of other factors than default settings.[13] The space and data requirements of the GA-based learner are not significantly greater than those of the TLA. Convergence is delayed when the learner is forced to traverse regions of hypothesis space containing local maxima, although the model is still feasible under these conditions.

[13]See Pulleyblank & Turkel (1996, to appear c) for an alternative approach to markedness phenomena in the framework of triggered learning.

Chapter 5—Conclusion

In this thesis, I have considered a number of computational learning scenarios. Each of these consisted of a parametric space, input to a learner, and a learning algorithm. These scenarios were meant to suggest ways in which parameter setting might occur or be constrained. Following Brent (1996), I called the scenarios *how* theories.

We can relate these theories to theories at the linguistic level. Typically, linguistic theories specify *what* happens and *why*. Using information from *how* theories, we can constrain *what/why* theories by requiring that they have effective implementations or other desirable properties.

For example, in Chapter 2 we saw that it is possible to measure the smoothness of a parameter space. A smoothness result for a given space is desirable to the extent that we wish to have neighbouring grammars generate similar languages. Given a pair of parameter sets with similar empirical coverage, smoothness measures can be used to pick one over the other.

In Chapter 3 we saw that noise defined relative to a particular learning algorithm may aid convergence. This result suggests that we should consider not only the ability of learning algorithms to identify a target grammar, but the convergence properties of learners given a mixture of triggers.

In Chapter 4 we saw that the kinds of traps that disrupt the performance of one kind of learning algorithm are not problematic for another learning algorithm.

One possibility is to try and limit the power of the learner, and then use a limited learner as a constraint on the development of linguistic theories. This is the path that Gibson & Wexler pursue (1994; also Bertolo, Broihier, Gibson & Wexler 1997a, 1997b). Such a strategy may be problematic, however. For example, Gibson & Wexler (1994) suggest that linguistically natural parameter spaces may have a distinguished initial grammar that will enable the learning algorithm not to get stuck in a local maximum. Work by Broihier (1995) and Pulleyblank & Turkel (to appear a, to appear b) shows that this is a problematic assumption. Furthermore, the psychological plausibility of the Triggering Learning Algorithm has been challenged by a number of authors (e.g., Berwick & Niyogi 1996, Brent 1996, Frank & Kapur 1996).

As a model of parameter setting, triggered learning foregrounds certain aspects of the problem at the expense of others. The structure of the algorithms suggest a search for the triggers of particular languages and for parametric differences between languages. Factors such as memory and continuity are highlighted. Other potentially relevant factors are completely ignored. I have already noted that the continuity and stationarity hypotheses are problematic. In addition, the simplified view of linguistic input as a sequence of abstract triggers is probably too simple. A more complete model will have to take into account perceptual development, infant-directed speech, caretaker-infant interaction, lexical acquisition, cross-cultural differences in language socialisation, and so on.

Pulleyblank & Turkel (to appear a) argue that different learning algorithms provide us with a rich characterisation of some aspect or aspects of the problem of language acquisition. For example, one learning algorithm that has been invoked from time to time (almost always as a straw man, e.g., Clark 1992b) is the *enumerative learner*. Such a learner operates by systematically considering each grammar in turn, until it converges on the target grammar. Consideration of the performance of such a learning algorithm brings out two aspects of triggered learning particularly forcefully. The first is that any reasonable set of parameters (or constraints in Optimality Theory) is going to give rise to a space which is far too large to search in this manner. The second is that a given enumeration of grammars is not necessarily going to have desirable smoothness properties: the grammar considered at a given time could be unrelated to the grammar considered previously—it doesn't matter from the perspective of the learning algorithm. This is not to say that enumerative learning could not form the basis of a more sophisticated learner; Wu (1994), for example, proposes a principled enumeration of the space of a set of parameters based on Chomsky (1995).

The study of computational learning scenarios can provide us with a way of constraining the development of linguistic theories. Given our current state of knowledge, it is probably best not to attempt to relate computational learning directly to the phenomenal level. Triggered learning is too simple to provide a model of the ontogenesis of the human child. As a model of how linguistic constructs may be

established, however, it has much to offer.

Bibliography

Bertolo, Stefano. 1995. Maturation and learnability in parametric systems. *Language Acquisition* 4(4):277.

Bertolo, Stefano, Kevin Broihier, Edward Gibson & Kenneth Wexler. 1997a. Characterizing learnability conditions for cue-based learners in parametric language systems. Ms., MIT, Cambridge, MA.

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'. Ms., MIT, Cambridge, MA.

Berwick, Robert C. 1985. *The acquisition of syntactic knowledge.* Cambridge, MA: MIT Press.

Berwick, Robert C. & Partha Niyogi. 1996. Learning from triggers. *Linguistic Inquiry* 27(4):605-622.

Bloom, Paul. 1994. Recent controversies in the study of language acquisition. In Morton Ann Gernsbacher, Ed. *Handbook of psycholinguistics.* San Diego, CA: Academic Press, pp. 741-779.

Brent, Michael R. 1996. Advances in the computational study of language acquisition. *Cognition* 61(1/2):1-38.

Broihier, Kevin. 1995. Phonological triggers. Presentation given at Maryland Mayfest 95: Formal Approaches to Learnability, College Park, MD.

Brown, Roger & Camille Hanlon. 1970. Derivational complexity and the order of acquisition in child speech. In John R. Hayes, Ed. *Cognition and the development of language.* New York: Wiley.

Chomsky, Noam. 1965. *Aspects of the theory of syntax.* Cambridge, MA: MIT Press.

Chomsky, Noam. 1981. *Lectures on government and binding.* Dordrecht: Foris.

Chomsky, Noam. 1986. *Knowledge of language.* New York: Praeger.

Chomsky, Noam. 1988. *Language and problems of knowledge: The Managua lectures.* Cambridge, MA: MIT Press.

Chomsky, Noam. 1995. *The minimalist program.* Cambridge, MA: MIT Press.

Clark, Robin. 1989. On the relationship between the input data and parameter setting. In Juli Carter & Rose-Marie Déchaine, Eds., *Proceedings of NELS 19.* Amherst, MA: GLSA, University of Massachusetts, Amherst.

Clark, Robin. 1992a. The selection of syntactic knowledge. *Language Acquisition* 2:85-149.

Clark, Robin. 1992b. Finitude, boundedness and complexity: Learnability and the study of first language acquisition. Ms., University of Pennsylvania, Philadelphia.

Clark, Robin & Ian Roberts. 1993. A computational model of language learnability and language change. *Linguistic Inquiry* 24:299-345.

Cvijović, Djurdje & Jacek Klinowski. 1995. Taboo search: An approach to the multiple minima problem. *Science* 267:664-666.

Dresher, B. Elan. 1994. Acquiring stress systems. In Eric Sven Ristad, Ed., *Language computations.* DIMACS Series in Discrete Mathematics and Theoretical Computer Science 17:71-92.

Dresher, B. Elan & Jonathan D. Kaye. 1990. A computational learning model for metrical phonology. *Cognition* 34:137-195.

Elman, Jeffrey L. 1993. Learning and development in neural networks: The importance of starting small. *Cognition* 48:71-99.

Fodor, Janet Dean. 1989. Learning the periphery. In Robert J. Matthews & William Demopoulous, Eds., *Learnability and linguistic theory.* Dordrecht: Kluwer, pp. 129-154.

Forrest, Stephanie. 1993. Genetic algorithms: Principles of natural selection applied to computation. *Science* 261:872-878.

Frank, Robert & Shyam Kapur. 1996. On the use of triggers in parameter setting. *Linguistic Inquiry* 27(4):623-660.

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

Gillis, Steven & Gert Durieux. 1996. Data-driven approaches to phonological acquisition: An empirical test. In Barbara Bernhardt, John Gilbert & David Ingram, Eds., *Proceedings of the UBC International Conference on Phonological Acquisition.* Somerville, MA: Cascadilla Press.

Gillis, Steven, Gert Durieux, Walter Daelemans. 1995. A computational model of P & P: Dresher & Kaye (1990) revisited. In F. Wijnen & M. Verrips, Eds., *Approaches to Parameter Setting.* Amsterdam series in child language development 4:135-173. Amsterdam: Universiteit van Amsterdam.

Gold, E. Mark. 1967. Language identification in the limit. *Information and Control* 10:447-474.

Goldberg, David E. 1989. *Genetic algorithms in search, optimization and machine learning.* Reading, MA: Addison-Wesley.

Gupta, Prahlad & David S. Touretzky. 1991. What a perceptron reveals about metrical phonology. In *Proceedings of the 13th Annual Conference of the Cognitive Science Society*, Hillsdale, NJ: Lawrence Erlbaum.

Gupta, Prahlad & David S. Touretzky. 1992. A connectionist learning approach to analyzing linguistic stress. In J. E. Moody, S. J. Hanson & R. P. Lippmann, Eds., *Advances in Neural Information Processing Systems* 4:225-232. San Mateo, CA: Morgan Kaufmann.

Kruskal, Joseph B. 1983. An overview of sequence comparison. In David Sankoff & Joseph B. Kruskall, Eds., *Time warps, string edits, and macromolecules.* Reading, MA: Addison-Wesley, pp. 1-44.

Lightfoot, David. 1991. *How to set parameters: Arguments from language change.* Cambridge, MA: MIT Press.

Marcus, Gary F. 1993. Negative evidence in language acquisition. *Cognition* 46:53-85.

Marr, David. 1982. *Vision.* San Francisco: W. H. Freeman and Company.

Niyogi, Partha & Robert C. Berwick. 1993. Formalizing triggers: A learning model for finite spaces. AI Memo 1449, CBCL Paper 86. Cambridge, MA: MIT.

Niyogi, Partha & Robert C. Berwick. 1995a. A dynamical systems model for language change. AI Memo 1515, CBCL Paper 114. Cambridge, MA: MIT.

Niyogi, Partha & Robert C. Berwick. 1995b. The logical problem of language change. AI Memo 1516, CBCL Paper 115. Cambridge, MA: MIT.

Niyogi, Partha & Robert C. Berwick. 1996. A language learning model for finite parameter spaces. *Cognition* 61:161-193.

Pinker, Steven. 1981. Comments on paper by Wexler. In Carl Lee Baker & John J. McCarthy, Eds., *The logical problem of language acquisition.* Cambridge, MA: MIT Press.

Pinker, Steven. 1984. *Language learnability and language development.* Cambridge, MA: Harvard University Press.

Prince, Alan & Paul Smolensky. 1993. *Optimality theory: Constraint interaction in generative grammar.* Technical report RUCCS 2. New Brunswick, NJ: Rutgers University Center for Cognitive Science.

Prince, Alan & Paul Smolensky. 1997. Optimality: From neural networks to universal grammar. *Science* 275(5306):1604.

Pulleyblank, Douglas & William J. Turkel. 1996. Optimality theory and learning algorithms: The representation of recurrent featural asymmetries. In Jacques Durand & Bernard Laks, Eds., *Current trends in phonology: Models and methods. Volume 2.* European Studies Research Institute, University of Salford Publications, pp. 653-684.

Pulleyblank, Douglas & William J. Turkel. To appear a. The logical problem of language acquisition in optimality theory. In Pilar Barbosa, Danny Fox, Paul Hagstrom, Martha Jo McGinnis & David Pesetsky, Eds., *Is the best good enough?* Cambridge, MA: MIT Press and MIT Working Papers in Linguistics.

Pulleyblank, Douglas & William J. Turkel. To appear b. Gradient retreat. In Iggy Roca, Ed., *Derivations and constraints in phonology.* Oxford: Oxford University Press.

Pulleyblank, Douglas & William J. Turkel. To appear c. Markedness asymmetries in tongue root harmony as an emergent property of competence/performance interaction. In Joost Dekkers, Frank van der Leeuw & Jeroen van de Weijer, Eds., *The Pointing Finger: Conceptual Studies in Optimality Theory.* Oxford: Oxford University Press.

Quartz, Steven R. 1993. Neural networks, nativism, and the plausibility of constructivism. *Cognition* 48:223-242.

Quartz, Steven R. & Terrence J. Sejnowski. 1995. The neural basis of cognitive development: A constructivist manifesto. Ms., Salk Institute for Biological Studies, San Diego.

Turkel, William J. 1996a. Acquisition by a genetic algorithm-based model in spaces with local maxima. *Linguistic Inquiry* 27(2):350-355.

Turkel, William J. 1996b. Biological metaphor in models of language acquisition. In Barbara Bernhardt, John Gilbert & David Ingram, Eds., *Proceedings of the UBC International Conference on Phonological Acquisition.* Somerville, MA: Cascadilla Press, pp. 266-276.

Wexler, Kenneth. 1991. On the argument from the poverty of the stimulus. In Asa Kasher, Ed., *The Chomskyan turn.* Cambridge, MA: Blackwell.

Wiesenfeld, Kurt & Frank Moss. 1995. Stochastic resonance and the benefits of noise: From ice ages to crayfish and SQUIDs. *Nature* 373:33-36.

Winston, Patrick H. 1992. *Artificial intelligence.* Reading, MA: Addison-Wesley.

Wu, Andi. 1994. *The spell-out parameters: A minimalist approach to syntax.* PhD dissertation, University of California, Los Angeles.