Markedness, antisymmetry and complexity of constructions*

Peter W. Culicover and Andrzej Nowak
The Ohio State University / University of Warsaw

Our concern in this paper is with the interactions between language change, language acquisition, markedness, and computational complexity of mappings between grammatical representations. We demonstrate through a computational simulation of language change that markedness can produce 'gaps' in the distribution of combinations of linguistic features. Certain combinations will not occur, simply because there are combinations that are computationally less complex. We argue that one contributor to markedness in this sense is the degree of the transparency of the mapping between superficial syntactic structure and Conceptual Structure. We develop a rough measure of complexity that takes into account the extent to which the syntactic structure involves stretching and twisting of the relations that hold in Conceptual Structure, and we show how it gives the right results in a number of specific cases.

Keywords: language change, language acquisition, computational simulation, markedness, computational complexity, syntactic complexity, language learnability

1. Introduction

One of the strongest arguments for the thesis that the human mind possesses a Universal Grammar (UG) with specific grammatical properties is that languages do not appear to have arbitrary and uncorrelated properties. What we find, rather, is that the properties of languages cluster, and that there are asymmetries among the logical possibilities. For example, VSO languages are always prepositional, and SOV languages are usually postpositional (Greenberg 1963:78–79). There are languages that express *wh*-questions using leftward movement to a peripheral

Linguistic Variation Yearbook 2 (2002), **3–28.**
ISSN 1568–1483 / E-ISSN 1569–9900

position in the clause, and there are languages that express *wh*-questions without overt movement. But there do not appear to be languages that express *wh*-questions using rightward movement to a peripheral position in the clause.

It is natural, given observations such as these, to posit that they are direct reflections of UG, which the language learner draws upon in choosing or constructing grammars. However, there are two other possibilities that have to be ruled out before such a conclusion can be drawn. First, the clustering of properties and the absence of certain logical possibilities may be due to social forces. In such a case we would not expect to find the same asymmetries in different parts of the world where languages are not genetically related or in contact. Second, these asymmetries may be due to the interaction between the grammatical or processing complexity of certain constructions and social forces. On this view, all of the logical possibilities are linguistic possibilities, but those that are more complex tend to lose out over time to their less complex competitors as linguistic knowledge is transmitted from generation to generation in a network of social interactions.

The intention of this paper is to explore and make somewhat more precise these scenarios. We make the background assumption that language change occurs in part as the consequence of different learners being exposed to different evidence regarding the precise grammar of the language that they are to learn. Following the original insight of Chomsky (1965), we assume that learners chose the most economical grammar consistent with their experience, and even overlook counterevidence to the most economical solution unless the counterevidence is particularly robust. It is reasonable to understand economy in terms of the complexity of the grammatical representation that is to be learned (although there are many other ideas around). To the extent that learners reduce complexity we will then expect language change to reflect this preference in the relative ubiquity of certain grammatical devices compared with others, and even in the appearance of universals (Briscoe 2000).

We will begin by illustrating the ways in which language change gives rise to correlations of properties; it will be demonstrated that some combinations are excluded purely as a consequence of social factors that have nothing to do with their linguistic content. We then note that if there is a bias in favor of some combination of properties, this results in a uniform pattern that cannot be explained in purely social terms.

This observation takes us to a consideration of the factors that determine complexity in this context. We suggest, following up on an idea in Culicover (1999) based on work of Hawkins (1994), that the complexity in this case is that

of the mapping between strings of words and conceptual structure (in the sense of Jackendoff 1990). In a fairly transparent sense such mappings define 'constructions', and the relative generality of a construction is determined by its grammatical complexity.[1]

2. Change and clustering

Imagine a society of speakers of a language, some of them competent speakers and some of them learners. Each speaker interacts with each of the other speakers with some frequency, in part as a function of the distances between them. (Distance may be physical and/or social.) As a consequence of drift, noise in the information channels, conscious innovation and contact with other languages there will be linguistic diversity in this society. Some learners may have considerable experience with diversity, others may have very little. Over the course of generations, learners interact with speakers whose language is determined by interactions with similar speakers, so that there is a consistency of grammar that may distinguish the social group from another, more distant group.

2.1 The simulation model

In order to test the general properties of the interaction between language learning and language change we developed a simulation model of social interaction based on the theory of social impact due to Latané and computational simulations based on this theory developed at the Center for Complex Systems at the Institute for Social Studies of the University of Warsaw by Andrzej Nowak and his colleagues.[2] Our intuition was that the transmission and clustering of linguistic properties though social contact should display the essential properties of the transmission and clustering of any cognitive features.

2.2 Gaps

2.2.1 *How gaps arise*

We suppose for the sake of the simulation that the class of possible grammars of natural languages can be characterized entirely in terms of values of features.[3] A prevalent view in current linguistic theory is that most if not all of the most theoretically interesting aspects of language variation, language change and language acquisition can be accounted for in terms of a small set of binary

features, called 'parameters'. For our purposes, however, it is sufficient to assume that whatever the features are, however many there are, and whatever values they have, learners are influenced to adopt the values of their community through social interaction.

Our simulation supposes that there are three two-valued features, which define eight distinct languages.

(1) +F1,+F2,+F3
+F1,+F2,−F3
+F1,−F2,+F3
+F1,−F2,−F3
−F1,+F2,+F3
−F1,+F2,−F3
−F1,−F2,+F3
−F1,−F2,−F3

Gaps occur when certain feature combinations are not attested. Our simulation shows that gaps may arise over the course of time, as the values of two of the features become strongly correlated. To take a simple example, if the geographical distribution of [−F2] becomes sufficiently restricted, it may fail to overlap with [+F1]. That is, [+F1] and [+F2] become highly correlated. In such a case, some of the languages, namely those with [+F1,−F2], will cease to exist. Such a situation may occur simply as a consequence of the social structure, and in itself tells us nothing interesting about the relationship between [+F1] and [−F2].

For the simulation, we may assume that at the outset of the simulation all possible combinations of features are possible (the 'Tower of Babel' state). The reasoning is that if certain combinations fail to exist after some period of time, this fact must be due to social factors, since there are no initial gaps. If we allowed for initial gaps, that is, innate implicational universals, then the appearance down the line of gaps would not provide any evidence about the effect of social interaction on the distribution and clustering of linguistic properties.

Figure 1 shows the random distribution of feature values for three features in a population of 2500 (=50×50). The upper lefthand image shows the distinct languages as differences on the gray scale. The other images show the distribution of + and − values for the three features FIRSTs, SECONDs and THIRDs.

The population of each of the eight languages is shown in the histogram in Figure 2. As can be seen, the languages are distributed more or less evenly over the entire population, as would be expected from a randomized assignment of feature values.

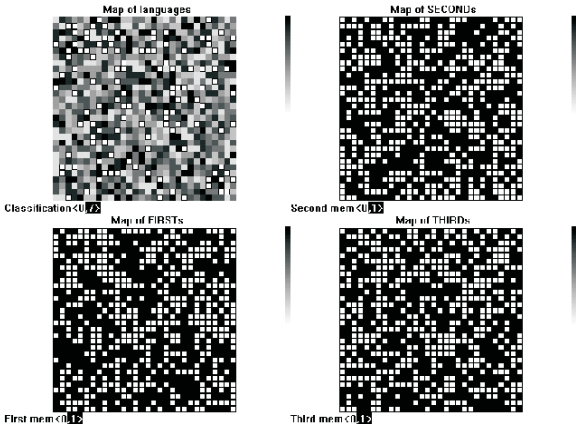


Figure 1. Initial random distribution of feature values.

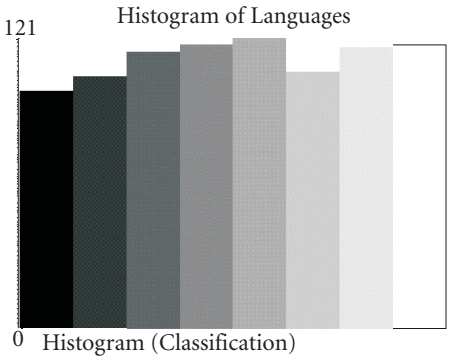


Figure 2. Population of the eight languages.

We have omitted intermediate steps in the simulation for reasons of time and space. After 69 steps the distribution of languages and features is as in Figure 3.

The histogram in Figure 4 shows the population levels of the eight languages at this point.

The loss of languages illustrated in this particular instance of the simulation is not unique. It is a consequence of the particular assumptions made in the simulation about how individuals interact in the network. Running the same simulation under the same parameters yields a different pattern of features and languages each time, but the results are the same. We repeated this simulation

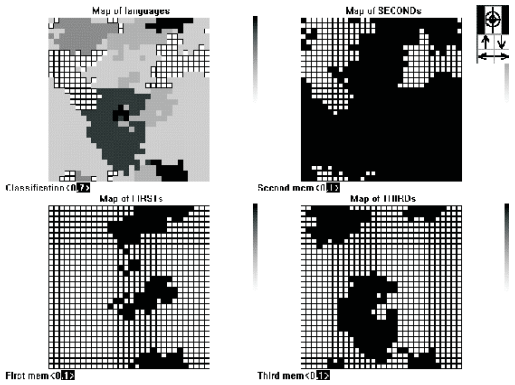


Figure 3. Distribution of languages and features after 150 steps.

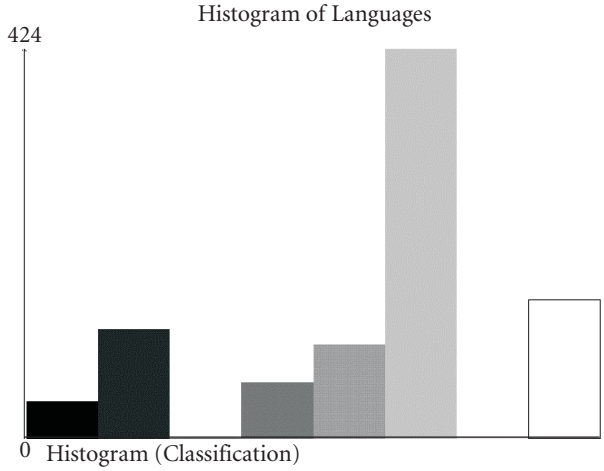


Figure 4. Population of languages after 150 steps.

100 times. The following chart shows the number of times a given number of languages remained in the simulation after 200 steps.

In 50 of the 100 runs of the simulation there were eight languages after 200 steps. But in 32 runs there were 7 languages, in 10 runs there were 6 languages, and so on. So while the precise number of languages that will remain after a certain number of steps is not predictable, it is clear that gaps in the set of languages can and will arise over the course of time as a consequence of the interaction in the network. The chart in Figure 6 shows that over a longer time span the number of languages for the same simulation tends to decline.

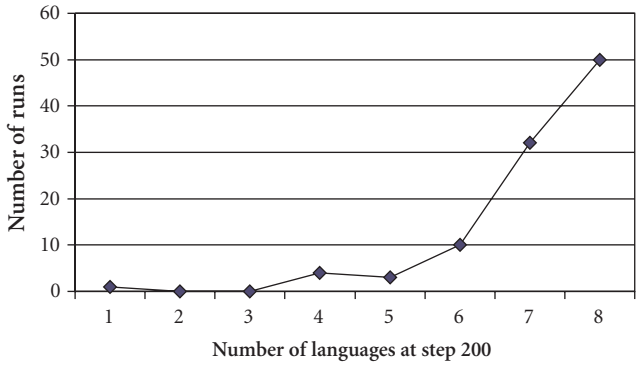


Figure 5. Loss of languages in repeated simulation.

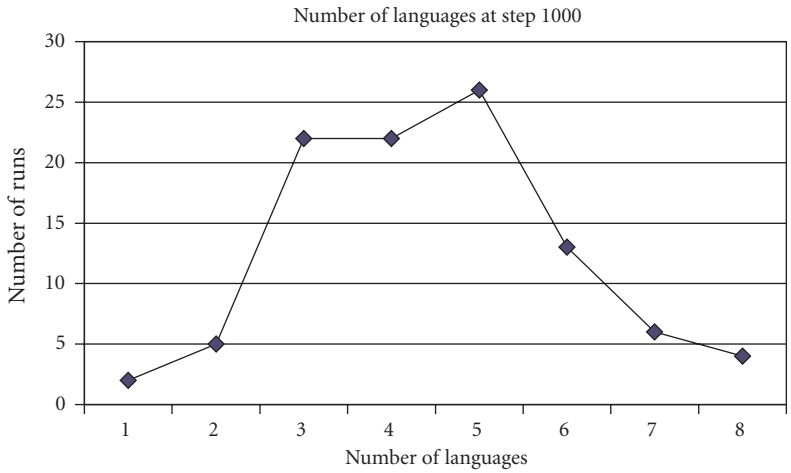


Figure 6. Distribution of languages after 1000 steps.

2.2.2 *Gaps and bias*

Let us now introduce bias into our simulation. Suppose that a particular combination of features, say [+F1,−F2], is less preferred than the other three combinations of these two features. On any run of the simulation model the results will look like those we have already seen. However, on every run of the simulation model the results will be more or less the same, in that there will be gaps or immanent gaps in [+F1,−F2] languages. It is known that simulations that assume bias in general show a clustering towards the same stable state;[4] the strength of the bias determines the predictability of the outcome.

This behavior of the simulation model suggests that it might be productive to look at the content of particular feature combinations in order to determine

what it is about them that yields more or less complexity. There are a number of candidates for complexity that should be considered.

– Optimality theory as applied to syntax posits that particular structures are produced by rules that violate various constraints. Given a particular formulation that captures a general tendency or a universal, it would be natural to ask what it is about the particular constraints that yields the observed ranking, since OT theory itself is not a theory of where the rankings come from. On the other hand, OT allows for different rankings of the same constraints, which suggests a priori that it might not shed much light on the question of whether there is an independent universal metric that ranks particular structures with respect to complexity.
– Chomsky's Minimalist Program (Chomsky 1994) proposes a measure of economy that ranks derivations. The metric is formulated in terms of formal operations and does not directly address the superficial properties of the languages produced. From the perspective of the learner it is the superficial properties that are most salient (or at least, for us, putting ourselves in the position of the learner). One cannot rule out the possibility that there is a relationship between derivational economy and superficial properties of the strings to be processed by the learner, but nothing springs to mind. See Jackendoff 1997 for discussion of the fact that derivation itself is far from being a necessary component of a descriptively adequate account of human language, as well as a vast amount of research in nonderivational theories, especially HPSG.[5]
– Parsing theory may offer some insight into what goes into the complexity of a particular string, in terms of the extent to which the structure corresponding to the string is transparently determined by the string.
– Learnability theory has also been concerned with complexity, not so much the complexity of individual examples as the complexity of a system of examples with respect to the grammar that accounts for their properties.

3. Markedness and computational complexity

3.1 OT

OT posits that knowledge of language can be expressed in terms of the ordering of constraints. The well-formed expressions of a language are those that optimally satisfy the constraints. In principle there may be more than one way

in which an expression can satisfy the constraints; the ranking of the constraints relative to one another determines which of these is optimal.

Let us take a familiar artificial example. Suppose that there is one constraint to the effect that some category α must appear in clause initial position, call it "Move", and another constraint that says that categories do not appear in other than their canonical position, call it "Stay". We may have two rankings of these two constraints:

(2) Stay > Move

(3) Move > Stay

Consider a string of the form in (4).

(4) α_i [… t_i …]

This string is optimal with respect to (3), but not with respect to (2). The tableaux in (5) illustrate.

(5) a.

String:	Stay	Move
$_{\alpha i}$[… t_i …]	*!	
[…α_i …]		*

b.

String:	Move	Stay
$_{\alpha i}$[… t_i …]		*
[…α_i …]	*!	

In (5a) the movement string is ill-formed with respect to the more highly ranked constraint, Stay, while the non-movement string is well-formed with respect to this constraint. The reverse situation holds in (5b). Thus we have grammars for two languages, one of which requires movement, and the other of which disallows it. The only difference between the two grammars in this case is the relative ordering of the constraints. This is the device for representing language variation in OT.

An account of this type raises two fundamental questions. First, what determines the set of possible constraints? Second, if some orderings of constraints are preferred to others, why is this the case? Beyond this there are difficult questions of computability and learnability (Tesar 1995).

In OT the set of possible constraints is determined by Universal Grammar. This much is not controversial, since any theory of grammar must provide some account of what the possibilities are that languages may choose among.[6] The critical question has to do with the rankings. In some cases there appears to be a natural ordering of the constraints, but there is nothing in the theory *per se* that rules out any particular orderings. If we find that there is a preferred ordering, this ordering of the constraints is an accounting of or an embodiment of the markedness relations, in some sense. But of course, in addition to representing markedness, we would like to be able to explain where it comes from.

Bresnan (2000) characterizes markedness in syntax in terms of the correspondence between representations, in particular, c-structure and f-structure: "there is not a perfect correspondence between the categorial (c-structure) head and the functional (f-structure) head". We believe that the notion of correspondence in general is the right one for the purpose of characterizing optimality; let us go back to the most primitive correspondence, however, that between sound and meaning, in order to find an explanation for markedness relations. If, as we suggest in the next section, markedness in the end corresponds to the complexity of mapping between strings and conceptual structures, an OT account, to the extent that it correctly captures the markedness relations, is parasitic on the underlying correspondence that is ultimately responsible for complexity.

3.2 The basis for markedness

3.2.1 *The Derivational Theory of Complexity*

We take it as given that the job of the grammar that the learner constructs or acquires is to map strings of words into conceptual structures and vice versa. This mapping is not one-to-one. A word or string of words may correspond simultaneously to several disjoint parts of the CS, and one part of the CS may correspond to several disjoint substrings. The hierarchical structure of CS does not correspond in a straightforward way to the ordering of the string. In the early days of generative grammar, transformations of phrase markers representing or corresponding to aspects of meaning, especially argument structure, was a device for capturing some of these mismatches. Given some canonical Deep Structure representation, the complexity of the mapping could be measured roughly by the number of operations required to get the string from the Deep Structure.[7] This was called the Derivational Theory of Complexity,[8] and was thoroughly repudiated by the end of the 1970s. Bresnan (2000) argues against an updated version as it appears in the OT syntax of Grimshaw (1997), formulated in terms

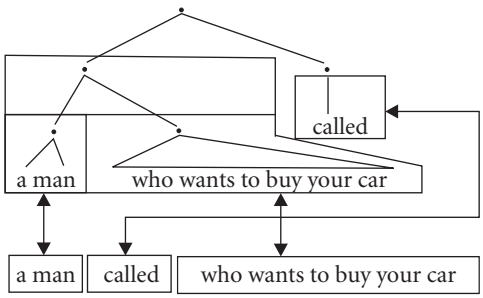
of movements of heads to functional categories and of phrases to Spec.

The problem with the DTC was that it calculated complexity on the basis of the number of transformational operations, and many of these operations were simply formal housekeeping devices required by the transformational theory of the time, such as Affix Hopping. While the number of such housekeeping devices might differ from sentence to sentence, there was no evidence that they contributed at all to relative processing complexity. But the DTC contains a core of insight. The important transformational operations that contribute to complexity are those that deform the canonical Deep Structure so that contiguous portions of the string do not correspond to contiguous portions of the Surface Structure. These correspondences constitute mismatches that the language learner and the language processor have to figure out.[9] To take a simple example, consider extraposition of relative clauses.

(6) A man called who wants to buy your car.

The interpretation of this example is 'a man who wants to buy your car called', but the relative clause and the head that it modifies are not adjacent in the string. Hence there is a mismatch between the hierarchical structure and the string, illustrated in (7).[10]

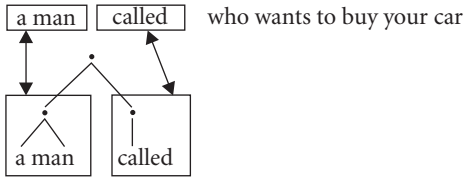
(7)



The crossing of mapping lines and the breaking up of the structure of the subject illustrates the mismatch. (The crossing has nothing to do with linear ordering in the structure, but with the way we display the hierarchical organization and how it maps into the string.)

Intuitively, discontinuity of the sort illustrated in (7) does not contribute significantly to processing complexity. If this intuition is correct, it would suggest that discontinuity in itself is not problematic.[11] Rather, complexity arises when there are factors that interfere with the resolution of the discontinuity. In the case of extraposition, on the assumption that extraposition is not

inherently complex, this may well be because it is treated as a special case of binding, along the lines suggested by Culicover and Rochemont (1990). The core idea, in this case, is that processing of the linear order of words produces a structure of the form in (8) at the point at which the extraposed constituent is encountered.

(8) 

Processing of the relative clause creates a predicate that must be applied to the representation of an object in CS; in this case the only available antecedent is the CS representation of *a man*. Mapping (8) into (7) depends on the extent to which this antecedent is computationally accessible. It is this accessibility that we believe underlies the complexity of the mapping between strings and CS, both for learners and for adult language processors, especially in the case of discontinuity but in other cases as well.[12]

This takes us close to a familiar idea in the domain of human sentence processing. Constituents that have been processed and interpreted are in general accessible to subsequent operations that require retrieval of their meanings (Bransford and Franks 1971); at the same time, the actual form of these constituents is difficult to retrieve as sentence processing continues.[13] One of the key ideas in this work is that local relations are easier to compute than more distant relations, which require memory for the elements that occur earlier. Memory may degrade with time or it may be overloaded by the need to perform multiple tasks; or it may be disordered by the need to perform multiple similar tasks. All of these are logically possible and empirical evidence exists to suggest that they are in fact realistic problems for a language processor. Again, we suggest that the language learner faces similar problems. The bottom line is, other things being equal, distance in the string between elements that are functionally related to one another in the interpretation of the string contributes to complexity of mapping that string into CS.

A further contributor to complexity of the mapping is that CS is not the only complex hierarchical structure that is mapped onto the string. There is also discourse structure, which we take here to be the representation of topic and focus. To some extent, which varies from language to language, these aspects of the discourse structure are expressed in terms of word order. In English, for

example, a topic may be identified through extraction to sentence-initial position (Prince 1987). Focus in certain languages is marked by extraction to a left peripheral position (as argued in a number of papers in Kiss 1992). The possibility that such relations are marked in a given language introduces an additional component of complexity to the mapping between the string and its interpretation.[14]

A measure of complexity that intuitively falls under this idea of complexity concerns the extent to which the order of words in a sentence corresponds uniformly to its branching structure. Hawkins (1994) has argued for the view that "words and constituents occur in the orders they do so that syntactic groupings and their immediate constituents can be recognized (and produced) as rapidly and efficiently as possible in language performance." Hawkins shows that different constituent orders require different sized spans of a string and corresponding phrase structure in order to determine what the immediate constituents are. The differences "appear to correspond to differences in processing load, therefore, involving the size of working memory and the number of computations performed simultaneously on elements within this working memory."[15]

The contribution of distance is not restricted to overt movement. In the case of so-called 'LF' movements, where an operator has scope over a region of a sentence, there is a measurable distance between the operator and the boundaries of what it takes scope over.

The direction that these observations point to is that one key to complexity, in the sense of language acquisition at least, and its impact on language change, is not formal syntactic complexity in the sense of the derivation of the phrase marker. Rather, it is the complexity of the syntactic construction as a way of conveying the corresponding conceptual structure. The construction may be *sui generis*, as is suggested by the example of Culicover and Jackendoff (1999) of *the more X the more Y*, or it may be the product of the interaction of a set of structural devices, such as fronting, scrambling, head movement, and so on.

3.2.2 *Learnability theory*

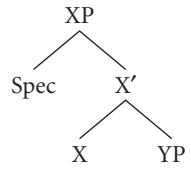
These two types of complexity, derivational complexity and processing complexity, take us to learnability. The basic problem of the complexity of the mapping between string and CS was addressed formally in Wexler and Culicover (1980).[16] There the sole criterion was the learnability of a class of grammars. A class of grammars is not learnable in a particular sense if it is possible for a learner to construct a grammar in which there is an error that can never be

corrected by subsequent experience, in principle. Errors that can be corrected on the basis of experience are called "detectable" errors; the proof of learnability involves demonstrating that there are no undetectable grammatical errors, given certain assumptions about the possible grammatical operations that may be hypothesized by the learner.

The identifiability of errors is an appropriate consideration in an account of learning that posits random construction or random selection of rules. In such a theory, the correctness of a particular hypothesis is determined by whether it produces errors. If we shift our perspective to a constructive account, then we shift our emphasis from the identification of grammatical errors to the relative complexity of the mapping.[17] If a mapping is relatively opaque then the ability of the learner to compute the mapping is severely limited. On this perspective, the most transparent mapping is one in which the string contains unambiguous, independent, and complete evidence about what the corresponding CS representation is.

We have already illustrated a mapping that involves a certain amount of complexity, in (7). Let us compare this with the type of situation envisaged in Kayne's Antisymmetry theory, where all branching is to the right, such that all phrases are of the form given in (9).[18]

(9)



Kayne assumes that there is a strict correlation between asymmetric c-command and linear order called the Linear Correspondence Axiom (LCA), such that if α c-commands β and β does not c-command α, then α precedes β. If there is no movement, and if the branching structure in (9) is taken to be the CS, then the mapping between strings and corresponding CS representations will be straightforward, in fact. All of the mappings will conform to the LCA. Moreover, the mapping will be maximally simple, in that in order to construct the mapping it is sufficient to scan the string from left to right, establishing a correspondence between each element in the string and each constituent of the CS.

4. The computation of complexity

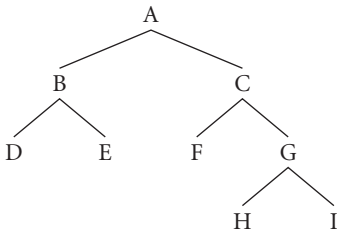
4.1 Distance

We have argued to this point that the distance between functionally related parts of a string is the crucial component of complexity, because of memory limitations. Here we formulate a rough measure of this distance. The essential idea is that in the simple case the string is an image of the CS representation, to a first approximation, and relative distance in the two domains should be relatively consistent. When it isn't, there is 'twisting' of the structure so that it can map into the string. The greater the twisting, the greater the complexity.

Let us begin with a CS representation. For convenience, will assume that the CS representation is a structure in which the terminals correspond to the individual words and functional heads of a string; in essence, it is like a D-structure in the classical sense. Using such a structure instead of a true CS along the lines of Jackendoff (1990) representation allows for substantial simplification. It allows us to develop a foundation for the intuition that uniform branching is optimal, which in turn allows us to view the objectives of Kayne's antisymmetry theory in terms of markedness in contrast to rigid constraints on structure.

In the representations that follow we take the capital letters to correspond to the types in the CS hierarchy; the terminals are basic concepts.

(10)



string = *defhi*

Let us say that the Image of D is *d*, and so on for the other terminals in the CS representation. We simplify dramatically here, because it is plausible that a single CS can be expressed in a number of different ways. We can also define an inverse relation and since there is more information in the tree than in the string, the inverse image defines a set containing one or more CS representations.

(11) $\text{Image}(\text{D}) = d$
$\text{Image}^{-1}(d) = \langle \text{D}, \text{D}', \ldots \rangle$

Hence the correspondences are many-to-many.

It is possible that the image of a higher level node in the tree is not decomposable into the image of its constituents, which would be typical of an idiom (e.g. Image^{-1}(*kick the bucket*) = ⟨DIE,…⟩). It is also possible that a single element in a string corresponds to a complex CS representation, as argued for example by Jackendoff (1990). And it is possible that there is a particular aspect of CS that corresponds to a class of strings that satisfy a certain structural description, as has been argued for the dative construction among others (see Goldberg 1995, Jackendoff 1997). We leave these more complex possibilities aside here.

We can measure the distance between constituents of the CS representation in terms of the height of the common ancestor. For sisters we will say that the CDistance, that is, the distance in the CS representation, is 0, which is the number of ancestors that they do not have in common. So for (10) we have

(12) CDistance(H,I) = 0

The CDistance between a node and the daughter of its sister is 1, as in the case of (F,H) and (F,I). In general, the CDistance between two nodes is the number of dominating nodes that the path between them passes through. A node is not a dominating node if the path through it links sisters; otherwise it is.

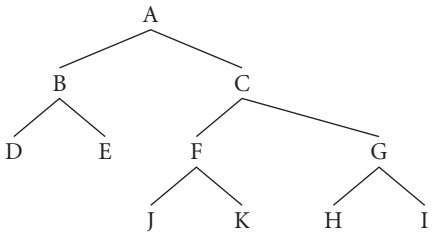
Given this notion of CDistance, we can relate the distance between substrings to linear relations between the corresponding parts of the CS representation. The general idea is the following. For a given distance between two elements (words, phrases, etc.) in the string, we posit that greater distance in CS requires greater processing, and hence produces greater complexity, other things being equal.

Consider the string *delhi*. Sisterhood at CS, that is, CDistance = 0, corresponds to adjacency in the string. If CDistance(α,β) = 0, and Image(α) precedes Image(β), then the right edge of Image(α) is adjacent to the left edge of Image(β). This is the case, for example, for α = B and β = C.

We use this property to measure the amount of deformation (or 'twisting') of a CS representation with respect to its corresponding string. In the case of adjacency there is no deformation. We may measure deformation in terms of the distance in the string between the right edge of Image(α) and the left edge of Image(β), which in this case is 0. But we must be careful to correlate these distances appropriately. So, for example, the distance between B and G is 1. Image(B) = *de* and Image(G) = *hi*. The distance between the right edge of *de* and the left edge of *hi* is one element, namely *f*, but this is simply because *f* is a terminal.

Suppose we replace F corresponding to *f* in the string in (10) with [$_{\text{F}}$ J K], corresponding to *jk* in the string.

(13)



string = *dejkhi*

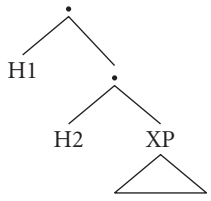
Now there are two elements in Image(F). But the distance between *de* and *jk* and is 1, if we treat Image(B) = *de* and Image(F) = *jk* as single units. They can be so treated because they correspond to constituents of CS. Let us call this distance between substrings that correspond to constituents the Parse Distance, or PDistance.

(14) Given a string s, containing initial substring a and final substring b such that Image(**a**) = a and Image(**b**) = b, PDistance(a,b) is the minimal number of strings $x_1, \ldots, x_n$ such that $s = a + x_1 + \ldots x_n + b$

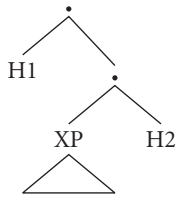
If a and b are adjacent then PDistance = 0. In (10), PDistance(Image(B), Image(G)) = 1. PDistance(e,i) = 2, and PDistance(d,i) = 3.

Consider now the most basic relation, that of head-complement. Hawkins' intuition that heads are optimally adjacent to the heads of their complements correlates in a natural way with the relative distance measures. For simplicity of exposition, let us identify Image(x) and x. We can then encode both CS and the string in a traditional ordered phrase marker, as shown in (15).

(15) a.



b.



We observe that in (15a),

(16) CDistance(H2,XP) = 0 PDistance(H2,XP) = 0
CDistance(H1,H2) = 1 PDistance(H1,H2) = 0
CDistance(H1,XP) = 1 PDistance(H1,XP) = 1

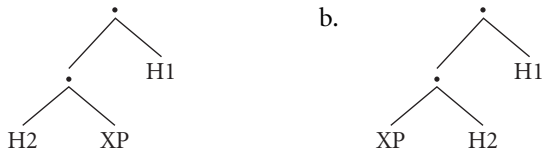
and in (15b),

(17) CDistance(H2,XP) = 0 PDistance(H2,XP) = 0
CDistance(H1,H2) = 1 PDistance(H1,H2) = 1
CDistance(H1,XP) = 1 PDistance(H1,XP) = 0

We have highlighted with underlining where the difference between the two cases lies. A twisting of the hierarchical structure is reflected by an increase or decrease in PDistance and constant CDistance. Such a relation occurs when a head and the heads of its complement are separated in the string; this requires that the head that occurs first be held in memory along with the lower material until the lower head comes along. The more complex structure is the one for which the PDistance between two heads is greater, while the CDistance is the same.

To see whether this is an accidental property of the particular configuration, let us see what happens when we have a uniform left branching structure.

(18) a. b.



For (18a),

(19) CDistance(H2,XP) = 0 PDistance(H2,XP) = 0
CDistance(H1,H2) = 1 PDistance(H1,H2) = 1
CDistance(H1,XP) = 1 PDistance(H1,XP) = 0

and for (18b),

(20) CDistance(H2,XP) = 0 PDistance(H2,XP) = 0
CDistance(H1,H2) = 1 PDistance(H1,H2) = 0
CDistance(H1,XP) = 1 PDistance(H1,XP) = 1

Again, the greater PDistance between heads that are adjacent in the structure occurs when the branching is not uniform, as in (18a).

The total deformation of a tree of course grows as the number of heads grows, and the extent to which they do not line up grows. So, if we take the pattern in (18a) and replicate it, the total PDistance between adjacent heads will equal the number of alternating pairs of heads, while the total CDistance between adjacent heads will remain 0. So we might surmise that a single head in an initial position with all other heads to the right might not be that costly in terms of complexity, and might optimize something else in the grammar. The computational cost would be minimized if the head in question was the highest,

since an internal 'outlier' would produce a cost with respect to the head immediately above it and the one immediately below it.

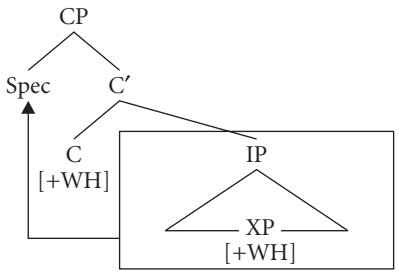
On this view, complexity of processing is correlated with memory load, and uniformity of branching reduces memory load. In this sense, the antisymmetry approach of Kayne (1994) is correct in placing a high value on uniformity of the direction of branching structure, but is too strong in that it does not allow for nonuniform branching at all. For our purposes, it is enough to say that uniformity is computationally less complex, other things being equal. The reduction of complexity, coupled with a theory of language change that reflects the computation biases of learners as discussed in §2, will produce a situation in which uniformity of branching is a very strong tendency without being an absolute universal, a result that appears to be correct (again, see Hawkins 1994).

4.2 Stretching and twisting

The measure of complexity in terms of distance is a crude one, but it is worth seeing whether it extends naturally to other phenomena. We have already discussed extraposition, and have argued that it is not inherently complex as long as the antecedent of the extraposed predict is accessible. It is well-known that extraposition is more difficult to process when there is an intervening potential antecedent (Ross 1967), a relation that is easily formulated in terms of relative PDistance.

Another phenomenon of some interest is that *wh*-movement and related constructions have been observed to be strictly leftward, not rightward. Kayne derives this result by postulating uniform rightward branching, so that the possible landing sites will always be to the left. Left branching languages typically lack such leftward movements, which Kayne explains by deriving the left branching structure from leftward movements that block other leftward movements. For example, movement of IP to SpecCP puts I in final position, and blocks subsequent movements to SpecCP.

(21)

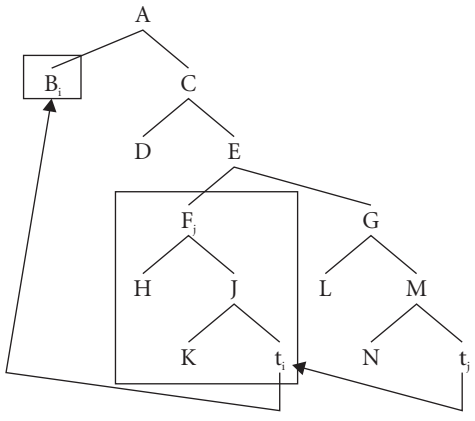


As we have already seen, a mirror image of a structure preserves all of the distance relations, so that it will not be possible to derive the absence of rightward movement from distance considerations alone. It is not implausible that operators that bind variables need to be processed before the variables that they bind, so that the variables may be identified as such.[19] Such functional considerations entail that movement of operators will be to positions where they precede the variables that they bind, not to the right.

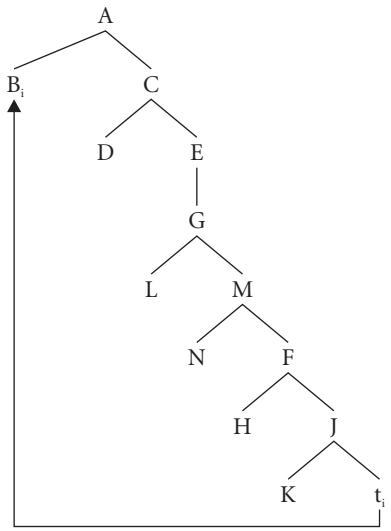
This does not tell us, however, why there is no leftward movement for purposes of marking scope in most if not all strictly head-final languages. One possible answer is that in head-final languages, the only possible movement for the operator would be to the head that defines its scope (typically the inflected verb, or something adjoined to the verb, such as a complementizer or a particle). In a head-final language this verbal head is on a right branch, of course. So the operator would have to move to the right, which is ruled out on the sorts of functional grounds we have just discussed. Note that there are head-final languages in which covert and overt markers are licensed to the right. In Korean, for example, the relative clause ends in a relative marker, although, strikingly, there is no overt movement of a relative pronoun.

Let us consider, finally, the cost of extracting from a moved constituent. (22) illustrates.

(22) a.



b.



Intuitions about complexity suggest that extraction from an extracted constituent is more problematic than extraction from an unmoved constituent. The first empirical evidence pointing this out is due to Postal (1972), who used it as an argument against successive cyclic movement in the Conditions framework of Chomsky (1973).

(23) a. Leslie believes that [a picture of Terry]$_i$, you would never find t_i in a shop like that.
b. *Terry is the person who$_j$ Leslie believes that [a picture of t_j], you would never find t_i in a shop like that.

Examples of the following sort are cited by Wexler and Culicover (1980) as evidence for the Freezing Principle, which blocks extraction from a moved constituent.

(24) a. Who$_i$ did you tell Mary [a story about t_i]?
b. *Who$_i$ did you tell t_j to Mary [a story about t_i]$_j$?

The Freezing Principle was motivated by considerations of learnability.

At the same time, we may take the view that extractions such as these are grammatical but marginal. This more closely fits our current perspective, which is that extreme deformation produces complexity but not necessarily complete ungrammaticality. Examples such as (24b) are judged by some speakers to be grammatical, and examples such as the following are not completely impossible.

(25) ?Terry is the person [of whom]$_j$ Leslie pointed out that [such pictures t_j]$_i$ you would never find t_i in a shop like that.

The intuition that we wish to develop about extraction, then, is that a simple movement to an accessible position is in effect a 'stretching' of the CS representation onto a particular linear order. Constituents that are close in CS are more distant syntactically, but the topological relations are not significantly distorted — the PDistance between a moved constituent and its trace is correlated with the CDistance. Presumably there is some falling off when these distances become large, but the intervening material is not problematic. However, when we extract from an extracted constituent, there is a 'twisting' of the structure in order to map it into the string. Attachment of B_i in (22a) is actually closer in PDistance and CDistance to its trace (shown in (26a)) than it is in (22b) (shown in (26b)) yet the complexity of this attachment is greater.

(26) a. PDistance(B_i,t_i) = 3
CDistance(B_i,t_i) = 4
PDistance(F_j,t_j) = 2
CDistance(F_j,t_j) = 2
b. PDistance(B_i,t_i) = 5
CDistance(B_i,t_i) = 5

When the trace is contained in a moved constituent, the complexity would be better represented by constructing a measure that takes this fact explicitly into account. One possibility is to multiply the CDistance from B_i to its trace times the CDistance from F_j to its trace in (22a), which yields 8 compared with 5 in (22b). Such a measure, while arbitrary, reflects the degree of deformation of the tree.

To sum up, there are essentially three ways to map a CS into a string. One is to align the constituents of the CS with the string without crossing constituents of the parse string. The second is to stretch a CS constituent to position the corresponding string in a position where it is not adjacent to its CS sisters. The third is to twist the lines so that the correspondences between strings and constituents of CS cross. Our intention is that the relative complexity accorded to this measure reflects the relative complexity in terms of memory requirements, and that we do not have to formulate an explicit theory of memory for sentence processing in order to be able to capture the basic outlines of comparative complexity.

Note that there are several complexities that we have not factored into our account here. A string of words may map into a CS representation so that there are fewer primitives in the CS representation than there are words in the string;

this is a characterization of idiomaticity. Or there may be more primitives in the CS representation than in the string; this is a characterization of a 'construction' in the sense of Construction Grammar. In both cases there is the opportunity for a mismatch in the CDistance and PDistance, since the two are equal when there is a uniform linearization of a branching structure, with a one-to-one correspondence between elements of the string and elements of the CS representation. To the extent that this additional complexity presents a burden for the learner, we might expect some effect on learning. But there is no twisting and so the burden, if it exists, is relatively light.

5. Summary

We have suggested that at its core the antisymmetry theory reflects the relative computational simplicity of mapping strings into structures assuming uniform branching. The branching really has to do with the relative linear order in the string between related heads and their identifiability, a measure that can be correlated with memory but that can be abstracted formulated for string/structure mappings. A computational bias for certain constructions will produce a clustering of certain structural features in languages, given a plausible theory of language change that ties up with a theory of language acquisition. Hence we expect to find, and in fact do find, that languages tend towards uniform branching. At the same time, greater complexity does not entail nonexistence, and deviations from the optimal are possible and attested, yielding variation among languages. Taking the perspective of markedness allows us to accommodate these deviations without taking the radical step advocated by Kayne (1994), that of allowing only uniform rightward binary branching, and accounting for all apparent counterexamples in derivational terms.

Notes

* We are indebted to Greg Carlson for his comments on an earlier version of this paper. His suggestions have led to numerous substantial improvements. Naturally, we are responsible for any remaining deficiencies.

1. This notion of construction is related to that of Construction Grammar (see Goldberg 1995 for example), in that we assume, with Jackendoff 1990, that grammatical knowledge consists of syntax-semantics correpsondences.

2. Latané (1996), Nowak et al. (1990). Nettle (1999) independently hit upon the idea of using the Latané/Nowak approach to using Social Impact theory in a computational simulation of language change.

3. In fact this must be true in a trivial sense; see Culicover 1999 for discussion.

4. This is demonstrated in the simulation Sitsim, by Latané, Nowak and Szamrej. Kirby (1994) notes the role of bias in change, while Briscoe (2000) has constructed computational simulations of the evolution of language in which biases play a major role.

5. The recent exchange in *NLLT* regarding the MP does not offer any particularly good motivation for derivational economy, in our view, but below we suggest an incompatible alternative view of derivational complexity that might be more satisfying.

6. Matters become somewhat more complex if we attempt to derive some of the constraints from functional considerations, rather than simply assume that they are all part of UG. For discussion, see Newmeyer to appear and Aissen and Bresnan to appear.

7. Deep Structure was renamed D-structure in subsequent syntactic theory.

8. Brown and Hanlon (1970); Fodor, et al. (1974).

9. For more on mismatches, see Culicover and Jackendoff (1995), Culicover and Jackendoff (1997) Culicover and Jackendoff (1999), among many others.

10. There are several familiar mechanisms for representing discontinuity in natural language, including movement and passing features of some gap within the larger string, so that the entire string inherits the ability to license the 'moved' constituent. The formal devices for capturing this type of relationship are not at issue here. The main point is that the mismatch introduces a level of complexity into the mapping, both from the perspective of computing it for a given string, and from the perspective of determining its precise characteristics on the basis of pairs consisting of string and corresponding CS.

11. It is often suggested that extraposition and other rightward movements improve processing by reducing center-embedding. See Hawkins (1994) and Wasow (1997).

12. Hence we follow the lead of Berwick (1987), who saw the connection very clearly.

13. There are many additional complexities, of course. See Kluender (1998) for a discussion of some of these.

14. Of course, we could suppose that CS includes a representation for discourse structure as well as a representation for argument structure, but this would not simplify the mapping problem, since we would then be dealing with a more complex CS with more possibilities.

15. One minor concern with the explanatory force of this argument is that we might have expected that human memory would have evolved so as to overcome the problems offered by non-uniform branching. Of course there are many reasons why this would not have happened and it is probably impossible to settle the issue. Shifting the burden of explanation to language acquisition rather than language processing sidesteps this problem, since we

probably do not want to attribute to early learners the adult's capacity to store and process long strings of linguistic material. See §3.2.

16. The mapping was formulated in terms of strings and base phrase markers, but the general problem is the same as the one that we are considering here.

17. This is not to say that grammatical errors per se are irrelevant, but simply that they are not the whole story. On the current perspective, a grammatical error would occur if a particular string is hypothesized to correspond to the wrong conceptual structure representation. We assume that such errors are always detectable on the basis of subsequent information in the form of ⟨string,CS⟩ pairs, but leave open the possibility that a particular formulation of the correspondences might give rise to pathological cases that would have to be addressed.

18. In principle all branching could be to the left in Kayne's approach, but Kayne introduces an additional stipulation that rules out leftward branching.

19. An absolute requirement along these lines is too strong, given that there are cases where an operator binds a variable to its left, such as *If he$_i$ wants to, each man$_i$ can vote* (G. Carlson, p.c.). We hypothesize that the correct account is one that assigns a strong preference to the case in which the operator precedes what it binds, presumably for processing reasons.

References

Aissen, Judith and Joan Bresnan. To appear. "Reply to Newmeyer". *Natural Language and Linguistic Theory*.

Berwick, Robert C. 1987. "Parsability and Learnability". *Mechanisms of Language Acquisition,* ed. by Brian MacWhinney. Hillsdale, N. J.: Lawrence Erlbaum Associates.

Bransford, John D. and Jeffrey J. Franks. 1971. "The Abstraction of Linguistic Ideas". *The European Journal of Cognitive Psychology* 2.331–350.

Joan Bresnan. 2000. "Optimal Syntax." *Optimality Theory: Phonology, Syntax and Acquisition*, ed by Joost Dekkers, Frank van der Leeuw and Jeroen van de Weijer, 334–385. Oxford: Oxford University Press.

Briscoe, Ted. 2000. "Grammatical Acquisition: Inductive Bias and Coevolution of Language and the Language Acquisition Device". *Language* 76. 245–296.

Brown, Roger and C. Hanlon. 1970. "Derivational Complexity and the Order of Acquisition in Child Speech". *Cognition and the Development of Language*, ed. by John R. Hayes, New York: Wiley.

Chomsky, Noam. 1965. *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

Chomsky, Noam. 1973. "Conditions on Transformations". *Festschrift for Morris Halle*, ed. by Stephen R. Anderson and Paul Kiparsky. 232–286. New York: Holt, Rinehart and Winston.

Chomsky, Noam. 1994. "Bare Phrase Structure". *MITWPL Occasional Papers*, Cambridge, MA: MIT.

Culicover, Peter W. 1999. *Syntactic Nuts*, Oxford University Press, Oxford.

Culicover, Peter W. and Ray Jackendoff. 1995. "*Something Else* for the Binding Theory". *Linguistic Inquiry* 26.249–275.

Culicover, Peter W. and Ray Jackendoff. 1997. "Syntactic Coordination Despite Semantic Subordination". *Linguistic Inquiry* 28.195–217.

Culicover, Peter W. and Ray Jackendoff. 1999. "The View from the Periphery: The English Comparative Correlative". *Linguistic Inquiry* 30.543–571.

Culicover, Peter W. and Michael S. Rochemont. 1990. "Extraposition and the Complement Principle". *Linguistic Inquiry* 21.23–48.

Fodor, Jerry A., Thomas Bever and Merrill Garrett. 1974. *The Psychology of Language: An Introduction to Psycholinguistics and Generative Grammar*. New York: McGraw-Hill.

Goldberg, Adele E. 1995. *Constructions: A Construction Grammar Approach to Argument Structure*. Chicago: University of Chicago Press.

Greenberg, Joseph. 1963. *Universals of Language*. Cambridge, MA: MIT Press.

Grimshaw, Jane. 1997. "Projections, Heads and Optimality". *Linguistic Inquiry* 28.373–422.

Hawkins, John A. 1994. *A Performance Theory of Order and Constituency*, Cambridge: Cambridge University Press.

Jackendoff, Ray. 1990. *Semantic Structures*. Cambridge, MA: MIT Press.

Jackendoff, Ray. 1997. *The Architecture of the Language Faculty*. Cambridge, MA: MIT Press.

Kayne, Richard S. 1994. *The Antisymmetry of Syntax*. Cambridge, MA: MIT Press.

Kirby, Simon. 1994. "Adaptive Explanations for Language Universals". *Sprachtypologie und Universalienforschung* 47.186–210.

Kiss, Katalin E. 1992. (ed.) *Discourse Configurationality*. Oxford: Oxford University Press.

Kluender, Robert. 1998. "On the Distinction Between Strong and Weak Islands: A Processing Perspective". *The Limits of Syntax*, ed. by Peter W. Culicover and Louise McNally. 241–279. New York: Academic Press.

Latané, Bibb. 1996. "The Emergence of Clustering and Correlation from Social Interactions". *Modelle Socialer Dynamiken: Ordnung, Chaos und Komplexita,* ed. by Rainer Hegselmann and Heinz-Otto Peitgen. Wien: Holder-Pichler-Tempsky, Mt.

Nettle, Daniel. 1999. "Using Social Impact Theory to Simulate Language Change". *Lingua* 108.95–117.

Newmeyer, Frederick. To appear. "Against Functional Optimality Theory". *Natural Language and Linguistic Theory*.

Nowak, Andrzej, Jacek Szamrej and Bibb Latané. 1990. "From Private Attitude to Public Opinion: A Dynamic Theory of Social Impact". *Psychological Review* 97.362–376.

Postal, Paul M. 1972. "On Some Rules That Are Not Successive Cyclic". *Linguistic Inquiry* 3.211–222.

Prince, Ellen F. 1987. "Topicalization and Left-Dislocation: A Functional Analysis." *Discourses in Reading and Linguistics.*, ed. by Sheila J. White, and Virginia Teller, 213–225. (Annals of the New York Academy of Sciences 433:). New York: New York Acad. of Sciences.

Ross, John Robert. 1967. *Constraints on Variables in Syntax*. Unpublished doctoral dissertation. Cambridge, MA.: MIT

Tesar, Bruce. 1995. "Computational Optimality Theory". unpublished doctoral dissertation. Boulder, CO.: University of Colorado.

Wexler, Kenneth and Peter W. Culicover. 1980. *Formal Principles of Language Acquisition*, Cambridge, MA.: MIT Press.