# Compression and Adaptation

Tracy Teal[1], Daniel Albro[2], Edward Stabler[2], and Charles E. Taylor[1]

[1] Department of Organismic Biology, Ecology and Evolution, Box 951606, University of California, Los Angeles, CA 90095, USA {tracyt,taylor}@biology.ucla.edu
[2] Department of Linguistics, Box 951543, University of California, Los Angeles, CA 90095, USA albro@humnet.ucla.edu, stabler@ucla.edu

**Abstract.** What permits some systems to evolve and adapt more effectively than others? Gell-Mann [3] has stressed the importance of "compression" for adaptive complex systems. Information about the environment is not simply recorded as a look-up table, but is rather compressed in a *theory* or *schema*. Several conjectures are proposed: (I) compression aids in generalization; (II) compression occurs more easily in a "smooth", as opposed to a "rugged", string space; and (III) constraints from compression make it likely that natural languages evolve towards smooth string spaces. We have been examining the role of such compression for learning and evolution of formal languages by artificial agents. Our system does seem to conform generally to these expectations, but the trade-offs between compression and the errors that sometimes accompany it need careful consideration.

## 1 Introduction

Why are some systems more adaptable than others? A core feature of nearly all successful adaptive systems is the ability to distill experience into schemas, models or theories and then employ those abstracted structures in new circumstances. Information about the environment is not simply recorded as a look-up table, with generalization to new situations happening only at look-up time. Rather, it is plausible that salient features of situations are noted, and then departures from expectations are noted. This is the essence of compression.

Gell-Mann [3] has argued that a compressed form "is usually approximate, sometimes wrong, but it may be adaptive if it can make useful predictions including interpolation and extrapolation and sometimes generalize to situations very different from those previously encountered. In the presence of new information from the environment, the compressed schema unfolds to give predictions or behavior or both." We would like to know more about the role of compression in adaptation. For example, can we identify features of compression that make some forms more or less likely to be successful? What sorts of compression are best? How much is desirable?

This perspective on learning is not new, especially for cultural evolution. It is evident, for example, that there is frequently a practical necessity to simplify things so that we can understand them. Hence the ubiquitous use of simplified models in science. Which models are themselves best, is also a matter of simplicity. William of Ockham, among others, has observed that "it is futile to do with more what can be done with less" [18]. In another domain, Chomsky [2] has placed the desirability of compressed, "minimal" grammar at the heart of his theory of human language. Nonetheless, there have been few attempts to explore, in a systematic way, how compression alone may affect adaptation.

In this paper we first state some heuristics that we conjecture to be generally, if not always, true. We then introduce some elementary definitions and principles about data compression and formal languages. We have chosen to work with formal languages because it is possible to discuss them concretely and because such systems quite clearly show changes in their ability to generalize [14, 6, 11, 10]. We next discuss how compression can occur as an agent learns a language by hearing examples of it. We also describe some experiments where agents learn the languages with compressed grammars, and where the languages evolve as a result. Finally, we discuss some features of compression and evolution in our system in the light of our conjectures. We emphasize that while our discussion will be directed mostly to cultural evolution - scientific theory and language, we believe these heuristics apply to other adaptive complex systems, such as organic evolution, immune systems, and neural networks.

## 2 Conjectures about compression and adaptation

*Conjecture 1.* **Compression aids in generalization.**

From a series of observations like "Crow A is black" and "Crow B is black" we compress a look-up table of crows and their colors to the generalization that "All crows are black." The generalization is clearly smaller, more "compressed" than a list of many instances. The precise characterization of the circumstances in which such generalization is appropriate, the problem of induction, is a long-standing philosophical problem.

The history of science is a history of finding generalizations that allow a succinct statement of the facts. Following the invention of the spectroscope, the spectral emissions from various elements, including hydrogen, were cataloged. About 1885 J. J. Balmer discovered formulae that would describe the frequencies for hydrogen emissions both compactly and accurately, though they were simply formulas without a model behind them. In 1913, Niels Bohr published a model for the atom that would compactly describe emissions from hydrogen, and several other atoms, in very compressed and desirable form.

While the history of science can be regarded as a series of successively better compressions, it should also be recognized that the resulting compressions may make predictions that are only approximate or even wrong. Although models can give us insight into systems, the actual model used greatly affects the predictions that can be made and the types of behaviors that can be explained. Many scientific theories, no matter how well they might compress a set of observations, are subsequently proven wrong.

*Conjecture 2.* **Compression occurs more easily in a "smooth", as opposed to "rugged", string space.**

Related or connected sets of observations form a better basis for generalization than do similar or unrelated ones. We would like to be concrete about the meaning of "related or connected" in this context. Kauffman [8] has explored the use of adaptive landscapes in a variety of contexts, and we build on his example by exploring a string space of languages, below.

Unrelated observations, like Lord Morton's mare for Darwin's theory of inheritance, can make theory formation quite difficult [13]. As a result, it is thought generally better to focus initial scientific study on simple and well-defined systems, where the smoother space is more easily explored, as Mendel did.

Smoothness and ruggedness are, to some extent, a property of the substitution operators. In molecular evolution for example, adding or deleting tandem repeats of 2 or 4 nucleotides is often easier than adding or deleting a single nucleotide [21].

*Conjecture 3.* **Constraints from compression make it likely that natural languages come to have smooth string spaces.**

Language learners are regarded as systems that aim to identify rule systems that describe the (infinite) language of the community on the basis of finite evidence. This can only be successful in certain circumstances, whether one assumes that success is perfect identification in the limit (the "Gold" paradigm) [4], or that success is feasible convergence to arbitrarily good approximate identification [9, 20]. Since humans clearly learn to speak natural languages that can generate an infinite number of sentences, and do so largely from examples and without conscious awareness of grammatical rules, most linguists believe that, roughly speaking, when a young person is confronted with a new sentence they will "try out" candidate grammars and then, from those grammars that can accommodate it, choose the one that is simplest [1, 5, 19].

Of course, human language learners hear ungrammatical sentences and sentence fragments, but this "noise" apparently does not make communication with the community impossible. We postulate that sufficiently rare complications are typically not incorporated; they will be catalogued as exceptions or simply ignored. As a result, there will be a natural selection for languages that are progressively smoother. This is perhaps analogous to the way that biological systems sometimes "solve" other NP-complete problems. They seem to employ only those parts of the problem space that can be solved simply, even though the general problem might be insoluble [15].

## 3 Data compression

Data can be compressed only when it contains some regularity that can be exploited. [17]. When there is a regularity, listing it repeatedly makes for an eliminable redundancy.

It is important to recognize that some schemes might provide very efficient coding of data, but would involve lengthy and complicated specifications of the decoder. The decoder could, for example, simply contain a list of any finite set of strings to be encoded. A better measure of compression would recognize both costs. The *minimum description length algorithm (MDL)* we use accomplishes this. In the formal language framework the sum of the *grammar-encoding-length* (the cost of the rule set) and the *data-encoding-length* (the cost of the coding of the data) is minimized [12, 16].

## 4 Formal Languages

A (formal) language is a set of strings of symbols drawn from some alphabet. Here we shall be concerned only with *regular* languages, languages that can be accepted by a *finite automaton.* Such automata can be described by a quintuple, $(Q, \Sigma, \delta, q_0, F)$, where $Q$ is a set of states, $\Sigma$ is an input alphabet, $q_0 \in Q$ is the initial state, $\delta \subseteq Q \times \Sigma \times Q$ is a transition function, and $F \subseteq Q$ is the set of final states [7]. An automaton is said to be *deterministic* if the transition $\delta$ is a function $\delta : Q \times \Sigma \rightarrow Q$ with respect to its first two arguments. Associated with each finite state automaton is a *transition diagram*, with an arc connecting states $q_1$ and $q_2$ labeled with vocabulary element $a$ if, and only if, $(q_1, a, q_2) \in \delta$. The automaton accepts a string $s$ if, and only if, the string labels a path from the initial state to a final state.

For purposes of illustration, consider languages `1-s` and `1-r`. Language `1-s` consists of the strings `aaaa, aaab, aaba, abaa` and `abba`. Language `1-r` consists of `aaab, abaa, aaba, abbb,` and `bbab`. Their respective "prefix tree" transition diagrams are shown in Figure 1.

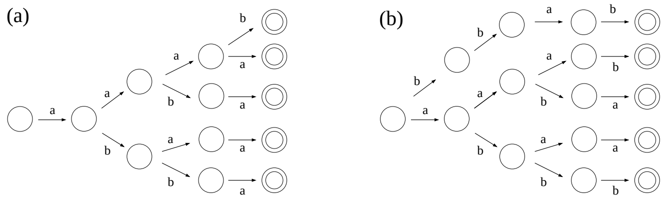

**Fig. 1.** Transition state diagrams for uncompressed grammars: (a) language 1-s (b) language 1-r.

Since each string can be specified by a path through a deterministic automaton, we calculate one simple approximation of the data-encoding-length in bits given by the formula:

$$\sum_{i=1}^{m} \sum_{j=1}^{|s_i|} \log_2 z_{i,j},$$

where $m$ is the number of sentences in the sequence of strings encoded, $|s_i|$ is the length of the $i$'th string $s_i$, and $z_{i,j}$ is the number of ways to leave the state reached on the $j$'th symbol of sentence $s_i$. (A more succinct encoding is obviously possible when the probabilities of the transitions are not uniform. The simple approximation suffices for purposes of this preliminary investigation.)

To specify the automaton itself, we must specify all the triples $(q_1, a, q_2) \in \delta$ and we must also specify the final states, so we calculate the grammar encoding length:

$$|\delta|[2(\log_2 |Q|) + \log_2 |\Sigma|] + |F|[\log_2 |Q|],$$

where $|\delta|$ is the number of triples in $\delta$ and $|F|$ is the number of final states in $F$. The $MDL$ of a language is defined as the sum of its grammar-encoding-length and its data-encoding-length. For language `1-s` the $MDL$ is 119.3 and for `1-r` it is 168.0.

We refer to language `1-s` as "smooth" and language `1-r` as "rugged." The reasons for this are evident from Figure 2, which shows how the strings in each set are related by symbol substitution. (More generally, one might use deletion and duplication operators, as well as symbol-substitutions. Duplication and deletion are known to be important for the evolution of DNA and for natural language, but these operators add considerable complication for some of the comparisons we are interested in, so for this paper we shall restrict our attention to operators that are simple symbol substitutions.)

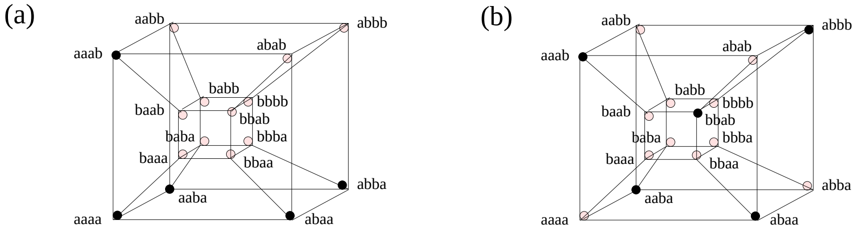

**Fig. 2.** Hypercube representation of the landscape for strings of length 4. Nodes connected by lines are one symbol-substitution away from each other. Black dots are included in the language. Gray dots are not included. (a) Representation of language `1-s` (b) Representation of language `1-r`.

Each string in `1-s` is one symbol-substitution away from its nearest neighbor. The symbol in only one position is changed from one string to the next, and the strings are very similar or regular, while in `1-r` no string is closer than two substitutions away, making the strings not very related. Further, in the hypercube representation, there is a hyperplane in `1-s` that provides regularity which might be utilized for compression, while such regularity is not evident in `1-r`. This occurs in much the same way that the equation of a line offers a compressed representation of a set of data points in linear regression. These two grammars also differ in their MDL measures: the $MDL$ of `1-s` is only 71% that of `1-r`.

## 5 Grammar compression

Our model of language learning will be that of a child, who when listening to an adult speaking a language s/he does not yet understand, tries out different candidate grammars that might accept that language. The minimalist theory supposes that those grammars found suitable are then examined further, and the grammar with the shortest $MDL$ is preferred and tentatively accepted as describing the parent's language.

In our study, an agent is exposed to all the legal sentences in the language – say of `1-s` or `1-r`. It constructs an automaton with associated grammar as shown in Figure 1, above. The agent then compresses the original grammar by attempting to combine states and transitions in such a way that the outcome is still a deterministic automaton and then combines them iff the $MDL$ of the new automaton is smaller than that of the starting one. Given a machine $A = (Q, \Sigma, \delta, q_0, F)$, the result of merging states $q_i, q_j \in Q$ is the machine $A'$ which has these two states replaced by a new state $q_{ij}$ as follows:

$$A' = ((Q - q_i, q_j) \cup \{q_{ij}\}, \Sigma, \delta', q_0', F')$$

where: $q_0' = q_{ij}$ if $q_0$ is either $q_i$ or $q_j$, $F' = (F - q_i, q_j) \cup \{q_{ij}\}$ if either $q_i$ or $q_j \in F$, and $\delta'$ is the result of replacing all instances of both $q_i$ and $q_j$ by $q_{ij}$ in all the triples $(q_n, a, q_m)$ that define $\delta$. This is applied recursively in a hill-climbing manner.

The compressed transition diagrams and hypercubes for languages `1-s` and `1-r` are shown in Figures 3 and 4.

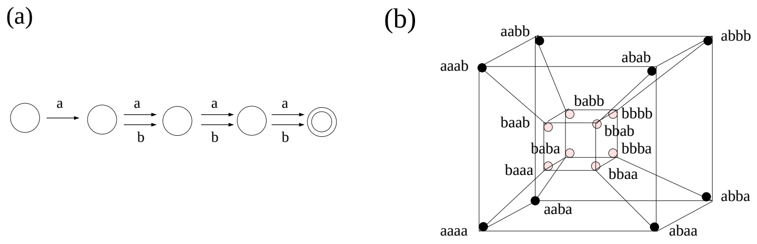

**Fig. 3.** (a)Transition diagram of the compressed grammar for language `1-s`. (b) Hypercube representation of the sentences included in the compressed grammar.

The compressed `1-s` is much smaller than that of the original ($MDL = 56.8$ vs. 119.3), due to changes both in the number of states (11 to 5) and in the number of transitions (12 to 7). All of the accepted strings in this example are of

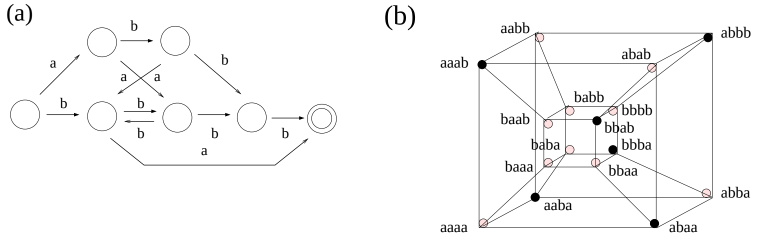

**Fig. 4.** (a)Transition diagram of the compressed grammar for language 1-r. (b) Hypercube representation of the sentences included in the compressed grammar.

the correct length, but the compressed version generalizes to include the entire outer cube of sentences in the hypercube. In "real life" such a generalization may or may not be desirable.

The compressed version of language 1-r is also substantially smaller than the original ($MDL$ = 92.6 vs. 168.0). Again this resulted both from a decrease in the number of states (15 to 7) and in the number of transitions (15 to 11). Compression did not lead to new planes or observable regularities, except that the one new sentence of length 4 which was added, bbba, was also a distance 2 from its nearest neighbor. Here, however, generalization was such that sentence lengths other than 4 were allowable, ranging from lengths of 2 to infinity, so long as certain regularities, such as embedded tandem repeats of bb, are observed.

## 6 Compression of languages

Compression results from exploiting regularities in the data. We expect that languages with smooth string spaces contain more regularity to be exploited than languages with rugged string spaces. Therefore, we expect more compression from smooth languages. Is this the case?

We created 40 languages with each of several degrees of smoothness. Each consisted of 10 strings, with lengths of 6 symbols drawn from the alphabet $\{a, b\}$. Smoothness was varied as follows: *distance-1, distance-2, distance-3* and *random*. All strings were 1, 2, 3 or random symbol-substitutions, respectively, from their nearest neighbor, drawn successively from a random starting string.

These languages were presented to agents who created the finite state automata grammars for them, then compressed the grammars using the algorithm described above. Compression was measured by $MDL$ of original and compressed grammars. Error was measured by presenting the agents with all possible strings of length 6, then counting the number that were accepted but were not strings in the original language. Recall that the compression algorithm will always accept the original language, so error can also be viewed as the amount of generalization (to strings of length 6) that the grammar achieves; it says nothing about error in recognizing strings of other lengths.

As would be expected, smoother languages could be encoded more economically than the rugged ones. The amount of compression that was achieved is measured by the compressed $MDL$ which was: 126.6 for distance-1, 199.6 for distance-2, 174.8 for distance-3 and 181.4 for the random language.

It is evident that compressed $MDL$ was much less (i.e. more compression was possible) for the distance-1 languages than for the distance-2 ones. This was expected. But the grammars for distance-3 and random languages suddenly became more compressible.

One possibility for the higher compression of distance-3 and random might be that they were unable to extract regularity and simply accepted more strings into the language. The mean numbers of errors was consistent with this explanation: 7.18 for distance-2, 29.37 for distance-3 and 27.42 for random languages. However, the mean number of errors for distance-1 was also large, 23.48, so at best the explanation is still unclear. It is possible that this will change if the agents are presented with more examples, so that the cost of encoding the data is significantly higher for distance-3 and random landscapes. We are attempting to understand this better.

## 7 Evolution of language

We explored the role of compression in the evolution of language in a simple way. Each language initially began with 10 strings of length 6, drawn from the alphabet $\{a, b\}$ as above. There were 10 replicates of distance-1 languages and 10 of distance-2. The parent in generation $n$ then produced 10 sentences, at random from its language, and presented these to generation $n + 1$, who would generate the appropriate grammar. The generation $n + 1$ would then compress that grammar and, with the compressed grammar, produce 10 more sentences for generation $n + 2$ and so on. This was continued for 10 generations.

There were clear changes observed with all languages. Foremost among the changes were the numbers of strings in the language. While all began with an average 7.7 strings without duplicates, after 10 generations this had changed to an average of 11.8 for distance-1 and to 6.1 for distance-2. This is statistically significant at the .05 level by a paired t-test. Transmission from generation 0 to 10 was lossy, with the average number of strings that were included in the original set of examples lost by generation 10 being 2.3 for distance-1 and 5.3 for distance-2.

The mean smallest distance between strings stayed about the same in both distance-1 (from 1.0 to 1.18) and distance-2 (from 2.0 to 1.72). We had expected that smoothness would increase with time, but this apparently was not the case since there were not significant changes. An increase must be expected from the distance-1 languages, because they began at the lowest value possible, and there is nowhere else to go but up. The mean distance did increase slightly. For the distance-2 languages, however, the mean smallest distance decreased only slightly.

Somewhat paradoxically, the mean $MDL$ for both grammars decreased, in spite of their lack of change in ruggedness. Figure 5 illustrates the change in $MDL$ which occurred, showing a large difference in the early generations, with only modest change later.

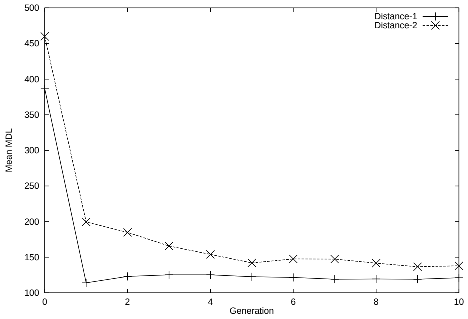

**Fig. 5.** Mean Minimum Description Lengths for evolving languages of symbol substitution distances 1 and 2 for generations 0 to 10.

It is clear that while the complexity of the grammar was decreased in both, the distance-2 grammars remained more complex, though they admitted a much smaller number of sentences.

## 8 Discussion

We have examined several conjectures about compression in adaptive complex systems. We employed agents that could learn and evolve one class of formal languages, using one compression algorithm. This learning and evolution was based on syntax alone, without any reference to semantics, which is likely to be important [10]. Our results, while so limited, do seem illuminating.

*Conjecture 1: Compression aids in generalization* This was examined with systems that learned and compressed languages of varied ruggedness. In all cases we observed a compression of grammar, and in nearly all cases the compressed grammars generalized to admit new strings into the language.

It should be recognized that such generalization may be desirable, because it reduces the complexity of rules, but it can also admit mistakes. To date we have made only a cursory study of the tradeoff between $MDL$ and error in our system, but there clearly are important issues that will warrant further study.

*Conjecture 2:Compression occurs more easily in a "smooth", as opposed to "rugged", string space* We explored compression in systems where strings were 1-, 2-, or 3- symbol substitutions apart from their nearest neighbor or were randomly placed in the sentence space. We observed that even the uncompressed grammars were smaller for the smooth than for the rugged languages. Compression, as measured by compressed $MDL$, was clearly greater in in languages where sentences were 2 symbol substitutions apart than if they were only 1 substitution apart. We interpret this to mean that patterns can more easily be identified and exploited by the compression algorithm in the smoother language. However, where smoothness is still less, in the distance-3 and random languages, the compression is also greater than for distance-2 languages. The reason(s) for this remain unclear, but there is a correlation between compression ratio, error rate, and number of examples presented that is important here and needs further exploration.

*Conjecture 3: Constraints from compression make it likely that natural languages come to have smooth string spaces*

For the purposes of this paper we accept as true the theory that when learning language we (a) compress grammar with simple rules and (b), all else being equal, apply these compressed rules preferentially to new situations. The agents in our study automatically created grammars which admitted all legal sentences. These grammars eliminated some redundancy, but retained logical equivalence. They were already *programmed*, quite literally, to conform to this theory. They were also programmed to compress those grammars, when so directed.

Here it is important to distinguish carefully between the smoothness of a language and the complexity of the grammar needed to describe it. While related, they are not the same. Both distance-1 and distance-2 languages evolved into languages with sentences that were, on average, separated by about the same number of symbol substitutions as when they started. That is to say, they did not become smoother in the sense of becoming more connected or to consist of strings lying together on the same hyperplane. At the same time, the grammars describing them came to have smaller $MDL$s. That is, both languages came to be described by simpler grammars. When we generated languages for compression, we observed that the smoother languages did have, on average, smaller $MDL$s, but not invariably so. Clearly there is some subtlety. The suggestion here is that grammatical complexity does, indeed, become simpler, but that this is not the same as saying that the languages become smooth, as smoothness is used in this paper.

In his study of evolving bit strings, albeit with semantic content, Kirby [10] observed two phase transitions in grammatical structure. The first of these occurred when the languages became suddenly more expressive, with a concomitant increase in grammatical complexity. The second phase change occurred after a high degree of expressivity was achieved, then the grammatical complexity suddenly started to become much less. That study, and ours, are in agreement that even in such simple cases as evolving bit strings there are unlikely to be simple rules about changes in smoothness of languages, or grammatical complexity – though in the long run both studies did result in simpler grammars after sufficient time.

In summary, we observed broad agreement with the conjectures made prior to the start of the study. It is evident, however, that even in our simple system there is significant subtlety that must be recognized in the inductive tradeoff of generalization through compression versus error of overgeneralization.

*Acknowledgement* This work was supported by grant #5BR9720410 from the National Science Foundation. We thank Stuart Kauffman and members of the UCLA Cognitive Science LIS Research Group for helpful discussions.

## References

1. N. Chomsky. *Aspects of the Theory of Syntax.* MIT Press, Cambridge, MA, 1965.
2. N. Chomsky. *The Minimalist Program.* MIT Press, Cambridge, MA, 1995.
3. M. Gell-Mann. Talk at Santa Fe Institute, January 9 1990.
4. E. M. Gold. Complexity of automaton identification from given data. *Information and Control*, 37:302–320, 1978.
5. P. Grünwald. A minimum description length approach to grammar inference. In G. Scheler S. Wermter, E. Riloff, editor, *Symbolic, Connectionist and Statistical Approaches to Learning for Natural Language Processing*, LNCS #1040. Springer-Verlag, Berlin, 1996.
6. T. Hashimoto. Usage-based structuralization of relationships between words. In P. Husbands and I. Harvey, editors, *Fourth European Conference on Artificial Life*, pages 483 – 492, Cambridge, MA, 1997. MIT Press.
7. J.E. Hopcroft and J.D. Ullman. *Introduction to Automata theory, Languages and Computation.* Addison Wesley, Reading, MA, 1979.
8. S.A. Kauffman. *The Origins of Order: Self-Organization in Evolution.* Oxford University Press, New York, 1993.
9. M.J. Kearns and U.V. Vazirani. *An Introduction to Computational Learning Theory.* MIT Press, Cambridge, MA, 1994.
10. S. Kirby. Syntax without natural selection: How compositionality emerges from vocabulary in a population of learners. Unpublished ms., 1999.
11. S. Kirby and J. Hurford. Learning, culture and evolution in the origin of linguistic constraints. In P. Husbands and I. Harvey, editors, *Fourth European Conference on Artificial Life*, pages 493 – 502, Cambridge, MA, 1997. MIT Press.
12. M. Li and P. Vitányi. Minimum description length induction, bayesianism and kolmogorov complexity. In *1998 IEEE International Symposium on Information Theory*, MIT, Cambridge, 1998.
13. J.A. Moore. *Science as a Way of Knowing.* Harvard University Press, Cambridge, MA, 1993.
14. P. Niyogi and R.C. Berwick. The logical problem of language change. Technical Report A. I. Memo No. 1516, MIT Artificial Intelligence Laboratory, July 1995.
15. C.H. Papadimtriou and M. Sideri. On the evolution of easy instances. *Unpublished manuscript*, 1998.
16. J. Rissanen and E. Ristad. Language acquisition in the {MDL} framework. In E. Ristad, editor, *Language Computations.* American Mathematical Society, Philadelphia, PA, 1994.
17. K. Sayood. *Introduction to Data Compression.* Morgan Kauffman, San Francisco, CA, 1996.
18. L. B. Slobodkin. *Simplicity and Complexity in Games of the Intellect.* Harvard University Press, Cambridge, MA, 1992.
19. E. Stabler. *Computational Minimalism: Acquiring and Parsing Languages with Movement.* Basil Blackwell, Oxford, 1999.
20. V.N. Vapnik. *Statistical Learning Theory.* Wiley, NY, 1998.
21. J. Weber and C. Wong. Mutation of human short tandem repeats. *Human Molecular Genetics*, 2(8):1123–1128, 1993.