Computation at the Onset of Chaos

James P. Crutchfield

Karl Young

Physics Department*
University of California
Berkeley, California 94720, USA

Abstract

Computation at levels beyond storage and transmission of information appears in physical systems at phase transitions. We investigate this phenomenon using minimal computational models of dynamical systems that undergo a transition to chaos as a function of a nonlinearity parameter. For period-doubling and band-merging cascades, we derive expressions for the entropy, the interdependence of ϵ-machine complexity and entropy, and the latent complexity of the transition to chaos. At the transition deterministic finite automaton models diverge in size. Although there is no regular or context-free Chomsky grammar in this case, we give finite descriptions at the higher computational level of context-free Lindenmayer systems. We construct a restricted indexed context-free grammar and its associated one-way nondeterministic nested stack automaton for the cascade limit language.

This analysis of a family of dynamical systems suggests a complexity theoretic description of phase transitions based on the informational diversity and computational complexity of observed data that is independent of particular system control parameters. The approach gives a much more refined picture of the architecture of critical states than is available via correlation functions, mutual information, and statistical mechanics generally. The analytic methods establish quantitatively the longstanding observation that significant computation is associated with the critical states found at the border between order and chaos.

Appearing in *Complexity, Entropy, and Physics of Information*, W. Zurek, editor, Addison-Wesley, Reading, Massachusetts (1989).

*JPC's Internet address is chaos@gojira.berkeley.edu. KY's permanent address: Physics Board of Studies, University of California, Santa Cruz, CA 95064. His Internet address is karl@gojira.berkeley.edu

Contents

Figure List

Beyond a Clock and a Coin Flip

The clock and the coin flip signify the two best understood behaviors that a physical system can exhibit. Utter regularity and utter randomness are the dynamical legacy of two millenia of physical thought. Only within this century, however, has their fundamental place been established. Today, realistic models of time-dependent behavior necessarily incorporate elements of both.

The regularity and Laplacian determinism of a clock are fundamental to much of physical theory. Einstein's careful philosophical consideration of the role of time is a noteworthy example. The use of a mechanical device to mark regular events is the cornerstone of relativity theory.† A completely predictable system, which we shall denote by $\mathbf{P_t}$, is essentially a clock; the hands indicate the current state and the mechanism advances them to the next state without choice. For a predictable system some fixed pattern is repeated every (say) $\mathbf{t}$ seconds.

Diametrically opposed, the coin flip, a picaresque example of ideal randomness, is the basic model underlying probability and ergodic theories. The next state in such a system is statistically independent of the preceding and is reached by exercising maximum choice. In ergodic theory the formal model of the coin flip is the Bernoulli flow $\mathbf{B_t}$, a coin flip every $\mathbf{t}$ seconds.

We take $\mathbf{B_t}$ and $\mathbf{P_t}$ as the basic processes with which to model the complexity of nonlinear dynamical systems. In attempting to describe a particular set of observations, if we find that they repeat then we can describe them as having been produced by some variant of $\mathbf{P_t}$. Whereas, if they are completely unpredictable then their generating process is essentially the same as $\mathbf{B_t}$. Any real system $\mathbf{S}$, of course, will contain elements of both and so naturally we ask whether it is always the case that some observed behavior can be decomposed into these separate components. Is $\mathbf{S} = \mathbf{B_t} \otimes \mathbf{P_t}$? Both ergodic and probability theories say that this cannot be done so simply in general. Ornstein showed that there are ergodic systems that cannot be separated into completely predictable and completely random processes.[1] The Wold-Kolmogorov spectral decomposition states that although the frequency spectrum of a stationary process consists of a singular spectral component associated with periodic and almost periodic behavior and a broadband continuous component associated with an absolutely continuous measure, there remain other statistical elements beyond these.[2,3,4]

What is this other behavior, captured neither by clocks nor by coin flips? A partial answer comes from computation theory and is the subject of the following.

The most general model of deterministic computation is the universal Turing machine (UTM).‡ Any computational aspect of a regular process like $\mathbf{P_t}$ can be programmed and so modeled with this machine. In order that the Turing machine *readily* model processes like $\mathbf{B_t}$ we augment it with a random register whose state it samples with a special instruction.§ The result is the Bernoulli-Turing machine (BTM). It captures both the completely predictable via its subset of deterministic operations and the completely unpredictable by accessing its stochastic

† It is not an idle speculation to wonder what happen's to Einstein's universe if his clock contains an irreducible element of randomness, or more realistically, if it is chaotic.

‡ This statement is something of an article of faith that is formulated by the Church-Turing Thesis: any reasonably specifiable computation can be articulated as a program for a UTM.[5]

§ This register can also be modeled with a second tape containing random bits. In this case, the resulting machine is referred to as an "Random Oracle" Turing Machine.[6] What we have in mind, although formally equivalent, is that the machine in question is physically coupled to an information source whose bits are random with respect the computation at hand. Thus, we do not require ideal random bits.

register. If the data is completely random, a BTM models it most efficiently by guessing. A BTM reads and prints the contents of its "Bernoulli" register, rather than implementing some large deterministic computation to generate pseudo-random numbers. What are the implications for physical theory? A variant of the Church-Turing thesis is appropriate: the Bernoulli-Turing machine is powerful enough to describe even the "other stuff" of ergodic and probability theories.

Let us delve a little further into these considerations by drawing parallels. One goal here is to infer how much of a data stream can be ascribed to a certain set of models $\{\mathbf{B_t}, \mathbf{P_t}\}$. This model basis induces a set of equivalences in the space of stationary signals. Thus, starting with the abstract notions of strict determinism and randomness, we obtain a decomposition of that space. A quantity that is constant in each equivalence class is an invariant of the modeling decomposition. Of course, we are also interested in those cases where the model basis is inadequate; where more of the computational power of the BTM must be invoked. When this occurs, it hints that the model basis should be expanded. This will then refine the decomposition and lead to new invariants.

An analogous, but restricted type of decomposition is also pursued formally in ergodic and computation theories by showing how particular examples can be mapped onto one another. The motivations being that the structure of the decomposition is a representation of the defining equivalence concept and, furthermore, the latter can be quantified by an invariant. A classic problem in ergodic theory has been to identify those systems that are isomorphic to $\mathbf{B_t}$. The associated invariant used for this is the metric entropy, introduced into dynamical systems theory by Kolmogorov[7,8] and Sinai[9] from Shannon's information theory.[10] Two Bernoulli processes are equivalent if they have the same entropy.[1] Similarly, in computation theory there has been a continuing effort to establish an equivalence between various hard-to-solve, but easily-verified, problems. This is the class of nondeterministic polynomial (NP) problems. If one can guess the correct answer, it can be verified as such in polynomial time. The equivalence between NP problems, called NP-completeness, requires that within a polynomial number of TM steps a problem can be reduced to one hardest problem.[6] The invariant of this polynomial-time reduction equivalence is the growth rate, as a function of problem size, of the computation required to solve the problem. This growth rate is called the algorithmic complexity.*

The complementarity between these two endeavors can be made more explicit when both are focused on the single problem of modeling chaotic dynamical systems. Ergodic theory is seen to classify complicated behavior in terms of information production properties, e.g. via the metric entropy. Computation theory describes the same behavior via the intrinsic amount of computation that is performed by the dynamical system. This is quantified in terms of machine size (memory) and the number of machine steps to reproduce behavior.† It turns out, as explained in more detail below, that this type of algorithmic measure of complexity is equivalent to entropy. As a remedy to this we introduce a complexity measure based on BTMs that is actually complementary to the entropy.

---

* In fact, the invariant actually used is a much coarsened version of the algorithmic complexity: a polynomial time reduction is required only to preserve the exponential character of solving a hard problem.

† We note that computation theory also allows one to formalize how much effort is required to infer a dynamical system from observed data. Although related, this is not our present concern.[11]

The emphasis in the following is that the tools of each field are complementary and both approaches are necessary to completely describe physical complexity. The basic result is that if one is careful to restrict the class of computational models assumed to be the least powerful necessary to capture behavior, then much of the abstract theory of computation and complexity can be constructively implemented.* From this viewpoint, phase transitions in physical systems are seen to support high levels of computation. And conversely, computers are seen to be physical systems designed with a subset of "critical" degrees of freedom that support computational fluctuations.

The discussion has a top-down organization with three major parts. The first, consisting of this section and the next, introduces the motivations and general formalism of applying computational ideas to modeling dynamical systems. The second part develops the basic tools of ϵ-machine reconstruction and a statistical mechanical description of the machines themselves. The third part applies the tools to the particular class of complex behavior seen in cascade transitions to chaos. A few words on further applications conclude the presentation.

Conditional Complexity

The basic concept of complexity that allows for dynamical systems and computation theories to be profitably linked relies on a generalized notion of structure that we will refer to generically as "symmetry". In addition to repetitive structure, we also consider statistical regularity to be one example of symmetry. The idea is that a data set is complex if it is the composite of many symmetries.

To connect back to the preceding discussion, we take as two basic dynamical symmetries those represented by the model basis $\{\mathbf{B_t}, \mathbf{P_t}\}$. A complex process will have, at the very least, some nontrivial combination of these components. Simply predictable behavior and purely random behavior will not be complex. The corresponding complexity spectrum is schematically illustrated in figure 1.

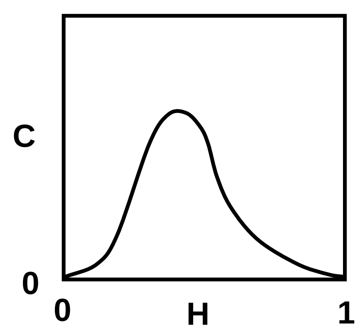

Figure 1 The complexity spectrum: complexity C as a function of the diversity of patterns. The latter is measured with the (normalized) Shannon entropy H. Regular data have low entropy; very random data have maximal entropy. However, their complexities are both low.

More formally, we define the conditional complexity $C(D|S)$ to be the amount of information in equivalence classes induced by the symmetry S in the data D plus the amount of data that

* At the highest computation level of universal Turing machines, descriptions of physical complexity are simply not constructive since finding the minimal TM program for a given problem is undecidable in general.[5]

is "unexplained" by S. If we had some way of enumerating all symmetries, then the absolute complexity $C(D)$ would be

$$C(D) = \inf_{\{S\}} C(D|S)$$

And we would say that an object is complex if, after reduction, it is its own symmetry. In that case, there are no symmetries in the object, other than itself.† If D is the best model of itself, then there is no unexplained data, but the model is large: $C(D|D) \propto \mathrm{length}(D)$. Conversely, if there is no model, then all of the data is unexplained: $C(D|\emptyset) \propto \mathrm{length}(D)$. The infimum formalizes the notion of considering all possible model "bases" and choosing those that yield the most compact description.‡

This definition of conditional modeling complexity mirrors that for algorithmic randomness[16,17,18,19,20] and is closely related to computational approaches to inductive inference.[21] A string $\mathbf{s}$ is random if it is its own shortest UTM description. The latter is a complexity measure called the Chaitin-Kolmogorov complexity $K(\mathbf{s})$ of the string $\mathbf{s}$. In the above notation $K(\mathbf{s}) = C(\mathbf{s}|UTM)$. The class of "symmetries" referred to here are those computable by a deterministic UTM. After factoring these out, any residual "unidentified" or "unexplained" data is taken as input to the UTM program. With respect to the inferred symmetries, this data is "noise". It is included in measuring the size of the minimal UTM representation. $K(\mathbf{s})$ measures the size of two components: an emulation program and input data to that emulation. To reconstruct $\mathbf{s}$ the UTM first reads in the program portion in order to emulate the computational part of the description. This computes the inferred symmetries. The (emulated) machine then queries the input tape as necessary to disambiguate indeterminant branchings in the computation of $\mathbf{s}$. $K(\mathbf{s})$ should not be confused with the proposed measure of physical complexity based on BTMs, $C(\mathbf{s}|BTM)$, which include statistical symmetries. There is, in fact, a degeneracy of terminology here that is easily described and avoided.

Consider the data in question to be an orbit $x_t(x_0)$ of duration t starting at state x_0 of a dynamical system admitting an absolutely continuous invariant measure.§ The algorithmic complexity[22] $\mathcal{A}(x_t(x_0))$ is the growth rate of the Chaitin-Kolmogorov complexity with longer orbits

$$\mathcal{A}(x_t(x_0)) = \lim_{t \to \infty} \frac{K(x_t(x_0))}{t}$$

Note that this artifice removes constant terms in the Chaitin-Kolmogorov complexity, such as those due to the particular implementation of the UTM, and gives a quantity that is machine independent. Then, the algorithmic complexity is the dynamical system's metric entropy, except

† Or, said another way, the complex object is only described by a large number of equivalence classes induced by inappropriate symmetries. The latter can be illustrated by considering an inappropriate description of a simple object. A square wave signal is infinitely complex with respect to a Fourier basis. But this is not an intrinsic property of square waves, only of the choice of model basis. There is a model basis that gives a very simple description of a square wave.

‡ This computational framework for modeling also applies, in principle, to estimating symbolic equations of motion from noisy continuous data.[12] Generally, minimization is an application of Occam's Razor in which the description is considered to be a "theory" explaining the data.[13] Rissanen's minimum description length principle, the coding theoretic version of this philosophical axiom, yields asymptotically optimal representations.[14,15]

§ In information theoretic terms we are requiring stationarity and ergodicity of the source.

for orbits starting at a measure zero set of initial conditions. These statements connect the notion of complexity of single strings with that of the ensemble of typical orbits. The Chaitin-Kolmogorov complexity is the same as informational measures of randomness, but is distinct from the BTM complexity.* To avoid this terminological ambiguity we shall minimize references to algorithmic and Chaitin-Kolmogorov complexities since in most physical situations they measure the same dynamical property captured by the information theoretic phrase "entropy". "Complexity" shall refer to conditional complexity with respect to BTM computational models. We could qualify it further by using "physical complexity", but this is somewhat misleading since it applies equally well outside of physics.†

We are not aware of any means of enumerating the space of symmetries and so the above definition of absolute complexity, while of theoretical interest, is of little immediate application. Nonetheless, we can posit that symmetries S be effectively computable in order to be relevant to scientific investigation. According to the physical variant of the Church-Turing thesis, then, S can be implemented on a BTM. Which is to say that as far as realizability is concerned, the unifying class of symmetries we have in mind is represented by operations of a BTM. Although the mathematical specification for a BTM is small; its range of computation is vast; at least as large as the underlying UTM. It is, in fact, unnecessarily powerful so that many questions, such as finding a minimal program for given data, are undecidable and many quantities, such as the conditional complexity $C(D|BTM)$, are noncomputable. More to the point, adopting too general a computational model results in there being little to say about a wide range of physical processes.

Practical measures of complexity are based on lower levels of Chomsky's computational hierarchy.‡ Indeed, Turing machines appear only at the pinnacle of this graded hierarchy. The following concentrates on deterministic finite automata (DFA) and stack automata (SA) complexity, the lowest two levels in the hierarchy. DFAs represent strictly clock and coin flip modeling. SAs are DFAs augmented by an infinite memory with restricted pushdown stack access. We will demonstrate how DFA models break down at a chaotic phase transition and how higher levels of computational model arise naturally. Estimating complexity types beyond SAs, such as linear bounded automata (LBA), is fraught with certain intriguing difficulties and will not be attempted here. Nonetheless, setting the problem context as broadly as we have just done is useful to indicate the eventual goals we have in mind and to contrast the present approach to other longstanding proposals that UTMs are the appropriate framework with which to describe the complexity of natural processes.§ Even with the restriction to Chomsky's lower levels a good deal of progress can be made since, as will become clear, contemporary statistical mechanics is largely associated with DFA modeling.

---

* We are necessarily skipping over a number of details, such as how the state x_t is discretized into a string over a finite alphabet. The basic point made here has been emphasized some time ago.[22,23]

† This definition of complexity and its basic properties as represented in figure 1 were presented by the first author at the International Workshop on "Dimensions and Entropies in Chaotic Systems", Pecos, New Mexico, 11-16 September 1985.

‡ Further development of this topic is given elsewhere.[24,25]

§ We have in mind Kolmogorov's work[19] over many years that often emphasizes dynamical and physical aspects of this problem. Also, Bennett's notion of "logical depth" and his analysis of physical processes typically employ UTM models.[26] Wolfram's suggestion[27] that the computational properties of intractability and undecidability will play an important role in future theoretical physics assumes UTMs as the model basis. More recently, Zurek[28] has taken up UTM descriptions of thermodynamic processes. The information metric used there was also developed from a conditional complexity.[29]

Reconstructing ϵ-Machines

To effectively measure intrinsic computational properties of a physical system we infer an ϵ-machine from a data stream obtained via a measuring instrument.[30] An ϵ-machine is a stochastic automaton of the minimal computational power yielding a finite description of the data stream. Minimality is essential. It restricts the scope of properties detected in the ϵ-machine to be no larger than those possessed by the underlying physical system. We will assume that the data stream is governed by a stationary measure. That is, the probabilities of fixed length blocks of measurements exist and are time-translation invariant.

The goal, then, is to reconstruct from a given physical process a computationally equivalent machine. The reconstruction technique, discussed in the following, is quite general and applies directly to the modeling task for forecasting temporal or spatio-temporal data series. The resulting minimal machine's structure indicates the inherent information processing, i.e. transmission and computation, of the original physical process. The associated complexity measure quantifies the ϵ-machine's informational size; in one limit, it is the logarithm of the number of machine states. The machine's states are associated with historical contexts, called morphs, that are optimal for forecasting. Although the simplest (topological) representation of an ϵ-machine at the lowest computational level (DFAs) is in the form of labeled directed graphs, the full development captures the probabilistic (metric) properties of the data stream. Our complexity measure unifies a number of disparate attempts to describe the information processing of nonlinear physical systems.[12,22,31,32,33,34,35,36,37] The following two sections develop the reconstruction method for the machines and their statistical mechanics.

The initial task of inferring automata from observed data falls under the purview of grammatical inference within formal learning theory.[11] The inference technique uses a particular choice S of symmetry that is appropriate to forecasting the data stream in order to estimate the conditional complexity $C(D|S)$. The aim is to infer generalized "states" in the data stream that are optimal for forecasting. We will identify these states with measurement sequences giving rise to the same set of possible future sequences.[‖] Using the temporal translation invariance guaranteed by stationarity, we identify these states using a sliding window that advances one measurement at a time through the sequence. This leads to the second step in the inference technique, the construction of a parse tree for the measurement sequence probability distribution. This is a coarse-grained representation of the underlying process's measure in orbit space. The state identification requirement then leads to an equivalence relation on the parse tree. The machine states correspond to the induced equivalence classes; the state transitions, to the observed transitions in the tree between the classes. We now give a more formal development of the inference method.

The first step is to obtain a data stream. The main modeling *ansatz* is that the underlying process is governed by a noisy discrete-time dynamical system

$$\vec{x}_{n+1} = \vec{F}(\vec{x}_n) + \vec{\xi}_n, \quad \vec{x}_0 \in \mathbf{M}$$

where $\mathbf{M}$ is the m-dimensional space of states, $\vec{x}_0 = \left(x_0^0, x_0^1, \ldots, x_0^{m-1}\right)$ is the system's initial state, $\vec{F}$ is the dynamic, the governing deterministic equations of motion, and $\vec{\xi}_n$ represents

‖ We note that the same construction can be done for past possibilities. We shall discuss this alternative elsewhere.

external time-dependent fluctuations. We shall concentrate on the deterministic case in the following. The (unknowable) exact states of the observed system are translated* into a sequence of symbols via a measurement channel.[39] This process is described by a parametrized partition

$$P_\epsilon = \left\{ c_i : \bigcup_{i=0}^{k-1} c_i = \mathbf{M},\ c_i \cap c_j = \emptyset, i \neq j;\ i,j = 0, \ldots, k-1 \right\}$$

of the state space $\mathbf{M}$, consisting of cells c_i of volume ϵ^m that are sampled every τ time units. A measurement sequence consists of the labels from the successive elements of P_ϵ visited over time by the system's state. Using the instrument $\mathbf{I} = \{P_\epsilon, \tau\}$, a sequence of states $\{\vec{x}_n\}$ is mapped into a sequence of symbols $\{s_n : s_n \in \mathbf{A}\}$, where $\mathbf{A} = \{0, \ldots, k-1\}$ is the alphabet of labels for the k $\left(\approx \epsilon^{-m}\right)$ partition elements.[12] A common example, to which we shall return near the end, is the logistic map of the interval, $x_{n+1} = r x_n (1 - x_n)$, observed with the binary generating partition $P_{\frac{1}{2}} = \{[0., .5), [.5, 1.]\}$ whose elements are labeled with $\mathbf{A} = \{0, 1\}$.[22] The computational models reconstructed from such data are referred to as ϵ-machines in order to emphasize their dependence on the measuring instrument $\mathbf{I}$.

Given the data stream in the form of a long measurement sequence $\mathbf{s} = \{s_0 s_1 s_2 \cdots : s_i \in \mathbf{A}\}$, the second step in machine inference is the construction of a parse tree. A tree $T = \{\mathbf{n}, \mathbf{l}\}$ consists of nodes $\mathbf{n} = \{n_i\}$ and directed, labeled links $\mathbf{l} = \{l_i\}$ connecting them in a hierarchical structure with no closed paths. The links are labeled by the measurement symbols $s \in \mathbf{A}$. An L-level subtree T_n^L is a tree that starts at node n and contains all nodes below n that can be reached within L links. To construct a tree from a measurement sequence we simply parse the latter for all length L sequences and from this construct the tree with links up to level L that are labeled with individual symbols up to that time. We refer to length L subsequences $s^L = \left\{ s_i \cdots s_j \cdots s_{i+L-1} : s_j = (\mathbf{s})_j \right\}$ as L-cylinders.† Hence an L level tree has a length L path corresponding to each distinct observed L-cylinder. Probabilistic structure is added to the tree by recording for each node n_i the number $N_i(L)$ of occurrences of the associated L-cylinder relative to the total number *N(L)* observed,

$$p_{n_i}^T(L) = \frac{N_i(L)}{N(L)}$$

This gives a hierarchical approximation of the measure in orbit space $\mathbf{M} \otimes \mathbf{time}$. Tree representations of data streams are closely related to the hierarchical algorithm used for estimating dynamical entropies.[22,39]

At the lowest computational level ϵ-machines are represented by a class of labeled, directed multigraph, or l-digraphs.[40] They are related to the Shannon graphs of information theory,[10] to Weiss's sofic systems in symbolic dynamics,[41] to discrete finite automata in computation theory,[5] and to regular languages in Chomsky's hierarchy.[42] Here we are concerned with probabilistic versions of these. Their topological structure is described by an l-digraph $G = \{\mathbf{V}, \mathbf{E}\}$ that consists of vertices $\mathbf{V} = \{v_i\}$ and directed edges $\mathbf{E} = \{e_i\}$ connecting them, each of the latter is labeled by a symbol $s \in \mathbf{A}$.

* We ignore for brevity's sake the question of extracting from a single component $\{x_n^i\}$ an adequate reconstructed state space.[38]

† The picture here is that a particular L-cylinder is a name for that bundle of orbits $\{\vec{x}_n\}$ each of which visited the sequence of partition elements indexed by the L-cylinder.

To reconstruct a topological ϵ-machine we define an equivalence relation, subtree similarity, denoted $\sim$, on the nodes of the tree T by the condition that the L-subtrees are identical:

$$n \sim n' \text{ if and only if } T_n^L = T_{n'}^L$$

Subtree equivalence means that the link structure is identical. This equivalence relation induces on T, and so on the measurement sequence **s**, a set of equivalence classes $\left\{ \mathbf{C}_m^L : m = 1, \ldots, K \right\}$ given by

$$\mathbf{C}_l^L = \left\{ n \in \mathbf{n} : n \in \mathbf{C}_l^L \text{ and } n' \in \mathbf{C}_l^L \text{ iff } n \sim n' \right\}$$

We refer to the archetypal subtree link structure for each class as a "morph". An l-digraph G_L is then constructed by associating a vertex to each tree node L-level equivalence class; that is, $\mathbf{V} = \left\{ \mathbf{C}_m^L \right\}$. Two vertices v_k and v_l are connected by a directed edge $e = (v_k \rightarrow v_l)$ if the transition exists in T between nodes in the equivalence classes,

$$n \rightarrow n' : n \in \mathbf{C}_k^L, n' \in \mathbf{C}_l^L$$

The corresponding edge is labeled by the symbol(s) $s \in \mathbf{A}$ associated with the tree links connecting the tree nodes in the two equivalence classes

$$\mathbf{E} = \left\{ e = (v_k, v_l; s) : \; v_k \underset{s}{\rightarrow} v_l \text{ iff } n \underset{s}{\rightarrow} n'; \; n \in \mathbf{C}_k^L, n' \in \mathbf{C}_l^L, s \in \mathbf{A} \right\}$$

In this way, ϵ-machine reconstruction deduces from the diversity of individual patterns in the data stream "generalized states", the morphs, associated with the graph vertices, that are optimal for forecasting. The topological ϵ-machines so reconstructed capture the essential computational aspects of the data stream by virtue of the following instantiation of Occam's Razor.

Theorem: Topological reconstruction of G_L produces the minimal and unique machine recognizing the language and the generalized states specified up to L-cylinders by the measurement sequence.

The generalization to reconstructing metric ϵ-machines that contain the probabilistic structure of the data stream follows by a straightforward extension of subtree similarity. Two L-subtrees are δ-similar if they are topologically similar and their corresponding links individually are equally probable within some $\delta \geq 0$. There is also a motivating theorem: metric reconstruction yields minimal metric ϵ-machines.

In order to reconstruct an ϵ-machine it is necessary to have a measure of the "goodness of fit" for determining ϵ, τ, δ, and the level L of subtree approximation. This is given by the graph indeterminacy, which measures the degree of ambiguity in transitions between graph vertices. The indeterminacy[39] I_G of a labeled digraph G is defined as the weighted conditional entropy

$$I_G = \sum_{v \in \mathbf{V}} p_v \sum_{s \in \mathbf{A}} p(s|v) \sum_{v' \in \mathbf{V}} p\big(v'|v;s\big) \log p\big(v'|v;s\big)$$

where $p(v'|v;s)$ is the transition probability from vertex v to v' along an edge labeled with symbol s, $p(s|v)$ is the probability that s is emitted on leaving v, and p_v is the probability of vertex v. A deterministically-accepting ϵ-machine is reconstructible from L-level equivalence classes if I_{G_L} vanishes. Finite indeterminacy, at some given $\{L, \epsilon, \tau, \delta\}$ indicates a residual amount of extrinsic noise at that level of approximation. In this case, the optimal machine in a set of machines consistent with the data is the smallest that minimizes the indeterminacy.[11]

Statistical Mechanics of ϵ-Machines

Many of the important properties of these stochastic automata models are given concisely using a statistical mechanical formalism that describes the coarse-grained scaling structure of orbit space. We recall some definitions and results necessary for our calculations.[30] The statistical structure of an ϵ-machine is given by a parametrized stochastic connection matrix

$$T_\alpha = \{t_{ij}\} = \sum_{s \in \mathbf{A}} T_\alpha^{(s)}$$

that is the sum over each symbol $s \in \mathbf{A}$ in the alphabet $\mathbf{A} = \{i : \; i = 0, \ldots, k-1; \; k = \mathcal{O}(\epsilon^{-m})\}$ of the state transition matrices

$$T_\alpha^{(s)} = \left\{ e^{\alpha \log p(v_i | v_j; s)} \right\}$$

for the vertices $v_i \in \mathbf{V}$. We will distinguish two subsets of vertices. The first $\mathbf{V_t}$ consists of those associated with transient states; the second $\mathbf{V_r}$, consists of recurrent states.

The α-order total Renyi entropy,[43] or "free information", of the measurement sequence up to n-cylinders is given by

$$H_\alpha(n) = (1-\alpha)^{-1} \log Z_\alpha(n)$$

where the partition function is

$$Z_\alpha(n) = \sum_{s^n \in \{s^n\}} e^{\alpha \log p(s^n)}$$

with the probabilities $p(s^n)$ defined on the n-cylinders $\{s^n\}$. The Renyi specific entropy, i.e. entropy per measurement, is approximated[22] from the n-cylinder distribution by

$$h_\alpha(n) = n^{-1} H_\alpha(n)$$
$$\text{or} \;\; h'_\alpha(n) = H_\alpha(n) - H_\alpha(n-1)$$

and is given asymptotically by

$$h_\alpha = \lim_{n \to \infty} h_\alpha(n)$$

The parameter α has several interpretations, all of interest in the present context. From the physical point of view, $\alpha \; (= 1 - \beta)$ plays the role of the inverse temperature β in the statistical mechanics of spin systems.[44] The spin states correspond to measurements; a configuration of spins on a spatial lattice to a temporal sequence of measurements. Just as the temperature increases the probability of different spin configurations by increasing the number of available states, α accentuates different subsets of measurement sequences in the asymptotic distribution. From the point of view of Bayesian inference α is a Lagrange multiplier specifying a maximum entropy distribution consistent with the maximum likelihood distribution of observed cylinder probabilities.[45] Following symbolic dynamics terminology, $\alpha = 0$ will be referred to as the topological or counting case; $\alpha = 1$, as the metric or probabilistic case or high temperature limit.

Varying α moves continuously from topological to metric machines. Originally in his studies of generalized information measures, Renyi introduced α as just this type of interpolation parameter and noted that the α-entropy has the character of a Laplace transform of a distribution.[43] Here there is the somewhat pragmatic, and possibly more important, requirement for α: it gives the proper algebra of trajectories in orbit space. That is, α is necessary for computing measurement sequence probabilities from the stochastic connection matrix T_α. Without it, products of T_α fail to distinguish distinct sequences.

An ϵ-machine's structure determines several key quantities. The first is the stochastic DFA measure of complexity. The α-order graph complexity is defined as

$$C_\alpha = (1-\alpha)^{-1} \log \sum_{v \in \mathbf{V}} p_v^\alpha$$

where the probabilities p_v are defined on the vertices $v \in V$ of the ϵ-machine's l-digraph. The graph complexity is a measure of an ϵ-machine's information processing capacity in terms of the amount of information stored in the morphs. As mentioned briefly later, the complexity is related to the mutual information of the past and future semi-infinite sequences and to the convergence[46,31] of the entropy estimates $h_\alpha(n)$. It can be interpreted, then, as a measure of the amount of mathematical work necessary to produce a fluctuation from asymptotic statistics.

The entropies and complexities are dual in the sense that the former is determined by the principal eigenvalue λ_α of T_α,

$$h_\alpha = (1-\alpha)^{-1} \log_2 \lambda_\alpha$$

and the latter by the associated left eigenvector of T_α

$$\vec{p}_\alpha = \{ p_v^\alpha : v \in \mathbf{V} \}$$

that gives the asymptotic vertex probabilities.

The specific entropy is also given directly in terms of the stochastic connection matrix transition probabilities

$$h_\alpha = \sum_{v \in \mathbf{V}} \frac{p_v^\alpha}{1-\alpha} \log \sum_{\substack{v' \in \mathbf{V} \\ s \in \mathbf{A}}} p^\alpha \left(v | v' ; s \right)$$

A complexity based on the asymptotic edge probabilities $\vec{p}_e = \{ p_e : e \in \mathbf{E} \}$ can also be defined

$$C_\alpha^e = (1-\alpha)^{-1} \log \sum_{e \in \mathbf{E}} p_e^\alpha$$

$\vec{p}_e$ is given by the left eigenvector of the ϵ-machine's edge graph. The transition complexity C_α^e is simply related to the entropy and graph complexity by

$$C_\alpha^e = C_\alpha + h_\alpha$$

There are, thus, only two independent quantities for a finite DFA ϵ-machine.[11]

The two limits for α mentioned above warrant explicit discussion. For the first, topological case $(\alpha = 0)$, T_0 is the l-digraph's connection matrix. The Renyi entropy $h_0 = \log\ \lambda_0$ is the topological entropy *h*. And the graph complexity is

$$C_0(G) = \log\ |\mathbf{V}|$$

This is $C(\mathbf{s}|\mathrm{DFA})$: the size of the minimal DFA description, or "program", required to *produce* sequences in the observed measurement language of which $\mathbf{s}$ is a member. This topological complexity counts all of the reconstructed states. It is similar to the regular language complexity developed for cellular automaton generated spatial patterns.[32] The DFAs in that case were constructed from known equations of motion and an assumed neighborhood template. Another related topological complexity counts just the recurrent states $\mathbf{V_r}$. The distinction between this and C_0 should be clear from the context in which they are used in later sections.

In the second, metric case $(\alpha = 1)$, h_α becomes the metric entropy

$$h_\mu = \lim_{\alpha \to 1} h_\alpha = -\frac{d\lambda_\alpha}{d\alpha}$$

The metric complexity

$$C_\mu = \lim_{\alpha \to 1} C_\alpha = -\sum_{v \in \mathbf{V}} p_v \log p_v$$

is the Shannon information contained in the morphs.[‡] Following the preceding remarks, the metric entropy is also given directly in terms of the stochastic connection matrix

$$h_\mu = \sum_{v \in \mathbf{V}} p_v \sum_{\substack{v' \in \mathbf{V} \\ s \in \mathbf{A}}} p\big(v|v';s\big) \log p\big(v|v';s\big)$$

A central requirement in identifying models from observed data is that a particular inference methodology produces a sequence of hypotheses that converge to the correct one describing the underlying process. The complexity can be used as a diagnostic for this since it is a direct measure of the size of the hypothesized stochastic DFA at a given reconstruction cylinder length. The identification method outlined in the preceding section converges with increasing cylinder length if the rate of change of the complexity vanishes. If, for example,

$$c_\alpha = \lim_{L \to \infty} \frac{2^{C_\alpha(L)}}{L}$$

vanishes, then the noisy dynamical system has been identified. If it does not vanish, then c_α is a measure of the rate of divergence of the model size and so quantifies a higher level of computational complexity. In this case, the model basis must be augmented in an attempt to find a finite description at some higher level. The following sections will demonstrate how this can happen. A more complete discussion of reconstructing various hierarchies of models is found elsewhere.[25]

‡ Cf. "set complexity" version of the regular language complexity[36] and "diversity" of undirected, unlabeled trees.[35]

Period-Doubling Cascades

To give this general framework substance and to indicate the importance of quantifying computation in physical processes, the following sections address a concrete problem: the complexity of cascade transitions to chaos. The onset of chaos often occurs as a transition from an ordered (solid) phase of periodic behavior to a disordered (gas) phase of chaotic behavior. A cascade transition to chaos consists of a convergent sequence of individual "bifurcations", either pitchfork (period-doubling) in the periodic regimes or band-merging in the chaotic regimes.*

The canonical model class of these transitions is parametrized two-lap maps of the unit interval, $x_{n+1} = f(x_n),\ x_n \in [0,1]$, with negative Schwartzian derivative; that is, those maps with two monotone pieces and admitting only a single attractor. We assign to the domain of each piece the letters of the binary alphabet $\Sigma = \{0,1\}$. The sequence space Σ^* consists of all 0-1 sequences. Some of these maps, such as the piecewise-linear tent map described in a later section, need not have the period-doubling portion of the cascade. Iterated maps are canonical models of cascade transitions in the sense that the same bifurcation sequence occurring in a set of nonlinear ordinary differential equations (say) is topologically equivalent to that found in some parametrized map.[47,48,49]

Although ϵ-machines were developed in the context of reconstructing computational models from data series, the underlying theory provides an analytic approach to calculating entropies and complexities for a number of dynamical systems. This allows us to derive in the following explicit bounds on the complexity and entropy for cascade routes to chaos.

We focus on the periodic behavior near pitchfork bifurcations and chaotic behavior at band-mergings with arbitrary basic periodicity.[50,51] In distinction to the description of universality of the period-doubling route to chaos in terms of parameter variation,[52] we have found a phase transition in complexity that is not explicitly dependent on control parameters.[30] The relationship between the entropy and complexity of cascades can be said to be super-universal in this sense. This is similar to the topological equivalence of unimodal maps of the interval,[53,54,55,56,57] except that it accounts for statistical and computational structures associated with the behavior classes.

In this and the next sections we derive the total entropy and complexity as a function of cylinder length n for the set of ϵ-machines describing the behavior at the different parameter values for the period-doubling and band-merging cascades. The sections following this then develop several consequences, viz. the order and the latent complexity of the cascade transition. With these statistical mechanical results established, the discussion turns to a detailed analysis of the higher level computation at the transition itself.

In the periodic regime below the periodicity $q = 1$ cascade transition we find the ϵ-machines for m-order period-doubling $2^m \rightarrow 2^{m+1}$ $(m = 0,1,2,3)$ shown in figures 2 - 5.

Figure 2 Topological l-digraph for period 1 attractor.

Figure 3 Topological l-digraph for period 2 attractor.

* The latter are not, strictly speaking, bifurcations in which an eigenvalue of the linearized problem crosses the unit circle. The more general sense of bifurcation is nonetheless a useful shorthand for qualitative changes in behavior as a function of a control parameter.

Figure 4 Topological l-digraph for period 4 attractor.

Figure 5 Topological l-digraph for period 8 attractor.

For periodic behavior the measure on the n-cylinders $\{s^n\}$ is uniform; as is the measure on the recurrent ϵ-machine states $\mathbf{V_r}$. Consider behavior with period $P = q \times 2^m$ at a given m-order period-doubling with basic cascade periodicity q. The uniformity allows us to directly estimate the total entropy in terms of the number $N(n, m)$ of n-cylinders with $n > P$

$$\begin{aligned} H_\alpha(n,m) &= (1-\alpha)^{-1} \log \sum_{s^n \in \{s^n\}} p^\alpha(s^n) \\ &= (1-\alpha)^{-1} \log \frac{N(n,m)}{N^\alpha(n,m)} \\ &= \log N(n,m) \end{aligned}$$

For periodic behavior and assuming$n > P$ the number of n-cylinders is given by the period $N(n,m) = P$. The total entropy is then $H_\alpha(n,m) = \log P$. Note that, in this case, h_α vanishes.

Similarly, the complexity is given in terms of the number $V_r = |\mathbf{V_r}|$ of recurrent states

$$\begin{aligned} C_\alpha &= (1-\alpha)^{-1} \log \sum_{v \in \mathbf{V}} p_v^\alpha \\ &= (1-\alpha)^{-1} \log |\mathbf{V}_\mathrm{r}|^{1-\alpha} \\ &= \log V_r \end{aligned}$$

The number V_r of vertices is also given by the period for periodic behavior and so we find $C_\alpha = \log P$. Thus, for periodic behavior the relationship between the total and specific entropies and complexity is simple

$$\begin{aligned} C_\alpha &= H_\alpha \\ \text{or} \quad C_\alpha &= n h_\alpha(n) \end{aligned}$$

This relationship is generally true for periodic behavior and is not restricted to the situation where dynamical systems have produced the data. Where noted in the following we will also use $C_0 = \log |\mathbf{V}|$ to measure the total number of machine states.

Chaotic Cascades

In the chaotic regime the situation is much more interesting. The ϵ-machines at periodicity $q = 1$ and m-order band-mergings$2^m \to 2^{m-1},\ m = 0, 1, 2, 3$, are shown in figures 6 - 9.

Figure 6 Topological l-digraph for single band chaotic attractor.

Figure 7 Topological l-digraph for $2 \to 1$ band chaotic attractor.

Figure 8 Topological l-digraph for 4 → 2 band chaotic attractor.

Figure 9 Topological l-digraph for 8 → 4 band chaotic attractor.

The graph complexity is still given by the number V_r of recurrent states as above. The main analytic task comes in estimating the total entropy. In contrast to the periodic regime the number of distinct subsequences grows with n-cylinder length for all n. Asymptotically, the growth rate of this count is given by the specific topological entropy. In order to estimate the total topological entropy at finite n, however, more careful counting is required than in the periodic case. This section develops an exact counting technique for all cylinder lengths that applies at chaotic parameter values where the orbit $f^n(x^*)$ of the critical point x^*, where $f'(x^*) = 0$, is asymptotically periodic. These orbits are unstable and embedded in the chaotic attractor. The set of such values is countable. At these (Misiurewicz) parameters there is an absolutely continuous invariant measure.[58]

There is an additional problem with the arguments used in the periodic case. The uniform distribution of cylinders no longer holds. The main consequence is that we cannot simply translate counting $N(n,m)$ directly into an estimate of $H_{\alpha\neq 0}(n,m)$. One measure of the degree to which this is the case is given by the difference in the topological entropy h and the metric entropy h_μ.[22]

Approximations for the total Renyi entropy can be developed using the exact cylinder counting methods outlined below and the machine state and transition probabilities from $\left\{T_\alpha^{(s)}\right\}$. The central idea for this is that the states represent a Markov partition of the symbol sequence space Σ^*. There are invariant subsets of Σ^*, each of which converges at its own rate to "equilibrium". Each subset obeys the Shannon-McMillan theorem[59] individually. At each cylinder length each subset is associated with a machine state. And so the growth in the total entropy in each subset is governed by the machine's probabilistic properties. Since the cylinder counting technique captures a sufficient amount of the structure, however, we will not develop the total Renyi entropy approximations here and instead focus on the total topological entropy.

We now turn to an explicit estimate of $N(n,m)$ for various cases. Although the techniques apply to all Misiurewicz parameters, we shall work through the periodicity $q = 1$ 2 → 1, 4 → 2, and 1 → 0 band-merging transitions (figure 6 - 9) in detail, and then quote the general formula for arbitrary order of band-merging.

The tree for 2 → 1 band merging n-cylinders is shown in figure 10.

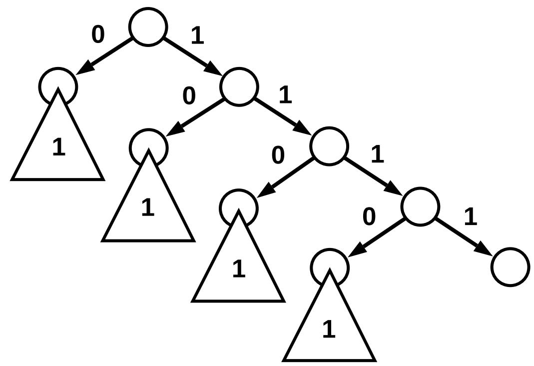

Figure 10 Parse tree associated with two chaotic bands merging into one. Tree nodes are shown for the transient spine only. The subtrees associated with asymptotic behavior, and so also with the equivalence classes corresponding to recurrent graph vertex 1 in figure 7, are indicated schematically with triangles.

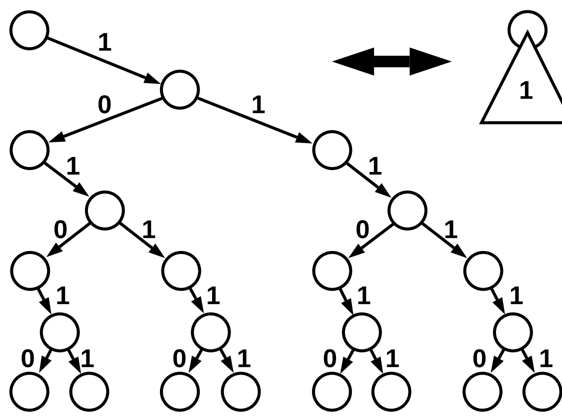

Figure 11 Subtree of nodes associated with asymptotic vertices in l-digraph for two bands merging to one.

An exact expression for $N(n,1)$ derives from splitting the enumeration of unique n-cylinders as represented on the tree into recurrent and transient parts. For two bands, Figure 10 illustrates the transient spine, the set of tree nodes associated with transient graph states, while schematically collapsing that portion of the tree associated with asymptotic graph vertices. The latter is shown in Figure 11. As will become clear the structure of the transient spine in the tree determines the organization of the counting method.

The sum for the n^{th} level, i.e. for the number of n-cylinders, is

$$N(n,1) = 1 + \sum_{i=0}^{\lfloor \frac{n}{2} \rfloor} 2^i + \sum_{i=0}^{\lfloor \frac{n-1}{2} \rfloor} 2^i$$

where $\lfloor k \rfloor$ is the largest non-negative integer less than k. The second term on the right counts the number of tree nodes that branch at even numbered levels, the third term is the number that branch at odd levels, and the first term counts the transient spine that adds a single cylinder.

For $n \geq 2$ and even, this can be developed into a renormalized expression that yields a closed form as follows

$$\begin{aligned} N(n,1) &= 1 + 2 \sum_{i=0}^{\frac{n-2}{2}} 2^i \\ &= 1 + 2 \left(1 + 2 \sum_{i=0}^{\frac{n-2}{2}} 2^i - 2 \cdot 2^{\frac{n-2}{2}} \right) \\ &= 1 + 2 \left(N(n,1) - 2^{\frac{n}{2}} \right) \\ \text{or} \quad N(n,1) &= 2 \left(2^{\frac{n}{2}} - 2^{-1} \right) \end{aligned}$$

For $n \geq 2$ and odd, we find $N(n,1) = 3 \cdot 2^{\frac{n-1}{2}} - 1$. This gives an upper bound on the growth envelope as a function of n. The former, a lower bound.

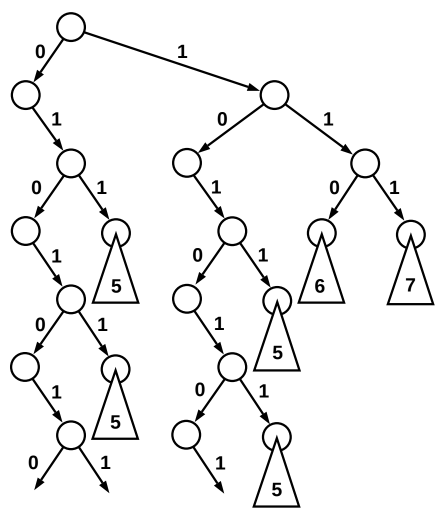

Figure 12 Transient spine for 4 → 2 band attractor. The asymptotic subtrees are labeled with the associated l-digraph vertex. (Compare figure 8.)

The analogous expression for the 4 → 2 band cylinder count can be explicitly developed. Figure 12 shows the transient spine on the tree that determines the counting structure. In this case, the sum is

$$N(n,2) = 2 + 2^{\left\lfloor \frac{n-4}{4} \right\rfloor} + 2^{\left\lfloor \frac{n-4}{4} \right\rfloor} + \sum_{i=0}^{\left\lfloor \frac{n-3}{4} \right\rfloor} 2^i + \sum_{i=0}^{\left\lfloor \frac{n-5}{8} \right\rfloor} 2^i + \sum_{i=0}^{\left\lfloor \frac{n-4}{4} \right\rfloor} 2^i + \sum_{i=0}^{\left\lfloor \frac{n-6}{8} \right\rfloor} 2^i$$

There are seven terms on the right hand side. In order they account for

1. The two transient cycles, begun on 0 and 1, each of which contributes 1 node per level;
2. Cycles on the attractor that are fed into the attractor via non-periodic transients (second and third terms);
3. Sum over tree nodes that branch by a factor of 2 at level $k + 4i, k = 3, 4, 5, 6$, respectively.

The sum greatly simplifies upon rescaling the indices to obtain a self-similar form. For $n \geq P = 4$ and $n = 4i$, we find

$$\begin{aligned} N(n,2) &= 2\left(1 + 2^{\frac{n-4}{4}} + \sum_{i=0}^{\frac{n-4}{4}} 2^i + \sum_{i=0}^{\frac{n-8}{8}} 2^i\right) \\ &= 2 + 4\left(1 + \sum_{i=0}^{\frac{n-4}{4}} 2^i\right) \\ &= 2 + 4\left(1 + 2\sum_{i=0}^{\frac{n-4}{4}} 2^i - 2^{\frac{n}{4}}\right) \\ &= 2 + 2\left(N(n,2) - 2^{\frac{n+4}{4}}\right) \\ \text{or } N(n,2) &= 2^2 2^{\frac{n}{4}} - 2 \end{aligned}$$

There are three other phases for the upper bound as a function of n.

For completeness we note that this approach also works for the single band $(m = 0)$ case

$$\begin{aligned} N(n,0) &= 1 + \sum_{i=0}^{n-1} 2^i \\ &= 1 + \left(2 + 2\sum_{i=0}^{n-1} 2^i - 2^n - 1\right) \\ &= 2N(n,0) - 2^n \\ \text{or } N(n,0) &= 2^n \end{aligned}$$

The preceding calculations were restricted by the choice of a particular phase of the asymptotic cycle at which to count the cylinders. With a little more effort a general expression for all phases is found. Noting the similarity of the l-digraph structures between different order band-mergings and generalizing the preceding recursive technique yields an expression for arbitrary order band-merging. This takes into account the fact that the generation of new n-cylinders via branching occurs at different phases on the various limbs of the transient spine. The number of n-cylinders from the exact enumeration for the $q = 1$ $2^m \to 2^{m-1}$ band-merging is

$$N(n,m) = \begin{cases} 2^n & m = 0 \\ 2^m\left(b_{n,m} 2^{n2^{-m}} - 2^{-1}\right) & m \neq 0 \end{cases}$$

where $n > P = 2^m$ and $b_{n,m} = (1 + \hat{n})2^{-\hat{n}}$ and $\hat{n} = 2^{-m}(n \bmod 2^m)$ account for the effect of relative branching phases in the spine. This coefficient is bounded

$$\begin{aligned} b_{min} &= \inf_{\{n,m \neq 0\}} b_{n,m} = 1 \\ b_{max} &= \sup_{\{n,m \neq 0\}} b_{n,m} = 3 \cdot 2^{-\frac{3}{2}} \approx 1.0606602 \end{aligned}$$

The second bound follows from noting that the maximum occurs when, for example, $n = 2^m + 2^{m-1}$. Note that the maximum and minimum values of the prefactor are independent of the phase and of n and m. We will ignore the detailed phase dependence and simply write b instead of $b_{n,m}$ and consider the lower bound case of $b = 1$.

Recalling that $C_0 = \log |V_r| = m$, we have

$$N(n) = 2^{C_0}\left(b2^{n2^{-C_0}} - 2^{-1}\right)$$

and the total (topological) entropy is given by

$$H_0(n) = \log_2 N(n)$$
$$H_0(n) = C_0 + \log_2\left(2^{n2^{-C_0}} - 2^{-1}\right)$$

where we have set $b = 1$. The first term recovers the linear interdependence that derives from the asymptotic periodicity; cf. the period-doubling case. The second term is due to the additional feature of chaotic behavior that, in the band-merging case, is reflected in the branching and transients in the l-digraph structure. In terms of the modeling decomposition introduced at the beginning, the first term corresponds to the periodic process $\mathbf{P_t}$ and the branching portion of the second term, to components isomorphic to the Bernoulli process $\mathbf{B_t}$.

From the development of the argument, we see that the factor 2^{-m} in the exponent controls the branching rate in the asymptotic cycle and so should be related to the rate of increase of the number of cylinders. The topological entropy is the growth rate of H_0 and so can now be determined directly

$$h_0(m) = \lim_{n\to\infty} \frac{H_0(n)}{n} = 2^{-m}$$

Rewriting the general expression for the lower bound in a chaotic cascade makes it clear how h_0 controls the total entropy

$$N(n,m) = V_r\left(2^{nh} - 2^{-1}\right)$$

where $h = \frac{f}{V_r}$ is the branching ratio of the number of vertices f that branch to the total number V_r of recurrent states.

The above derivation used periodicity $q = 1$. For general periodicity band-merging, we have $V_r = q \cdot 2^m$ and $f = 1$. It is clear that the expression works for a much wider range of ϵ-machines with isolated branching within a cycle that do not derive from cascade systems. Indeed, the results concern the relationship between eigenvalues and asymptotic state probabilities in the family of labeled Markov chains with isolated branching among cyclic recurrent states.

As a subset of all Misiurewicz parameter values, band-merging behavior has the simplest computational structure. In closing this section, we should point out that there are other cascade-related families of Misiurewicz parameters whose machines are substantially more complicated in the sense that the stochastic element is more than an isolated branching. Each family is described by starting with a general labeled Markov chain as the lowest order machine. The other family members are obtained by applications of a period-doubling operator.[47] Each is a product of a periodic process and the basic stochastic machine. As a result of this simple decomposition, the complexity-entropy analysis can be carried out. This will be reported elsewhere. It explains many of the complexity-entropy properties above the lower bound case of band-merging. The numerical experiments later give examples of all these types of behavior.

Cascade Phase Transition

The preceding results are used in this section to demonstrate that the cascade route to chaos has a complexity-entropy phase transition. It was established some time ago that this route to chaos is a phase transition as a function of a nonlinearity parameter,[52] with an external (dis)ordering field[51] and a natural (dis)order parameter.[60] Here we focus on the information processing properties of this transition. First, we estimate for finite cylinder lengths the complexity and specific entropy at the transition. Second, we define and compute the transition's latent complexity that gives the computational difference between ϵ-machines above and below the transition. Finally, we discuss the transition's order.

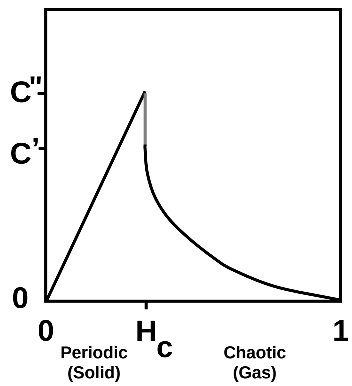

Figure 13 Complexity versus specific entropy estimate. Schematic representation of the cascade lambda transition at finite cylinder lengths. Below H_c the behavior is periodic; above, chaotic. The latent complexity is given by the difference of the complexities C'' and C' at the transition on the periodic and chaotic branches, respectively.

Given the lower bound expressions for the entropy and complexity above and below the transition to chaos as a function of cylinder length n, we can easily estimate the complexities $C'(n)$ and $C''(n)$ and the critical entropy $H_c(n)$. Figure 13 gives a schematic representation of the transition and shows the definitions of the various quantities. The transition is defined as the divergence in the slope of the chaotic branch of the complexity-entropy curve. That is, the critical entropy H_c and complexity C' are defined by the condition

$$\frac{\partial H}{\partial C} = 0$$

From this, we find

$$C' = \log_2 n - \log_2 \log_2 y$$
$$nH_c = C' + \log_2 \left(by - 2^{-1} \right)$$

where $y = 2^{n2^{-C'}}$ is the solution of

$$y \log_e y - y + \frac{1}{2} = 0$$

that is, $y \approx 2.155535035$. Numerical solution for $n = 16$ gives

$$C'(16) \approx 3.851982$$
$$C''(16) \approx 4.579279$$
$$H_c(16) \approx 0.286205$$

at $b = 1$.

The latent complexity ΔC of the transition we define as the difference at the critical entropy H_c of the complexities on the periodic and chaotic branches

$$\Delta C = C'' - C'$$

Along the periodic branch the entropy and complexity are equal and so from the previous development we see that

$$nH_c = C'' = C' + \log_2 \left(by - \frac{1}{2} \right)$$
$$\text{or } \Delta C = \log_2 \left(by - \frac{1}{2} \right)$$

For $b = 1$ this gives by numerical solution

$$\Delta C \approx 0.7272976887 \text{ bit}$$

which, we note, is independent of cylinder length.

In classifying this transition thermodynamically, the complexity plays the role of a heat capacity. It is by our definition a computational "capacity". Just as the thermodynamic temperature controls the multiplicity of available states, H appears as an "informational" temperature and H_c as a critical amount of information (energy) per symbol (spin) at which long range fluctuations occur. The overall shape is then similar to a lambda phase transition in that there is a gradual increase in the capacity from both sides and a jump discontinuity in it at the transition. The properties supporting this follow from the bounds developed earlier. And so, there is at least one component of the cascade transition that is a second order transition, i.e. that associated with periodicity $q = 1$. There is also a certain degeneracy due to the phase dependence of the coefficient $b_{n,m}$. This is a small effect, but it does indicate a range of different limiting values as $n \to \infty$ for the chaotic critical complexity C'. It does not change the order of the transition. To completely characterize the transition, though, an upper bound on complexity at fixed n is also needed. This requires accounting for the typical chaotic parameters, by which we mean those associated with aperiodic behavior of the critical point. An approach to this problem will be reported elsewhere.

It should also be emphasized that the above properties were derived for finite cylinder lengths; that is, far away from the thermodynamic limit of infinite cylinders. The overall shape and qualitative properties hold not only in the thermodynamic limit but also at each finite size. In the thermodynamic limit the entropy estimates $n^{-1}H(n)$ go over to the entropy growth rates h_α. As a result, all of the periodic behavior lies on the $h_\alpha = 0$ line in the (h_α, C_α)-plane. This limiting behavior is consistent with a zero temperature phase transition of a one-spatial-dimension spin system with finite interaction range.

This analysis of the cascade phase transition should be contrasted with the conventional descriptions based on correlation function and mutual information decay. The correlation length of a statistical mechanical system is defined most generally as the minimum size L at which there is no qualitative statistical difference between the system of size L and the infinite (thermodynamic limit) system. This is equivalent in the present context to defining a correlation length L_α at which L-cylinder α-order statistics are close to asymptotic.* If we consider the total entropy $H_\alpha(L)$ as the (dis)order parameter of interest, then for finite ϵ-machines,† away from the transition on the chaotic side, we expect its convergence to asymptotic statistics to behave like

$$2^{H_\alpha(L)} \propto 2^{\frac{L}{L_\alpha}}$$

But for L sufficiently large

$$2^{H_\alpha(L)} \propto 2^{h_\alpha L}$$

where $h_\alpha = \log_2 \lambda_\alpha$. By this argument, the correlation length is simply related to the inverse of the specific entropy: $L_\alpha \propto h_\alpha^{-1}$. We would conclude, then, that the correlation function description of the phase transition is equivalent in many respects to that based on specific entropy.

Unfortunately, this argument, which is often used in statistical mechanics, confuses the rate of decay of correlation with the correlation length. These quantities are proportional only assuming exponential decay or, in the present case, assuming finite ϵ-machines. The argument does indicate that as the transition is approached the correlation length diverges since the specific entropy vanishes. For all behavior with zero metric entropy, periodic or exactly at the transition, the correlation length is infinite. As typically defined, it is of little use in distinguishing the various types of zero entropy behavior.

The correlation length in statistical mechanics is determined by the decay of the two-point autocorrelation function

$$\mathcal{C}(L) = \langle s_i s_{i+L} \rangle = \frac{1}{N} \sum_{i=0}^{N-1} \left(s_i s_{i+L} - s_i^2 \right)$$

Its information theoretic analog is the two–point 1-cylinder mutual information

$$I_\alpha(\mathbf{s}_i, \mathbf{s}_{i+L}) = H_\alpha(\mathbf{s}_i) - H_\alpha(\mathbf{s}_{i+L} | \mathbf{s}_i)$$

where $\mathbf{s}_i$ is the i^{th} symbol in the sequence $\mathbf{s}$ and $H_\alpha(\cdot)$ is the Renyi entropy.‡ Using this to describe phase transitions is an improvement over the correlation function in that, for periodic data, it depends on the period $P: \;\; I_\alpha \propto \log P$. In contrast, the correlation function in this case does not decay and gives an infinite correlation length.

The convergence of cylinder statistics to their asymptotic (thermodynamic limit) values is most directly studied via the total excess entropy[30,46,61]

$$F_\alpha(L) = H_\alpha(L) - h_\alpha L$$

It measures the total deviation from asymptotic statistics, up to L-cylinders.§ As $L \to \infty$, it measures the average mutual information between semi-infinite past and future sequences. It

* Cf. the entropy "convergence knee" n^*.[31]

† The statistical mechanical argument, from which the following is taken, equivalently assumes exponential decay of the correlation function.

‡ The correlation length is most closely related to I_2.

§ A scaling theory for entropy convergence to the thermodynamic limit that includes the effect of extrinsic noise has been is described previously.[31]

follows from standard information theoretic inequalities that the two-point 1-cylinder mutual information is an overestimate of the excess entropy and so of the convergence properties. In particular,

$$L^{-1} F_\alpha(L) \leq I_\alpha(\mathbf{s}_i, \mathbf{s}_{i+L})$$

since I_α ignores statistical dependence on the symbols between $\mathbf{s}_i$ and $\mathbf{s}_{i+L}$. The DFA ϵ-machine complexity is directly related to the total excess entropy[30]

$$C_\alpha(L) \underset{L\to\infty}{\propto} F_\alpha(L)$$

As a tool to investigate computational properties, the two-point mutual information is too coarse, since it gives at most an upper bound on the DFA complexity.

At the transition correlation extends over arbitrarily long temporal and spatial scales and fluctuations dominate. It is the latter that support computation at higher levels in Chomsky's hierarchy. The computational properties at the phase transition are captured by the diverging ϵ-machines' structure. To the extent that their computational structure can be analyzed, a more refined understanding of the phase transition can be obtained.

Cascade Limit Language

The preceding section dealt with the statistical character of the cascade transition, but we actually have much more information available from the ϵ-machines. Although the DFA model diverges in size, its detailed computational properties at the phase transition reveal a finite description at a higher level in Chomsky's hierarchy. With this we obtain a much finer classification than is typical in phase transition theory.

The structure of the limiting machine can be inferred from the sequence of machines reconstructed at $2^m \rightarrow 2^{m+1}$ period-doubling bifurcation on the periodic side and from those reconstructed at $2^m \rightarrow 2^{m-1}$ band-merging on the chaotic side. (Compare figures 2 and 6, 3 and 7, 4 and 8, 5 and 9.) All graphs have transient states of pair-wise similar structure, except that the chaotic machines have a period 2^{m-1} unstable cycle. All graphs have recurrent states of period 2^m. In the periodic machines this cycle is deterministic. In the chaotic machines, although the states are visited deterministically, the edges have a single nondeterministic branching.

The order of the phase transition depends on the structural differences between the ϵ-machines above and below the transition to chaos. In general, if this structural difference alters the complexity at constant entropy, then the transition will be second order. At the transition to chaos via period doubling there is a difference in the complexities due to

1. The single vertex in the asymptotic cycle that branches; and
2. The transient 2^{m-1} cycle in the machines on the chaotic side.

At constant complexity the uncertainty developed by the chaotic branching and the nature of the transient spine determine the amount of dynamic information production required to make the change from predictable to chaotic ϵ-machines.

The following two subsections summarize results discussed in detail elsewhere.

Critical Machine

The machine M that accepts the sequences produced at the transition, although minimal, has an infinite number of states. The growth of machine size $|\mathbf{V}(L)|$ versus reconstruction cylinder size L at the transition is demonstrated in figure 14. The maximum growth is linear with slope $c_0 = 3$. Consequently, the complexity diverges logarithmically.* The growth curve itself is composed of pieces with alternating slope 2 and slope 4

$$|\mathbf{V}(L)| = \begin{Bmatrix} 2L & 2^i \leq L < 3 \cdot 2^{i-1} \\ 4L & 3 \cdot 2^{i-1} \leq L < 2^{i+1} \end{Bmatrix}$$

The slope 2 learning regions correspond to inferring more of the states that link the upper and lower branches of the machine. (The basic structure will be made clearer in the discussion of figure 15 below.) The slope 4 regions are associated with picking up groups of states along the long deterministic chains that are the upper and lower branches. Recalling the definition of c_α in a previous section, we note that finite c_0 indicates a constant level of complexity using a more powerful computational model than $\{\mathbf{P_t}, \mathbf{B_t}\}$.

Figure 14 Growth of critical machine M. The number $|\mathbf{V}(L)|$ of reconstructed states versus cylinder length L for the logistic map at the periodicity $q = 1$ cascade transition. Reconstruction is from length 1 to length 64 cylinders on $2L$-cylinder trees.

Self-similarity of machine structure at the limit is evident if the machine is displayed in its "dedecorated" form. A portion of the infinite l-digraph at the transition is shown in figure 15 in this form. A decoration of an l-digraph is the insertion of a deterministic chain of states between two states.[62] In a dedecorated l-digraph chains of states are replaced with a single edge labeled with the equivalent symbol sequence. In the figure structures with a chain followed by a single branching have been replaced with a single branching each of whose edges are labeled with the original symbol sequence between the states. The dedecoration makes the self-similarity in the infinite machine structure readily apparent.

Figure 15 Self-similarity of machine structure at cascade limit is shown in the dedecorated l-digraph of M.

The strict regularity in the limit machine structure indicates a uniformity in the underlying computation modeled at a higher level. Indeed, the latter can be inferred from the infinite machine by applying the equivalence class morph reconstruction algorithm to the machine itself.† The result is the non-DFA machine M_c shown in figure 16, where the states in dedecorated M (figure 15) are coalesced into new equivalence classes based on the subtree similarity applied to the sequence of state transitions. The additional feature that must be inferred, once this higher level machine is reconstructed, is the production rule for the edge labels. These describe strings that double in length according to the production $B \rightarrow BB'$, where B is a register variable and B' is the contents of B with the last symbol complemented. The production appends to the register's contents the string B'.

On a state transition the contents of the register are output either directly or as the string B'. The l-digraph edges are labeled accordingly in the figure. On a transition from states signified by

* The total entropy also depends logarithmically on cylinder length.

† The general framework for reconstructing machines at different levels in a computational hierarchy is presented elsewhere.[25]

squares, the register production is performed first and then the transition is made. The machine begins in the start state with a “1” in the register.

Figure 16 Higher level production-rule machine, or stack automaton, M_c that accepts L_c.

M_c accepts the full language L_c produced at the transition including the transient strings with various prefixes. At its core, though, is the simple recursive production $(B \rightarrow BB')$ for the itinerary ω_c of the critical point x^*. We will now explore the structure of this sequence in more detail in order to see just what computational capabilities it requires. We shall demonstrate how and where it fits in the Chomsky hierarchy.

Critical Language and Grammar

Before detailing the formal language properties of the symbol sequences generated at the cascade transition, several definitions of restricted languages are in order. First, of course, is the critical language itself L_c which we take to be the set of all subsequences produced asymptotically by the dynamical system at the cascade transition. M_c is a deterministic acceptor of L_c. Second, the most restricted language, denoted L_1, is the sequence of the itinerary of the map's maximum x^*. That is,

$$L_1 = \left\{ \omega_c : \omega = s_1 s_2 s_3 \cdots \text{ and } f^i(x^*) < x^* \Rightarrow s_i = 0, \text{ otherwise } s_i = 1 \right\}$$

a single sequence. Third, a slight generalization of this, L_2, consists of all length 2^n subwords of ω_c that start at the first symbol

$$L_2 = \left\{ \omega : \omega = s_1 s_2 \cdots s_{2^i} \text{ , i} = 0, 1, 2, \ldots \text{ and } s_j = [w_c]_j \right\}$$

where $[\omega]_k = s_k$ if $\omega = s_1 s_2 s_3 \cdots s_k \cdots$. Finally, we define L_3 to be the set of subsequences of any length that start at the first symbol of ω_c

$$L_3 = \left\{ \omega : \omega = s_1 s_2 \cdots s_i \text{ , i} = 1, 2, 3, \ldots \text{ and } s_j = [w_c]_j \right\}$$

Note that L_c is the further generalization including subsequences that start at any symbol in ω_c

$$L_c = \left\{ \omega_k : \omega_k = s_1 s_2 \cdots s_i \text{ , i} = 1, 2, 3, \ldots \text{ and } s_j = [w_c]_{j+k}, \ k \geq 0 \right\}$$

With these various languages, we can begin to delineate the formal properties of the transition behavior. First, we note that an infinite number of words occur in L_c even though the metric entropy is zero. Additionally, there are an infinite number of inadmissible sequences and so an infinite number of words in the complement language $\bar{L}_c$, i.e. words not in L_c. One consequence is that the transition is not described by a subshift of finite type since there is no finite list of words whose concatenation generates L_c.[63]

Second, in formal language theory “pumping lemmas” are used to prove that certain languages are not in some language class.[5] Typically this is tantamount to demonstrating that particular recurrence or cyclic properties of the class are not obeyed by sufficiently long words in the language in question. Regular languages (RL) are those accepted by DFAs. Using the pumping lemma for regular languages it is easy to demonstrate that $L \in \{L_1, L_2, L_3, L_c\}$ is not regular. This follows from noting that there is no length n such that each word $z \in L$ with $|z| \geq n$ can be broken into three subwords, $z = uvw$ with $|uv| \leq n$, where the middle (nonempty) subword can be repeated arbitrarily many times. That is, sufficiently long strings cannot be decomposed such that $z \in L \Rightarrow uv^i w \in L \; \forall i \geq 0$. In fact, no substrings can be arbitrarily pumped. The lack of such a cyclic property also follows from noting that in M all the states are transient and there are no transient cycles. The observation of this structural property also leads to the conclusion that L_c is also not finitely-described at the next level of the complexity hierarchy: context-free languages (CFL), i.e. those accepted by pushdown automata. This can be established directly using the pumping lemma for context-free languages.

Third, in the structural analysis of M we found states at which the following production is applied: $A \rightarrow AA'$, where $A' = s_0 \cdots \bar{s}_k$ if $A = s_0 \cdots s_k$ and $\bar{s}$ is the complement of s. This production generates L_1 and L_2. It is most concisely expressed as a context-free Lindenmayer system.[64] The general class is called 0L grammars: $G = \{\Sigma, P, \alpha\}$ consisting of the symbol alphabet, production rules, and start string, respectively. This computational model is a class of parallel rewrite automata in which all symbols in a word have the production rules simultaneously applied, with the neighboring symbols playing no role in the selection of which production. The symbol alphabet is $\Sigma = \{0, 1\}$. The production rules P are quite simple$P = \{0 \rightarrow 11, 1 \rightarrow 10\}$ and start with the string $\alpha = \{1\}$. This system generates the infinite sequence L_1 and allowing the choice of when to stop the productions, it generates $L_2 = \{1, 10, 1011, 10111010, \ldots\}$.

Although the L-system model of the transition behavior is quite simple, as a class of models its parallel nature is somewhat inappropriate. L-systems produce both "early" and "late" symbols in a string at every production step; whereas the dynamical system in question produces symbols sequentially. This point is even more obvious when these symbol sequences are considered as sequential measurements. The associated L-system model would imply that the generating process had an infinite memory of past measurements and accessed them arbitrarily quickly. The model class is too powerful.

This can be remedied by converting the 0L-system to its equivalent in the Chomsky hierarchy of sequential computation.[5] The Chomsky equivalent is a restricted indexed context-free grammar $G_c = \{\mathbf{N}, \mathbf{I}, \mathbf{T}, \mathbf{F}, \mathbf{P}, S\}$.[65] A central feature of the indexed grammars is that they are a natural extension of the context-free languages that allow for a limited type of context-sensitivity via indexed productions, while maintaining properties of context-free languages, such as closure and decidability, that are important for compilation. For the limit language the components are defined as follows. $\mathbf{N} = \{S, T\}$ is the set of nonterminal variables with S the start symbol; $\mathbf{I} = \{A, B, C, D, E, F\}$ is the set of intermediate variables; $\mathbf{T} = \{0, 1\}$ is the set of terminal symbols; $\mathbf{P} = \{S \rightarrow Tg, T \rightarrow Tf, T \rightarrow BA, C \rightarrow BB, D \rightarrow BA, E \rightarrow 0, F \rightarrow 1\}$ is the set of productions; and $\mathbf{F} = \{f, g\}$ with $f = [A \rightarrow C, B \rightarrow D]$ and $g = [A \rightarrow E, B \rightarrow F]$ are indexed productions. The grammar just given is in its "normal" form since the variables in the indexed productions $\mathbf{F}$ do not have productions in $\mathbf{P}$. The indexed grammar is restricted in that there are no intermediate variables with productions that produce new indices. The latter occurs only via the $S \rightarrow Tg$ and $T \rightarrow Tf$ productions. Note that once this is no longer used, via the application of $T \rightarrow BA$, no new indices appear.

The above indexed grammar sequentially produces symbols in words from L_2. Two example "left-most" derivations are

$$
\begin{aligned}
(1) \quad & S \to Tg \to BgAg \\
& \to FAg \to 1Ag \to 1E \to 10 \\
(2) \quad & S \to Tg \to Tfg \to BfgAfg \\
& \to DgAfg \to BgAgAfg \to FAgAfg \\
& \to 1AgAfg \to 1EAfg \to 10Afg \\
& \to 10Cg \to 10BgBg \to 10FBg \\
& \to 101Bg \to 101F \to 1011
\end{aligned}
$$

Productions are applied to the leftmost nonterminal in each step. Consequently, the terminal symbols $\{0, 1\}$ are produced sequentially left to right in "temporal" order. In the first line, notice how the indices distribute over the variables produced by the production $T \to BA$. When an indexed production is used an index is consumed: as in $Bg \to F$ in going from the first to the second line above.

All of the languages in the Chomsky hierarchy have dual representations as grammars and as automata. The machine corresponding to an indexed context-free language is the nested stack automaton (NSA).[66] This is a generalization of the pushdown automaton: a finite state control augmented with a last-in first-out memory or stack. An NSA has the additional ability to move into the stack in a read-only mode and to insert a new (nested) stack at the current stack symbol being read. It cannot move higher in the stack until it has finished with the nested stack and removed it. The restricted indexed context-free grammar for L_2 is recognized by the one-way nondeterministic NSA (1NNSA) shown in figure 17. The start state is q. The various actions label the state transition edges. $ denotes the top of the current stack and the cent sign, the current stack bottom. The actions are one of three forms

1. $\alpha \to \beta$, where α and β are patterns of symbols on the top of the current stack;
2. $\alpha \to$ {1, -1}, where the latter indicates moving the head up and down the stack, respectively, upon seeing the pattern α at current stack top.
3. (t,$t) $\to$ (1,$), where t is a symbol read off of the input tape and compared to the symbol at the top of the stack. The '1' indicates that the input head advances to the next symbol on the input tape. The symbol on the stack's top is removed: $t $\to$ $.

In all but one case the actions are in the form of a symbol pattern on the top of the stack leading to a replacement pattern and a stack head motion. The notation on the figure uses a component-wise shorthand. For example, the productions are implemented on the transition labeled ${S,T,T,C,D,E,F} $\to$ ${Tg,Tf,BA,BB,BA,0,1} which is shorthand for the individual transitions: $S $\to$ $Tg, $T $\to$ $Tf, $T $\to$ $BA, $C $\to$ $BA, $D $\to$ $BB, $E $\to$ $0, and $F $\to$ $1. The operation of the 1NNSA mimics the derivations in the indexed grammar. The nondeterminism here means that there exists some set of transitions that will accept words from L_2. L_3 is accepted by the same 1NNSA, but modified to accept when the end of the input string is reached and the previous input has been accepted.

Figure 17 One-way nondeterministic nested stack automaton for limit languages L_2 and L_3.

There are three conclusions to draw from these formal language results. First, it should be emphasized that the particular details in the preceding analysis are not essential. Rather, the most important remark is that the description at this higher level is finite and, indeed, quite small. Despite the infinite DFA complexity, a simple higher level description can be found once the computational model is augmented. Indeed, the deterministic Turing machine program to generate words in the limit language is simple: (i) copy the current string on the tape onto its end and (ii) invert the last bit. The limit language for the cascade transition uses little of the power of the indexed grammars. The latter can recognize, for example, context-sensitive languages. The limit machine is thus exceedingly weak in its implied computational structure. Also, the only nondeterminism in the 1NNSA comes from anticipating the length of the string to accept; a feature that can be replaced to give a deterministic and so less powerful automaton.

Second, it is relatively straightforward to build a continuous-state dynamical system with an embedded universal Turing machine.[‡] With this in mind, and for its own sake, we note that by the above construction the cascade transition does not have universal computation embedded in it. Indeed, it barely aspires to be much more than a context-free grammar. With the formal language analysis we have bounded the complexity at the transition to be greater than regular and context-free languages and no more powerful than indexed context-free. Furthermore, the complexity at this level is measured by a linearly bounded DFA growth rate $c_0 = 3$. These properties leave open the possibility, though, that the language could be a one-way nondeterministic stack automaton (1NSA).[5]

Finally, we demonstrated by an explicit analysis that nontrivial computation, beyond information storage and transmission, arises at a phase transition. One is forced to go beyond DFA models to the higher stack automaton level since the former require an infinite representation. These properties are only hinted at by the infinite correlation length and the slow decay of two-point mutual information at the transition.

Logistic Map

The preceding analysis holds for a wide range of nonlinear systems since it rests only on the symbolic dynamics and the associated probability structure. It is worthwhile, nonetheless, to test it quantitatively on particular examples. This is possible because it rests on a (re)constructive method that applies to any data stream. This section and the next report extensive numerical experiments on two one-dimensional maps. The first is the logistic map, defined shortly, and the second, the piecewise linear tent map.

The logistic map is a map of the unit interval given by

$$x_{n+1} = r x_n (1 - x_n),\ x_0 \in [0,1] \text{ and } r \in [0,4]$$

where the parameter r controls the degree of nonlinearity. $\frac{r}{4}$ is the map's height at its maximum $x^* = \frac{1}{2}$. This is one of the simplest, but nontrivial, nonlinear dynamical systems. It is an extremely rich system about which much is known.[47] It is fair to say, however, that even at the present time there are still a number of unsolved mathematical problems concerning

‡ A two-dimensional map with an embedded 4 symbol, 7 state universal Turing machine[67] was constructed.[68]

the behavior at arbitrary chaotic parameter values. The (generating) measurement partition is $P_{\frac{1}{2}} = \{[0,.5),[.5,1.]\}$.

The machine complexity and information theoretic properties of this system have been reported previously.[30] Figure 18 shows the complexity versus specific entropy for 193 parameter values $r \in [3,4]$. One of the more interesting general features of the complexity-entropy plot is clearly demonstrated by this figure: all of the periodic behavior lies below the critical entropy H_c; and all of the chaotic, above. This is true even if the periodic behavior comes from cascade windows of periodicity $q > 1$ within the chaotic regime at high parameter values. The (H_α, C_α) plot, therefore, captures the essential information processing, i.e. computation and information production, in the period-doubling cascade independent of any explicit system control.

Figure 18 Observed complexity versus specific entropy estimate for the logistic map at 193 parameter values $r \in [3,4]$ within both periodic and chaotic regimes. Estimates on 32-cylinder trees with 16-cylinder subtree machine reconstruction; where feasible.

The lower bound derived in the previous sections applies exactly to the periodic data $(H < H_c)$ and to the band-merging parameter values. The fit to the periodic data is extremely accurate, verifying the linear relationship except for high periods beyond that resolvable at the chosen reconstruction cylinder length. The fit in the chaotic regime is also quite good. (See figure 19.) The data are systematically lower (~2%) in entropy due to the use of the topological entropy in the analysis. The measured critical entropy H_c and complexity C'' at the transition were 0.28 and 4.6, respectively.

Figure 19 Fit of logistic map periodic and chaotic data to corresponding functional forms. The data is from the periodicity 1 band-merging cascade and also includes all of the periodic data found in the preceding figure. The theoretical curves $C_0(H_0)$ are shown as solid lines.

Tent Map

The second numerical example, the tent map, is in some ways substantially simpler than the logistic map. It is given by

$$x_{n+1} = \begin{cases} a x_n & x_n \leq \frac{1}{2} \\ a(1 - x_n) & x_n > \frac{1}{2} \end{cases}$$

where the parameter a controls the height $\left(= \frac{a}{2}\right)$ of the map at the maximum $x^* = \frac{1}{2}$. The main simplicity is that there is no period-doubling cascade and, for that matter, there are no stable periodic orbits, except at the origin for $a < 1$. There is instead only a periodicity $q = 1$ chaotic band-merging cascade that springs from x^* at $a = 1$.

The piecewise linearity also lends itself to further analysis of the dynamics. Since the map has the same slope everywhere, the Lyapunov exponent λ, topological, metric, and Renyi specific entropies are all equal and given by the slope $\lambda = h_\alpha = \log_2 a$. We can simply refer to these as the specific entropy. From this, we deduce that, since $h_\alpha = 2^{-m}$ for $2^m \rightarrow 2^{m-1}$ band-mergings, the parameter values there are

$$a_{2^m \rightarrow 2^{m-1}} = 2^{2^{-m}}$$

For $L \geq 2^m$ the complexity is given by the band-merging period. And this, in turn, is given by the number of bands. Thus, we have $C_\alpha = -\log_2 h_\alpha$ or

$$C_\alpha = -\log_2 \log_2 a$$

as a lower bound for $a > 1$ and $L > 2^m$ at an m-order band merging.

Since there is no stable periodic behavior, other than period one, there is a forbidden region in the complexity-entropy plot below the critical entropy. The system cannot exist at finite "temperatures" below H_c, except at absolute zero $H_c = 0$.

Figure 20 gives the complexity-entropy plot for 200 parameter values $a \in [1, 2]$. There is a good deal of structure in this plot beyond the simple band-merging lower bounds we have concentrated on. Near each band-merging complexity-entropy point, there is a slanted cluster of points. These are associated with families of parameter values at which the iterates $f^n(x^*)$ are asymptotically periodic of various periods. We shall discuss this structure elsewhere, except to note here that it also appears in the logistic map, but is substantially clearer in this example.

Figure 20 Tent map complexity versus entropy at 200 parameter values $a \in [1, 2]$. The quantities were estimated with 20-cylinder reconstruction on 40-cylinder trees; where feasible.

Figure 21 shows band-merging data estimated from 16- and 20- cylinders along with the appropriate theoretical curves for those and in the thermodynamic limit ($L = 256$).

Figure 21 Effect of cylinder length. Tent map data at 16- and 20-cylinders (triangle and square tokens, respectively) along with theoretical curves $C_0(H_0)$ for the same and in the thermodynamic limit ($L = 256$).

From the two numerical examples it is clear that the theory quite accurately predicts the complexity-entropy dependence. It can be easily extended in several directions. Most notably, the families of Misiurewicz parameters associated with unstable asymptotically periodic maxima can be completely analyzed. And this appears to give some insight into the general problem of the measure of parameter values where iterates of the maximum are asymptotically aperiodic. Additionally, the computational analysis is being applied to transitions to chaos via intermittency and via frequency-locking.

Computation at Phase Transitions

To obtain a detailed understanding of the computational structure of a phase transition, we have analyzed one example of a self-similar family of attractors. The period-doubling cascade is just one of many routes to chaos. The entire family is of nominal interest, providing a rather complete analysis of a phase transition and how statistical mechanics applies. More importantly, for general phase transitions the approach developed here indicates a type of superuniversality that is based only on the intrinsic information processing performed by a dynamical or, for that matter, physical system. This information processing consists of conventional communication theoretic quantities, that is the storage and transmission of information, and the computational aspects, most clearly represented by the detailed structure and formal language properties of reconstructed ϵ-machines.

By analyzing in some detail a particular class of nonlinear systems, we have attempted to strengthen the conjecture that it is at phase transitions where high level computation occurs. Application to other examples, such as the phase transition in the 2D Ising spin system and cellular automata and lattice dynamical systems generally, will go even further toward establishing this general picture. These applications will be reported elsewhere. Nonetheless, it is clear that computational ideas provide a new set of tools for investigating the physics of phase transitions. The central conclusion is that via reconstruction they can be moved out of the realm of mathematics and theoretical computer science and applied to the scientific study and engineering of complex processes.

The association of high level computation and phase transitions is not made in isolation. Indeed, we should mention some early work addressing similar questions. Type IV cellular automata (CA) were conjectured by Wolfram to support nontrivial and perhaps universal computation.[69] These CA exhibit long-lived transients and propagating structures out of which elementary computations can be constructed. The first author and Norman Packard of the University of Illinois conjectured some years ago that type IV behavior was generated by CA on bifurcation sets in the discretized space of all CA rules. This was suggested by studies of bifurcations in a continuous-state lattice dynamical system as a function of a nonlinearity parameter.[70] The continuous local states were discretized to give CA with varying numbers of states. By comparing across a range of state-discretization and nonlinearity parameter, the CA bifurcation sets were found to be associated with bifurcations in the continuous-state lattice system. More recent work by Chris Langton of Los Alamos National Laboratory has confirmed this in substantially more detail via Monte Carlo sampling of CA rule space. This work uses mutual information, not machine complexity, measures of the behavior. As pointed out above, there is an inequality relating these measures. Mutual information versus entropy density plots for hundreds of CA rules reveal a phase transition structure similar to that shown in the complexity-entropy diagram of figure 18. Seen in this light, the present paper augments these experimental results with an analytic demonstration of what appears to be a very general organization of the space of dynamical systems, whether discrete or continuous.

Complexity of Critical States

Recall that the Wold-Kolmogorov spectral decomposition says the spectrum of a stationary signal has three components.[2,3,4] The first is a singular measure consisting of δ-functions. This describes periodic behavior. The second component is associated with an absolutely continuous invariant measure and so broadband power spectra. The final component is unspecified and typically ignored. From the preceding investigation, though, we can make a comment and a conjecture. The comment is that finite stochastic DFA ϵ-machines capture the first two components of the decomposition: $\mathbf{P_t}$ and $\mathbf{B_t}$, respectively. The conjecture, and perhaps more interesting remark, is that the third component appears to be associated with higher levels in the computational hierarchy.

This conjecture can be compared to recent discussion of ergodic theory. Ornstein suggested that most "chaotic systems that arise naturally are abstractly the same as $\mathbf{B_t}$".[1] We can now see in what sense this can be true. If it is only at accumulation points of bifurcations that infinite

DFA machines occur, then in the space of all dynamical systems the dimensionality of the set of such systems will be reduced and so the set's measure will be zero. From this viewpoint, high complexity (non-$\mathbf{B_t}$) systems would be rare. We can also see how the conjecture can be false. If there is some constraint restricting the space of systems in which we are interested, then with respect to that space infinite machines might have positive measure and so be likely. Systems, such as those found in biology, that survive by adaptively modifying themselves to better model and forecast their environment would tend to exhibit high levels of complexity. Within the space of successful adaptive systems, high complexity presumably is quite probable. Another sense in which Ornstein's conjecture is too simplified is that the Bernoulli shift is computationally simple. It is equivalent, in one representation, to the piecewise linear Baker's transformation of the torus. In contrast, most "natural" physical systems are modeled with smooth (non-piecewise-linear) nonlinearities. The associated physical properties contribute substantially to a system's ability to generate complex behavior independent of the complexity of boundary and initial conditions.[71] Physical systems governed by a chaotic piecewise linear dynamic simply excavate microscopic fluctuations, amplifying them to determine macroscopic behavior.[72] This information transmission across scales is computationally trivial. It is not the central property of complex behavior and structure.

We have seen that away from the cascade phase transition it is only at band-merging parameters where chaotic behavior can be factored into periodic and Bernoulli components. A similar decomposition of ϵ-machines into periodic and finite stochastic components occurs at Misiurewicz parameters, where $f^n(x^*)$ is asymptotically periodic and unstable and the invariant measure is absolutely continuous. But these parameter values are countable. Furthermore, the measure of chaotic parameter values as one approaches a Misiurewicz value is positive and so is uncountable.[73] Typical parameters in this set appear to be characterized by aperiodic $f^n(x^*)$ that are in a Cantor set not containing x^*. Such a case is modeled by infinite DFA ϵ-machines. Taken together these indicate that a large fraction of "naturally arising" chaotic systems are isomorphic neither to $\mathbf{B_t}$ nor to $\mathbf{B_t} \otimes \mathbf{P_t}$.

Computational ergodic theory, the application of computational analysis in ergodic theory, would appear to be a useful direction in which to search for rigorous refined classifications of complex behavior. This suggests, for example, the use of DFA and SA complexity, and other higher forms, as invariants to distinguish further the behavior of dynamical systems, such as K-flows and zero-entropy flows. For example, although described by a minimal DFA with a single state, the inequivalence of the binary $\mathbf{B_2}\left(\frac{1}{2},\frac{1}{2}\right)$ and ternary $\mathbf{B_3}\left(\frac{1}{3},\frac{1}{3},\frac{1}{3}\right)$ Bernoulli shifts follows from the fact that they are not structurally equivalent. Not only do the entropies differ, but the machine for the first has two edges; for the second, three. Restating this in entropy-complexity notation

$$\begin{aligned} \text{although} \quad & C_\alpha(\mathbf{B_2}) = C_\alpha(\mathbf{B_3}) = 0, \\ & h_\alpha(\mathbf{B_2}) \neq h_\alpha(\mathbf{B_3}) \\ \text{and} \quad & C_\alpha^e(\mathbf{B_2}) \neq C_\alpha^e(\mathbf{B_3}) \end{aligned}$$

The full entropy-complexity plane appears as a useful first step toward a more complete classification. Recall the (H_α, C_α) plots for the logistic and tent maps. (See figures 18 and 20.) Similar types of structural distinctions and new invariants will play a central role in computational ergodic theory.

But what does this have to say about physical systems? In what sense can a turbulent fluid, a noisy Josephson junction, or a quantum field, be said to perform a computation? The answer is that while computational aspects appear in almost any process, since we can always estimate some low level ϵ-machine, only nontrivial computation occurs in physical systems on the order-disorder border. Additionally these systems have very special phase-transition-like, or “critical”, subspaces. ϵ-machine theory gives a much more refined description of such critical behavior than that currently provided by statistical mechanics. Indeed, the well-known phase transitions should be re-examined in this light. In addition to a more detailed structural theory of the dynamic mechanisms responsible for phase transitions, such studies will give an improved understanding of the macroscopic thermodynamic properties required for computation.

Computers are, in this view, physical systems designed to be in a critical state. They are constructed to support arbitrarily long time correlations within certain macroscopic “computational” degrees of freedom. This is achieved by decoupling these degrees of freedom from error-producing heat bath degrees of freedom. Computers are physical systems designed to be in continual phase transition within entropic-disordered environments. From the latter they derive the driving force that moves computations forward. But, at the same time, they must shield the computation from environmentally induced fluctuations.

As already emphasized, the general measure of complexity introduced at the beginning is not limited to stochastic DFAs and SAs, but applies in principle to any computational level or, indeed, to any modeling domain where a “symmetry” can be factored out of data. In particular, this can be done hierarchically as we have shown in the analysis of the computational properties of the cascade critical machine. In fact, a general hierarchical reconstruction is available.[25] The abstract definition of complexity applies to all levels of the Chomsky hierarchy where each computation level represents, in a concrete way, a class of symmetries with respect to which observed data is to be “expanded” or modeled. This notion of complexity is also not restricted to the Chomsky hierarchy. It can be applied, for example, to spatially-extended or network dynamical systems. Since these are computationally equivalent to parallel and distributed machines, respectively, ϵ-machine reconstruction suggests a constructive approach to parallel computation theory. We hope to return to these applications in the near future.

Acknowledgements

The authors have benefited from discussions with Charles Bennett, Eric Friedman, Chris Langton, Steve Omohundro, Norman Packard, Jim Propp, Jorma Rissanen, and Terry Speed. They are grateful to Professor Carson Jeffries for his continuing support. The authors thank the organizers of the Santa Fe Institute Workshop on “Complexity, Entropy, and Physics of Information” (May 1989) for the opportunity to present this work, which was supported by ONR contract N00014-86-K-0154.

Bibliography

1. D. S. Ornstein. Ergodic theory, randomness, and chaos. *Science*, 243:182, 1989.
2. A. N. Kolmogorov. Interpolation and extrapolation of stationary time series. In T. Kailath, editor, *Linear Least-Squares Estimation*, page 52. Dowden, Hutchinson, and Ross, New York, 1977.
3. A. N. Kolmogorov. Stationary sequences in hilbert space. In T. Kailath, editor, *Linear Least-Squares Estimation*, page 66. Dowden, Hutchinson, and Ross, New York, 1977.
4. H. O. A. Wold. *A Study in the Analysis of Stationary Times Series*. Almqvist and Wiksell Forlag, Stockholm, 1954.
5. J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, Reading, 1979.
6. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, New York, 1979.
7. A. N. Kolmogorov. A new metric invariant of transient dynamical systems and automorphisms in Lebesgue spaces. *Dokl. Akad. Nauk. SSSR*, 119:861, 1958. (Russian) Math. Rev. vol. 21, no. 2035a.
8. A. N. Kolmogorov. Entropy per unit time as a metric invariant of automorphisms. *Dokl. Akad. Nauk. SSSR*, 124:754, 1959. (Russian) Math. Rev. vol. 21, no. 2035b.
9. Ja. G. Sinai. On the notion of entropy of a dynamical system. *Dokl. Akad. Nauk. SSSR*, 124:768, 1959.
10. C. E. Shannon and W. Weaver. *The Mathematical Theory of Communication*. University of Illinois Press, Champaign-Urbana, 1962.
11. J. P. Crutchfield. Inferring the dynamic, quantifying physical complexity. In N. B. Abraham, A. M. Albano, A. Passamante, and P. E. Rapp, editors, *Measures of Complexity and Chaos*, page 327, New York, 1990. Plenum Press.
12. J. P. Crutchfield and B. S. McNamara. Equations of motion from a data series. *Complex Systems*, 1:417 – 452, 1987.
13. J. G. Kemeny. The use of simplicity in induction. *Phil. Rev.*, 62:391, 1953.
14. J. Rissanen. Modeling by shortest data description. *Automatica*, 14:462, 1978.
15. J. Rissanen. Universal coding, information, prediction, and estimation. *IEEE Trans. Info. Th.*, IT-30:629, 1984.
16. G. Chaitin. On the length of programs for computing finite binary sequences. *J. ACM*, 13:145, 1966.
17. G. Chaitin. Randomness and mathematical proof. *Sci. Am.*, May:47, 1975.
18. A. N. Kolmogorov. Three approaches to the concept of the amount of information. *Prob. Info. Trans.*, 1:1, 1965.
19. A. N. Kolmogorov. Combinatorial foundations of information theory and the calculus of probabilities. *Russ. Math. Surveys*, 38:29, 1983.
20. P. Martin-Lof. The definition of random sequences. *Info. Control*, 9:602, 1966.
21. R. J. Solomonoff. A formal theory of inductive control. *Info. Control*, 7:224, 1964.

22. J. P. Crutchfield and N. H. Packard. Symbolic dynamics of one-dimensional maps: Entropies, finite precision, and noise. *Intl. J. Theo. Phys.*, 21:433, 1982.
23. A. A. Brudno. Entropy and the complexity of the trajectories of a dynamical system. *Trans. Moscow Math. Soc.*, 44:127, 1983.
24. J. P. Crutchfield. Compressing chaos. in preparation, 1989.
25. J. P. Crutchfield. Reconstructing language hierarchies. In H. A. Atmanspracher and H. Scheingraber, editors, *Information Dynamics*, pages 45 – 60, New York, 1991. Plenum.
26. C. H. Bennett. Thermodynamics of computation - a review. *Intl. J. Theo. Phys.*, 21:905, 1982.
27. S. Wolfram. Intractability and undecidability in theoretical physics. In S. Wolfram, editor, *Theory and Applications of Cellular Automata*. World Scientific, Singapore, 1986.
28. W. H. Zurek. Thermodynamic cost of computation, algorithmic complexity, and the information metric. preprint, 1989.
29. J. P. Crutchfield. Information and its metric. In L. Lam and H. C. Morris, editors, *Nonlinear Structures in Physical Systems - Pattern Formation, Chaos and Waves*, page 119, New York, 1990. Springer-Verlag.
30. J. P. Crutchfield and K. Young. Inferring statistical complexity. *Phys. Rev. Let.*, 63:105–108, 1989.
31. J. P. Crutchfield and N. H. Packard. Symbolic dynamics of noisy chaos. *Physica*, 7D:201 – 223, 1983.
32. S. Wolfram. Computation theory of cellular automata. *Comm. Math. Phys.*, 96:15, 1984.
33. R. Shaw. *The Dripping Faucet as a Model Chaotic System*. Aerial Press, Santa Cruz, California, 1984.
34. C. H. Bennett. On the nature and origin of complexity in discrete, homogeneous locally-interacting systems. *Found. Phys.*, 16:585–592, 1986.
35. C. P. Bachas and B. A. Huberman. Complexity and relaxation of hierarchical structures. *Phys. Rev. Let.*, 57:1965, 1986.
36. P. Grassberger. Toward a quantitative theory of self-generated complexity. *Intl. J. Theo. Phys.*, 25:907, 1986.
37. H. Pagels and S. Lloyd. Complexity as thermodynamic depth. *Ann. Phys.*, 188:186, 1988.
38. N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw. Geometry from a time series. *Phys. Rev. Let.*, 45:712, 1980.
39. J. P. Crutchfield. *Noisy Chaos*. PhD thesis, University of California, Santa Cruz, 1983. Published by University Microfilms Intl, Ann Arbor, Michigan.
40. D. M. Cvetkovic, M. Doob, and H. Sachs. *Spectra of Graphs*. Academic Press, New York, 1980.
41. R. Fischer. Sofic systems and graphs. *Monastsh. Math*, 80:179, 1975.
42. N. Chomsky. Three models for the description of language. *IRE Trans. Info. Th.*, 2:113, 1956.
43. A. Renyi. On the dimension and entropy of probability distributions. *Acta Math. Hung.*, 10:193, 1959.
44. K. Huang. *Statistical Mechanics*. J. Wiley and Sons, New York, 1963.

45. E. T. Jaynes. Where do we stand on maximum entropy? In E. T. Jaynes, editor, *Essays on Probability, Statistics, and Statistical Physics,* page 210. Reidel, London, 1983.
46. J. P. Crutchfield and N. H. Packard. Noise scaling of symbolic dynamics entropies. In H. Haken, editor, *Evolution of Order and Chaos,* pages 215–227, Berlin, 1982. Springer-Verlag.
47. P. Collet and J.-P. Eckmann. *Maps of the Unit Interval as Dynamical Systems*. Birkhauser, Berlin, 1980.
48. J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Springer-Verlag, New York, 1983.
49. M. J. Feigenbaum. Universal behavior in nonlinear systems. *Physica*, 7D:16, 1983.
50. J. P. Crutchfield, J. D. Farmer, N. H. Packard, R. S. Shaw, G. Jones, and R. Donnelly. Power spectral analysis of a dynamical system. *Phys. Lett.*, 76A:1, 1980.
51. J. P. Crutchfield and B. A. Huberman. Fluctuations and the onset of chaos. *Phys. Lett.*, 77A:407, 1980.
52. M. J. Feigenbaum. The universal metric properties of nonlinear transformations. *J. Stat. Phys.*, 21:669, 1979.
53. P. J. Myrberg. Iteration reellen polynome zweiten grades iii. *Ann. Akad. Sc. Fennicae A, I*, 336/3:1, 1963.
54. N. Metropolis, M. L. Stein, and P. R. Stein. On finite limit sets for transformations on the interval. *J. Combin. Th.*, 15A:25, 1973.
55. J. Milnor and W. Thurston. On iterated maps of the interval. *Springer Lecture Notes*, 1342:465–563, 1988.
56. J. Guckenheimer. Sensitive dependence to initial conditions for one-dimensional maps. *Comm. Math. Phys.*, 70:133, 1979.
57. P. Collet, J. P. Crutchfield, and J. P. Eckmann. Computing the topological entropy of maps. *Comm. Math. Phys.*, 88:257, 1983.
58. M. Misiurewicz. Absolutely continuous measures for certain maps of an interval. Preprint IHES M-79-293, 1979.
59. R. E. Blahut. *Principles and Practice of Information Theory*. Addison-Wesley, Reading, Massachusetts, 1987.
60. J. P. Crutchfield, J. D. Farmer, and B. A. Huberman. Fluctuations and simple chaotic dynamics. *Phys. Rep.*, 92:45, 1982.
61. N. H. Packard. *Measurements of Chaos in the Presence of Noise*. PhD thesis, University of California, Santa Cruz, 1982.
62. H. N. V. Temperley. *Graph Theory and Applications*. Halsted Press, New York, 1981.
63. V. M. Alekseyev and M. V. Jacobson. Symbolic dynamics. *Phys. Rep.*, 25:287, 1981.
64. A. Lindenmayer and P. Prusinkiewicz. Developmental models of multicellular organisms: A computer graphics perspective. In C. G. Langton, editor, *Artificial Life*, page 221, Redwood City, California, 1989. Addison-Wesley.

65. A. V. Aho. Indexed grammars - an extension of context-free grammars. *J. Assoc. Comp. Mach.*, 15:647, 1968.
66. A. V. Aho. Nested stack automata. *J. Assoc. Comp. Mach.*, 16:383, 1969.
67. M. Minsky. *Computation: Finite and Infinite Machines*. Prentice-Hall, Englewood Cliffs, New Jersey, 1967.
68. J. P. Crutchfield. Turing dynamical systems. unpublished, 1987.
69. S. Wolfram. *Theory and Applications of Cellular Automata*. World Scientific Publishers, Singapore, 1986.
70. J. P. Crutchfield and K. Kaneko. Phenomenology of spatio-temporal chaos. In Hao Bai-lin, editor, *Directions in Chaos*, page 272. World Scientific Publishers, Singapore, 1987.
71. J. P. Crutchfield and J. E. Hanson. Chaotic arithmetic automata. *in preparation*, 1989.
72. R. Shaw. Strange attractors, chaotic behavior, and information flow. *Z. Naturforsh.*, 36a:80, 1981.
73. M. V. Jacobson. Absolutely continuous invariant measures for one-parameter families of one-dimensional maps. *Comm. Math. Phys.*, 81:39, 1981.