in **Modeling Complex Phenomena**, L. Lam and V. Naroditsky, editors,
Springer, Berlin (1992) 66 — 101.

SFI 91–09–035

KNOWLEDGE AND MEANING ... CHAOS AND COMPLEXITY

J. P. Crutchfield

What are models good for? Taking a largely pedagogical view, the following essay discusses the semantics of measurement and the uses to which an observer can put knowledge of a process's structure. To do this in as concrete a way as possible, it first reviews the reconstruction of probabilistic finite automata from time series of stochastic and chaotic processes. It investigates the convergence of an observer's knowledge of the process's state; assuming that the process is in, and the observer also uses for internal representation, that model class. The conventional notions of phase and phase-locking are extended beyond periodic behavior to include deterministic chaotic processes. The meaning of individual measurements of an unpredictable process is then defined in terms of the computational structure in a model that an observer built.

1. Experimental Epistemology

> *The grammar of a language can be viewed as a theory of the structure of this language. Any scientific theory is based on a certain finite set of observations and, by establishing general laws stated in terms of certain hypothetical constructs, it attempts to account for these observations, to show how they are interrelated and to predict an indefinite number of new phenomena.*
>
> *General linguistic theory can be viewed as a metatheory which is concerned with the problem of how to choose such a grammar in the case of each particular language on the basis of a finite corpus of sentences.*
>
> N. Chomsky[1]

There is a deep internal conflict in the scientific description of a classical universe. The first aspect of this conflict is the conviction that nature is, in a sense, a deductive system: pushing

Physics Department, University of California, Berkeley, California 94720, USA; For Internet use chaos@gojira.berkeley.edu or try chaos@UCBphysi.BITNET.

forward through time according to the dictates of microscopic deterministic equations of motion. The second, polar aspect is that a scientist, or any physical subsystem of the universe which attempts to model other portions of the universe, is an inferential system: destined always to estimate and approximate from finite data and, for better or worse, project unverifiable structure onto the local environment. The conflict is simply that the logical types of these concomitant views do not match. Presumably due to the difficulty represented by this tension, the premises underlying scientific description leave unanswered the primary questions: Of what does knowledge consist? And, from what does meaning emerge?

Answers to questions like these usually fall under the purview of the philosophy of science.[2] Unfortunately, investigations along these lines tend to be more concerned with developing an internally-consistent language for the activity of science than with testing concrete hypotheses about the structure of knowledge and the mechanisms governing the emergence of meaning in physical systems. Retreating some distance from the general setting of epistemology and ontology, the following attempts to address these questions in the philosophically restricted domain of nonlinear modeling. The presentation is narrowed, in fact, even further to building stochastic automata from time series of finite resolution measurements of nonlinear stationary processes. Nonetheless, I think that some progress is made by introducing these issues into modeling complex phenomena and that the results reflect back positively on the larger problems.

Why are semantics and epistemology of interest to the study of chaos and complexity? Surely such philosophical questions have arisen before for mechanical systems. The early days of information theory, cybernetics, control theory, and linguistics, come to mind. Perhaps the most direct early approach is found in MacKay's collection **Information, Meaning and Mechanism**.[3] But I find the questions especially compelling, and even a bit poignant, in light of the recent developments in dynamics. We now know of easily-expressed nonlinear systems to whose temporal behavior we ascribe substantial complication. In stark opposition to Laplacian determinism, even if we know their governing equations of motion, they can remain irreducibly unpredictable. The behavior's complication and its unpredictability contrasts sharply with the simplicity of the equations of motion.[*] And so, in the microcosm of nonlinear modeling a similar epistemological tension arises: the deterministic evolution of simple systems gives rise to complex phenomena in which an observer is to find regularity.

I should also mention another premise guiding these concerns. A goal here is to aid investigations in the dynamics of evolution. The premise is that the questions of experimental epistemology and the emergence of semantics must be addressed at the very low level of

[*] This is also testimony to how much we still do not understand about how genuine complexity is generated by dynamical systems such as the oft-studied logistic and circle maps. The engineering suggestions[4–7] that there exist physically plausible dynamical systems implementing Turing machines, though they appear to address the problem of physical complexity, are irrelevant as they skirt the issue of its emergence through time evolution.

the following. Additionally, these issues must be tackled head on if we are to understand genuine learning and intrinsic self-organization beyond mere statistical parameter estimation and pattern stabilization, with which they respectively are often confused.

The epistemological problem of nonlinear modeling is: Have we discovered something in our data or have we projected the new-found structure onto it?* This was the main lesson of attempting to reconstruct equations of motion from a time series:[9] When it works, it works; when it doesn't, you don't know what to do; and in both cases it is ambiguous what you have learned. Even though data was generated by well-behaved, smooth dynamical systems, there was an extreme sensitivity to the assumed model class that completely swamped "model order estimation". Worse still there was no *a priori* way to select the class appropriate to the process.† This should be contrasted with what is probably one of the more important practical results in statistical modeling: within a model class a procedure exists to find, given a finite amount of data, an optimal model that balances prediction error against model complexity.[10] Despite representations to the contrary, this "model order estimation" procedure does not address issues of class inappropriateness and what to do when confronted with failure.

There appears to be a way out of the model class discovery dilemma. The answer that hierarchical machine reconstruction gives is to start at the lowest level of representation, the given discrete data, and to build adaptively a series of models within a series of model classes of increasing computational capability until a finite causal model is found.[11] Within each level there is a model-order-estimation inference of optimal models, as just indicated. And there is an induction from a series of approximate models within a lower "inappropriate" class to the next higher model class. This additional inductive step, beyond the standard model order estimation procedure, is the price to be paid for formalizing adaptive modeling.

The success at each stage in hierarchical reconstruction is controlled by the amount of given data, since this puts an upper bound on statistical accuracy, and an error threshold, which is largely determined by the observer's available computational resources. The goal is to find a finite causal model of minimal size and prediction error while maximizing the extraction of information from the given data.

Within this hierarchical view the epistemological problem of nonlinear modeling can be crudely summarized as the dichotomy between engineering and science. As long as a representation is effective for a task, an engineer does not care what it implies about the underlying mechanisms; to the scientist though the implication makes all the difference in the world. The engineer certainly is concerned with minimizing implementation cost, such as representation size, compute time, and storage; but the scientist presumes, at least, to be

---

* In philosophy such considerations currently appear under the rubric of "intentionality".[8] I do not have in mind, however, verbal intercourse, but rather the intentionality arising in the dialogue, and sometimes monologue, between science and nature. One of the best examples of this is the first "law" of thermodynamics: energy conservation.

† In artificial intelligence this is referred to as the "representation problem".

focused on what the model means *vis á vis* natural laws. The engineering view of science is that it is mere data compression; scientists seem to be motivated by more than this.

From these general considerations a number of pathways lead to interesting and basic problems. The following will address the questions of what individual measurements mean to the observer who has made them and how an observer comes to know the state of a process. There are two main themes: knowledge convergence and measurement semantics. While these are relatively simple compared to the problems of an experimental epistemology, they do shed some light on where to begin and so complement direct studies of the emergence of intrinsic computation. Before the two themes are considered I quickly review stochastic automata and their reconstruction. Indeed, to complete the semantic chain of meaning and to suggest how it can emerge, I must say first where models come from. Then, after the main topics are presented, the discussion ends with some general remarks on causality and complexity.

2. COMPUTATIONAL MECHANICS

The overall goal is to infer from a series of measurements of a process a model of the generating mechanism. Additionally, the model is to indicate the process's computational structure. This refers not only to its statistical properties, such as the decay of correlation, but also to the amount of memory it contains and, for example, whether or not it is capable of producing the digit string of (say) $\sqrt{3}$. The extraction of these properties from the model determines the utility of the model class beyond mere temporal prediction of the process's behavior.

The first subsection starts off by defining the character of the data from which the model is built. The next two subsections then review machine reconstruction and the statistical mechanics of the inferred stochastic machines. The section closes with some comments on complexity and model minimality.

2.1 The Measurement Channel

The universe of discourse for nonlinear modeling consists of a process P, the measuring apparatus $\mathcal{I}$, and the modeler itself. Their relationships and components are shown schematically in Fig. 1. The goal is for the modeler, by taking advantage of its available resources, to make the "best" representation of the nonlinear process.

The process P, the object of the modeler's ultimate attention, is the unknown, but hopefully knowable, variable in this picture. And so there is little to say, except that it can be viewed as governed by stochastic evolution equations

$$\vec{X}_{t+\Delta t} = \vec{F}\left(\vec{X}_t, \vec{\xi}_t, t\right) \tag{1}$$

where $\vec{X}_t$ is the configuration at time t, $\vec{\xi}_t$ some noise process, and $\vec{F}$ the governing equations of motion. The following discussion also will have occasion to refer to the process's

measure $\mu\left(\vec{X}\right)$ on its configuration space and the entropy rate $h_\mu\left(\vec{X}\right)$ at which it produces information.

The measuring apparatus is a transducer that maps $\vec{X}_t$ to some accessible states of an instrument $\mathcal{I}$. This instrument has a number of characteristics, most of which should be under the modeler's control. The primary interaction between the instrument and the process is through the measurement space $\mathcal{R}^{\mathrm{D}}$ which is a projection $\mathcal{P}$ of $\vec{X}_t$ onto (say) a Euclidean space whose dimension is given by the number D of experimental probes. An instrument that distinguishes the projected states to within a resolution ϵ partitions the measurement space into a set $\Pi_\epsilon(\mathrm{D}) = \left\{\pi_i : \pi_i \subset \mathcal{R}^{\mathrm{D}}, i = 0, \ldots \epsilon^{-\mathrm{D}}\right\}$ of cells. Each cell π_i is the equivalence class of projected states that are indistinguishable using that instrument. The instrument represents the event of finding $\mathcal{P}\left(\vec{X}_t\right) \in \pi_i$ by the cell's label i. With neither loss of generality nor information, these indices are then encoded into a time-serial binary code. As each measurement is made its code is output into the data stream. In this way, a time series of measurements made by the instrument becomes a binary string, the data stream $\mathbf{s}$ available to the modeler. This is a discretized set of symbols $\mathbf{s} = \ldots s_{-4}s_{-3}s_{-2}s_{-1}s_0s_1s_2s_3s_4\ldots$ where in a single measurement made by the modeler the instrument returns a symbol $s_t \in \mathbf{A}$ in an alphabet $\mathbf{A}$ at time index $t \in \mathbf{Z}$. Here we take a binary alphabet $\mathbf{A} = \{0, 1\}$.

Beyond the instrument, one must consider what can and should be done with information in the data stream. Acquisition of, processing, and inferring from the measurement sequence are the functions of the modeler. The modeler is essentially defined in terms of its available inference resources. These are dominated by storage capacity and computational power, but certainly include the inference method's efficacy, for example. Delineating these resources constitutes the barest outline of an observer that builds models. Although the following discussion does not require further development at this abstract a level, it is useful to keep in mind since particular choices for these elements will be presented.

The modeler is presented with $\mathbf{s}$, the bit string, some properties of which were just given. The modeler's concern is to go from it to a useful representation. To do this the modeler needs a notion of the process's effective state and its effective equations of motion. Having built a model consisting of these two components, any residual error or deviation from the behavior described by the model can be used to estimate the process's effective noise level. This level is determined by the amount of data "unexplained" by the inferred model. It should be clear when said this way that the noise level and the model's sophistication depend directly on the data and on the modeler's resources. Finally, the modeler may have access to experimental control parameters. And these can be used to aid in obtaining different data streams useful in improving the model by (say) concentrating on behavior where the effective noise level is highest.

The central problem of nonlinear modeling now can be stated. Given an instrument, some number of measurements, and fixed *finite* inference resources, how much computational structure in the underlying process can be extracted?

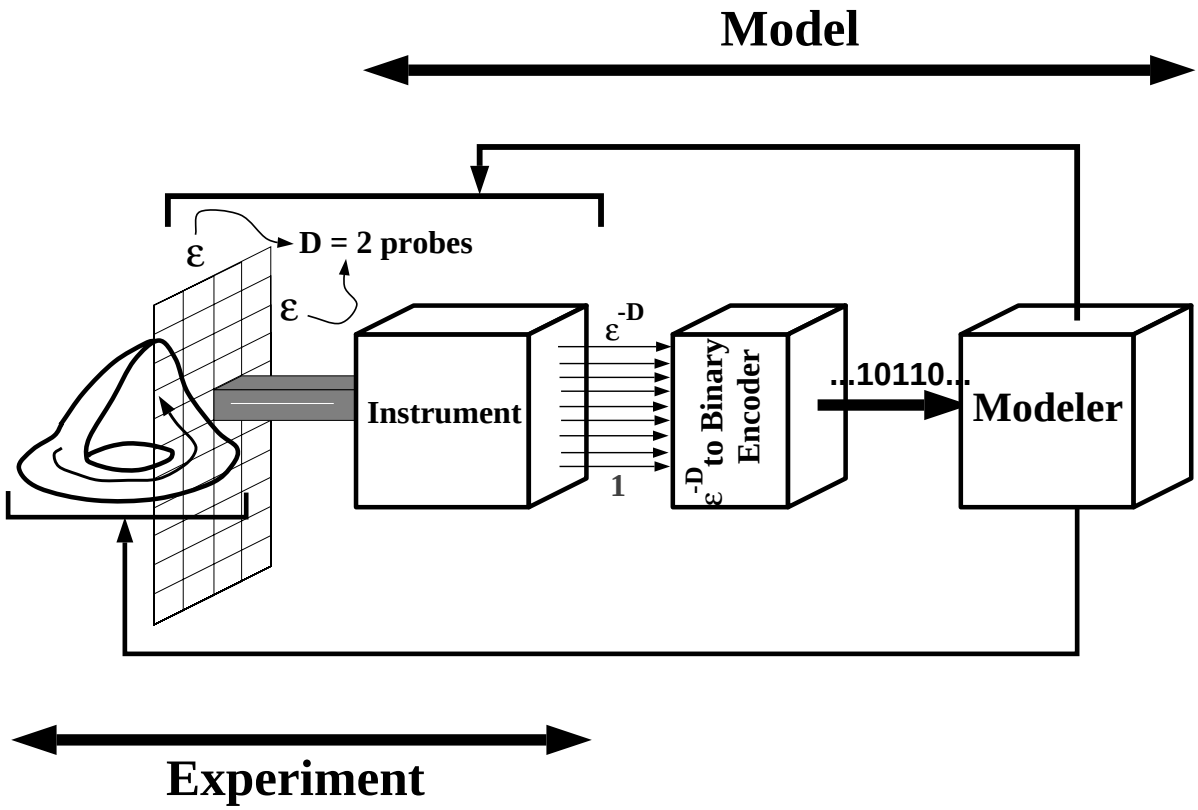

Fig. 1 The Big Channel. The flow of information (measurements) on the shortest time scales is from the left, from the underlying process, to the right toward the modeler. The latter's task is to build the "best" representation given the available data set and computational resources. On longer time scales the modeler may modify the measuring apparatus and vary the experimental controls on the process. These actions are represented by the left-going arrows. Notice that from the modeler's perspective there is a region of ambiguity between the model and the experiment. The model includes the measuring apparatus since it instantiates many of the modeler's biases toward what is worth observing. But the experiment also includes the measuring apparatus since it couples to the process. Additionally, the apparatus is itself a physical device with its own internal dynamics of which the modeler may be unaware or incapable of controlling.

Before pursuing this goal directly it will be helpful to point out several limitations imposed by the data or, rather, the moder's interpretation of it.

In describing the data stream's character it was emphasized that the individual measurements are only indirect representations of the process's state. If the modeler interprets the measurements as the process's state, then it is unwittingly forced into a class of computationally less powerful representations. These consist of finite Markov chains with states in **A** or in some arbitrarily selected state alphabet. This will become clearer through several examples used later on. The point is that it is important to not over-interpret the measurements' content at this early stage of inference as this might limit the quality of the resulting models.

The instrument itself obviously constrains the observer's ability to extract regularity from the data stream and so it directly affects the model's utility. The most basic of these constraints are given by Shannon's coding theorems.[12] The instrument was described as a transducer, but it also can be considered to be a communication channel between the process and the modeler. The capacity of this channel in bits per measurement is $\dot{I} = \tau^{-1} H(\Pi_\epsilon(\mathrm{D}))$, where $H(\{p_i\}) = -\sum_{\{p_i\}} p_i \log_2 p_i$ is the Shannon entropy of the distribution

$\{p_i = \mu(\pi_i) : \pi_i \in \Pi_\epsilon\}$. If the process is deterministic and has entropy $h_\mu\left(\vec{X}\right) > 0$, a theorem of Kolmogorov's says that this rate is maximized for a given process if the partition $\Pi_\epsilon(\mathrm{D})$ is generating.[13] This property requires infinite sequences of cell indices to be in a finite-to-one correspondence with the process's states. A similar result was shown to hold for the classes of process of interest here: deterministic, but coupled to an extrinsic noise source.[14] Note that the generating partition requirement necessarily determines the number D of probes required by the instrument.

This is, however, the mathematical side of the problem here: verification of a generating partition requires knowledge of the equations of motion. Thus, Kolmogorov's result turns the inference problem on its head. Here interest is focused on the opposite, pragmatic problem of discovering structure from given data. As indicated in [14], the notion of generating partitions gives an operational definition of a good instrument that is useful if the instrument can be altered. If the data is simply given, one can still proceed with modeling. The conclusions cannot be as strong, though, as when the equations of motion are known. Then, again, such knowledge is an idealization.

For an instrument with a generating partition, Shannon's noiseless coding theorem says that the measurement channel must have a capacity higher than process's entropy

$$\dot{I} \geq h_\mu\left(\vec{X}\right) \tag{2}$$

If this is the case then the modeler can use the data stream to reconstruct a model of the process and, for example, estimate its entropy and complexity. These can be obtained to within error levels determined by the process's extrinsic noise level and the number of measurements.

If $\dot{I} < h_\mu\left(\vec{X}\right)$, then Shannon's theorems say that the modeler will not be able to reconstruct a model with an effective noise level less than the equivocation $h_\mu\left(\vec{X}\right) - \dot{I}$ induced by the instrument. That is, there will be an "unreconstructable" portion of the dynamics represented in the signal.

These results assume, as is also done implicitly in Shannon's existence proofs for codes, that the modeler has access to arbitrary inference resources. When these are limited there will be yet another corresponding loss in the quality of the model and an increase in the apparent noise level. It is interesting to note in this context that if one were to adopt Laplace's philosophical stance that all (classical) reality is deterministic and update it with the modern view that it is chaotic, then the inferential limitations discussed here are the general case. Apparent randomness is a consequence of them and not a property of unobserved nature.

2.2 Computation from a Time Series

On what sort of structure in the data stream should the models be based? If the goal includes prediction, as the preceding assumed, then a natural object to reconstruct from the data series is a representation of the process's instantaneous state. Unfortunately, as already

noted, individual measurements are only indirect representations of the state. Indeed, the instrument simply may not supply data of a quality sufficient to discover the true states, independent of the amount of data. So how can the process's "effective" states be accessed?

The answer to this turns on a generalization of the "reconstructed states" introduced, under the assumption that the process is a continuous-state dynamical system, by Packard *et al*.[15] The contention there was that a single time series necessarily contained all of the information about the dynamics of that time series. The notion of reconstructed state was based on Poincaré's view of the intrinsic dimension of an object.[16] This was defined as the largest number of successive cuts through the object resulting in isolated points. A spherical shell in three dimensions by his method is two dimensional since the first cut typically results in a circle and then a second cut, of that circle, isolates two points. One way Packard *et al*. implemented this used probability distributions conditioned on values of the time series' derivatives. That is, the coordinates of the reconstructed state space were taken to be successive time derivatives and the cuts were specified by setting their values. This was, in fact, an implementation of the differential geometric view of the derivatives as locally spanning the graph of the dynamic.

In this reconstruction procedure a state of the underlying process is identified by increasing the number of conditioning variables, i.e. employing successively higher derivatives, until the conditional probability distribution peaks. It was noted shortly thereafter that in the presence of extrinsic noise a number of conditions is reached beyond which the conditional distribution is no longer sharpened.[14] And, as a result the process's state cannot be further identified. The width of the resulting distribution then gives an estimate of the effective extrinsic noise level and so also an estimate of the maximum amount of information contained in observable states. The minimum number of conditions first leading to this situation is an estimate of the effective dimension.

The method of time derivative reconstruction gives the key to discovering states in discrete times series. For discrete time series a state is defined to be the set of subsequences that render the future conditionally independent of the past.[17]* Thus, the observer identifies a state at different times in the data stream as its being in identical conditions of ignorance about the future. (See Fig. 2.)

Let's introduce some notation here. Consider two parts of the data stream $\mathrm{s} = \ldots s_{-2}s_{-1}s_0s_1s_2\ldots$. The one-sided forward sequence $\mathrm{s}_t^{\rightarrow} = s_t s_{t+1} s_{t+2} s_{t+3}\ldots$ and one-sided reverse sequence $\mathrm{s}_t^{\leftarrow} = \ldots s_{t-3}s_{t-2}s_{t-1}s_t$ are obtained from s by splitting it at time t into the forward- and reverse-time semi-infinite subsequences. Consider the joint distribution of possible forward sequences $\{\mathrm{s}^{\rightarrow}\}$ and reverse sequences $\{\mathrm{s}^{\leftarrow}\}$ over all times t

$$Pr(\mathrm{s}^{\rightarrow}, \mathrm{s}^{\leftarrow}) = Pr(\mathrm{s}^{\rightarrow}|\mathrm{s}^{\leftarrow})Pr(\mathrm{s}^{\leftarrow}) \tag{3}$$

* This notion of state is widespread; appearing in various guises in early symbolic dynamics and ergodic and automata theories. It is close to the basic notion of state in Markov chain theory. Interestingly, the notion of conditional independence is playing an increasingly central role in the design of expert systems.[18]

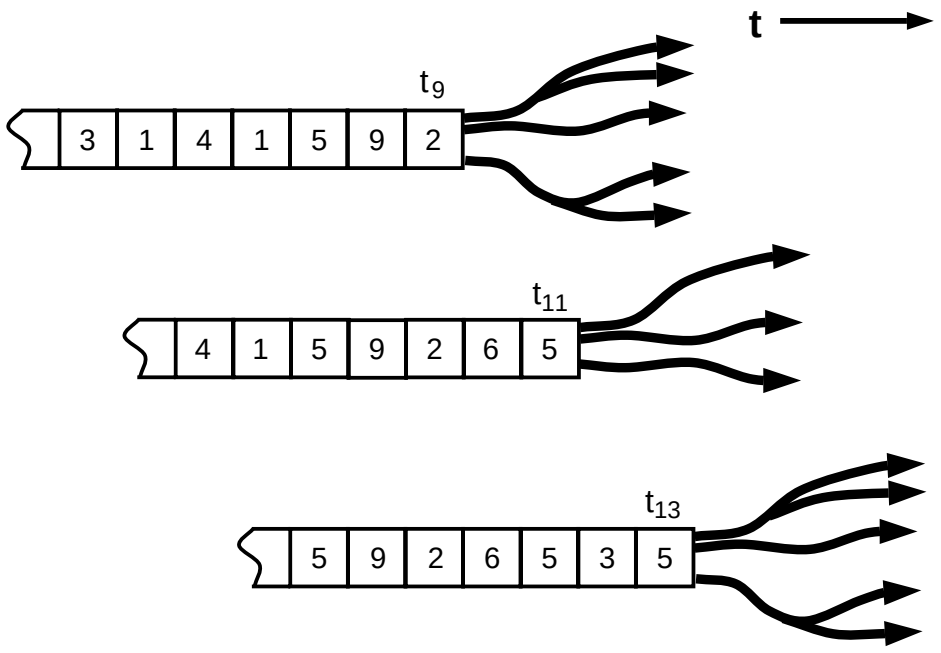

Fig. 2 Morph-equivalence induces conditionally independent states. When the template of future possibilities, i.e. allowed future subsequences and their past-conditioned probabilities, has the same structure then the process is in the same generalized state. At t_9 and at t_{13}, the process is in the same state; at t_{11} it is in another different state.

The conditional distribution $Pr(\mathrm{s}^{\rightarrow}|\omega)$ is to be understood as a function over all possible forward sequences $\{\mathrm{s}^{\rightarrow}\}$ that can follow the particular sequence ω where ever it occurs in s.

Then the same state $S \in \mathbf{S}$ is associated with all those times $t, t' \in \{t_{i_1}, t_{i_2}, t_{i_3} \ldots : i_k \in \mathbf{Z}\}$ such that past-conditioned future distributions are the same. That is,

$$t \sim t' \text{ if and only if } Pr(\mathrm{s}^{\rightarrow}|\mathrm{s}^{\leftarrow}_t) = Pr(\mathrm{s}^{\rightarrow}|\mathrm{s}^{\leftarrow}_{t'}) \tag{4}$$

If the source generating the data stream is ergodic, then there are several comments that serve to clarify how this relation defines states. First, the sequences $\mathrm{s}^{\leftarrow}_t$ and $\mathrm{s}^{\leftarrow}_{t'}$ are typically distinct. If $t \sim t'$, Eq. (4) means that upon having seen different histories one can be, nonetheless, in the same state of anticipation or ignorance about what will happen in the future. Second, $\mathrm{s}^{\leftarrow}_t$ and $\mathrm{s}^{\leftarrow}_{t'}$, when considered as particular symbol sequences, will each occur in s many times other than t and t', respectively. Finally, the conditional distributions $Pr(\mathrm{s}^{\rightarrow}|\mathrm{s}^{\leftarrow}_t)$ and $Pr(\mathrm{s}^{\rightarrow}|\mathrm{s}^{\leftarrow}_{t'})$ are functions over a nontrivial range of "follower" sequences $\mathrm{s}^{\rightarrow}$.

This gives a formal definition to the set $\mathbf{S}$ of states as equivalence classes of future predictability: $\sim$ is the underlying equivalence relation that partitions temporal shifts of the data stream into equivalence classes. The states are simply labels for those classes. For a given state S the set of future sequences $\left\{\mathrm{s}^{\rightarrow}_S : S \in \mathbf{S}\right\}$ that can be observed from it is called its future morph. The set of sequences that lead to S is called its past morph. Anticipating a later section somewhat, note that the state and its morphs are the contexts in which an individual measurement takes on semantic content. Each measurement is anticipated or "understood" by the observer *vis á vis* its model and in particular the structure of the states.

Once these states are found, the temporal evolution of the process, its (symbolic) dynamic, is given by a mapping from states to states $T : \mathbf{S} \to \mathbf{S}$; that is, $S_{t+1} = TS_t$.

The available reconstruction algorithms infer the states $\mathbf{S}$ via various approximations of the equivalence class conditions specified in (4).[11,17,19] I refer to these procedures generically as "machine reconstruction". The result of machine reconstruction, then, is the discovery of the underlying process's "hidden" states. This should be contrasted with the *ad hoc* methods employed in hidden Markov modeling in which a set of states and a transition structure are imposed by the modeler at the outset.[20,21]

Thus, the overall procedure has two steps. This first is to identify the states and the second is to infer the transformation T. In the following I review the simplest implementation since this affords the most direct means of commenting on certain properties of the resulting representations.

Initially, a parse tree is built from the data stream $\mathbf{s}$. A window of width D is advanced through $\mathbf{s}$ one symbol at a time. If $\mathbf{s} = \ldots 1010101110101 \ldots$ then at some time t the $D = 5$ subsequence $s_t^5 = 10101$ is seen, followed by $s_{t+1}^5 = 01010$. Each such subsequence is represented in the parse tree as a path. The tree has depth $D = 5$. (See Fig. 3).

Counts are accumulated at each tree node as each subsequence is put into to the parse tree. If the associated path is not present in the tree, it is added; new nodes each begin with a count of 1 and counts in existing nodes are incremented. If $\mathbf{s}$ has length N, then a node probability, which is also the probability of the sequence leading to it, is estimated by its relative frequency

$$Pr(n) = \frac{c_n}{N - D} \tag{5}$$

where c_n is the count accumulated at node n. The node-to-node transition probabilities $Pr(n \to n')$ are estimated by

$$Pr\big(n \to n'\big) = Pr\Big(s|s^k\Big) = \begin{cases} \frac{Pr\left(ss^k\right)}{Pr(s^k)} \approx \frac{c_{n'}}{c_n} & c_n > 0 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where the length k sequence s^k leads to node n and the length $k+1$ sequence $s^{k+1} = ss^k$ leads to node n' on symbol $s \in \mathbf{A}$.

The window length D is an approximation parameter. The longer it is, the more correlation, structure, and so on, is captured by the tree. Since $\mathbf{s}$ is of finite length the tree depth cannot be indefinitely extended. Thus, to obtain good estimates of the subsequences generated by the process and of the tree structure, the tree depth must be set at some optimal value. That value in turn depends on the amount of available data. The procedure for finding the optimal depth D is discussed elsewhere.

To infer the set of states we look for distinct subtrees of a given depth L. This is the length of future subsequences over which the morphs must agree. This step introduces a second approximation parameter, the morph depth L. Naturally, $L \leq D$ and typically one

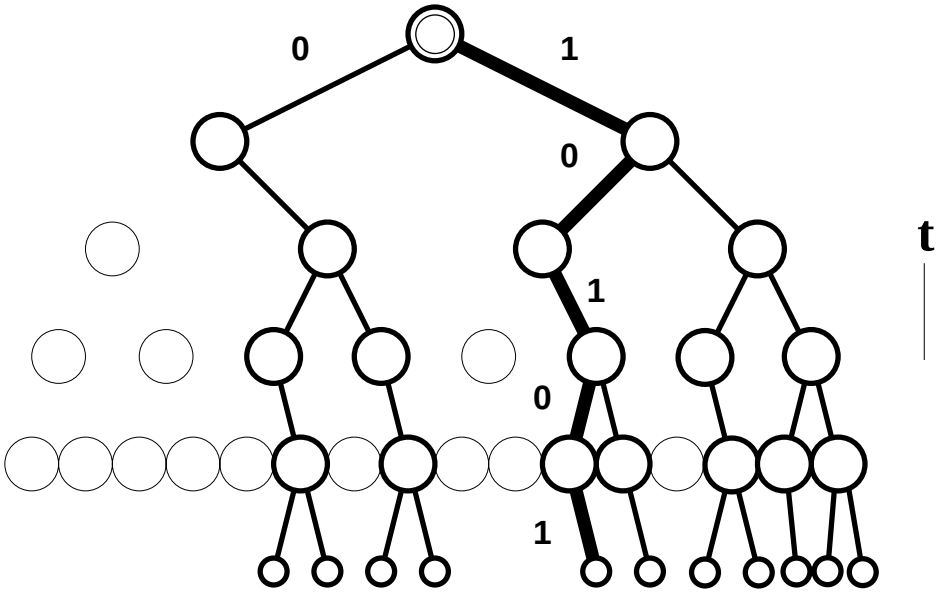

Fig. 3 The parse tree for the process "every other symbol is a 1". The tree is a hierarchical representation of the subsequences produced by the process. Here the tree depth is $D = 5$. If $\mathbf{s} = \ldots 1010101110101\ldots$, then the sequence $s_t^5 = 10101$ will be seen. This is put into the parse tree, as shown by the bold line. Note that time goes down the tree. When counts of the subsequence that lead to a given tree node are accumulated, the tree gives a hierarchical representation of the infinite sequence probability distribution.

takes $2L = D$. Over a range of sources and for a given tree depth, that choice balances two requirements for the morph's optimal statistical estimation. The first is the need for a large number of examples of any given morph. This suggests taking $L \ll D$ since the upper bound on the number of morph instances is $2^{D-L+1} - 1$. The second requirement is to allow a sufficient diversity of morphs; and so of states. This inclines one toward taking $L \approx D$, since the number of distinct depth L subtrees is $2^{2^L} - 1$. The analysis of this optimization is presented elsewhere. Investigating the parse tree of Fig. 3 using depth $L = 2$ subtrees one finds the three morphs shown in Fig. 4. In this way, three "machine" states $\mathbf{S} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$ have been discovered for the "every other symbol is a 1" process of Fig. 3 up to the approximation implied by setting $D = 5$ and $L = 2$. It can be shown that these are sufficient to infer an exact model of the process.

The state-to-state transition structure is obtained by looking at how the morphs change into one another upon moving down the parse tree, i.e. upon reading $s = 0$ or $s = 1$. The resulting transformation T is represented graphically by the machine shown in Fig. 5. This should be compared to the parse tree (Fig. 3). There the morph below the top node, which is associated with machine state A, makes a transition on a 1 to a tree node with a morph below it in the same equivalence class (machine state A). On a 0, though, the top tree node makes a transition to a tree node with a morph associated with machine state B. In just this way the machine of Fig. 5 summarizes the morph to morph transition structure T on the parse tree.

This exposition covered only topological reconstruction: subtrees were compared only up to the subsequences $\{\mathbf{s}_t^{\rightarrow}\}$ which were observed. Of course, what is needed are states that are not only topologically distinct, but also distinct in probability as indicated by (4).

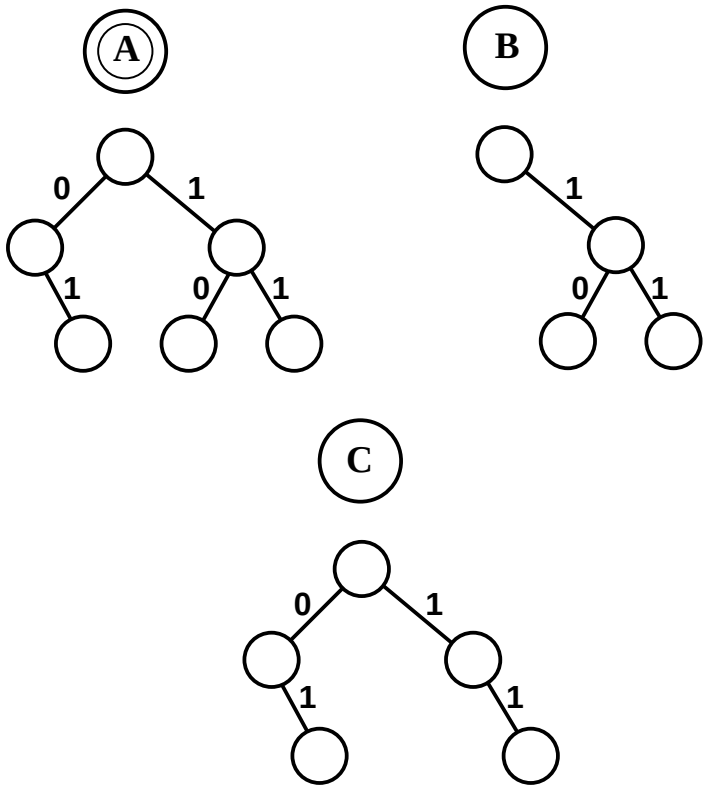

Fig. 4 The morph depth $L = 2$ subtrees found below tree nodes down to depth 3 in Fig. 3's parse tree. Each morph, or subtree, has been labeled by its associated machine state (cf. Fig. 5).

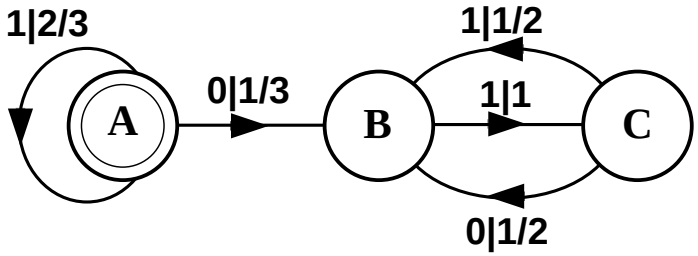

Fig. 5 The (topologically) reconstructed machine for the process "every other symbol is a 1".

For this we must choose some metric to compare the conditional histograms associated with the morphs. I will briefly sketch one way to do this.

The goal is to develop an implementable approximation to (4) which defines states in terms of tree node conditional probability equivalence classes. Those equivalence classes require a subtree similarity relation between two arbitrary tree nodes; denote these n and n'. First assume that these nodes are topologically similar; that is, their morphs m_n and $\mathrm{m}_{n'}$ are the same, up to some morph depth L,

$$\begin{aligned} n \underset{L}{\sim} n' \text{ if and only if } \mathrm{m}_n = \mathrm{m}_{n'} \\ \text{where } \mathrm{m}_n = \left\{ s^L | n \right\} \text{ and } m_{n'} = \left\{ s^L | n' \right\} \end{aligned} \tag{7}$$

where s^L is a length L sequence. If $\mathrm{m}_n \neq \mathrm{m}_{n'}$, then certainly the nodes cannot be similar in probability. Probabilistic similarity is then an additional constraint and defined by

$$n \underset{\delta}{\sim} n' \text{ if and only if } \left| Pr(\omega|n) - Pr\left(\omega|n'\right) \right| \leq \delta, \ \forall \omega \in \mathrm{m}_n \tag{8}$$

δ is yet another approximation parameter. Without going into details, it is important to note that the implied comparisons for subtree similarity can be done in a relatively efficient recursive algorithm that stops as soon as a difference in the morph structure or node transition probabilities is found at some level in the two subtrees under comparison. Finally, the state to state transition probabilities are then taken from the estimated node-to-node transition probabilities. Detailed statistical analysis of the overall procedure and the optimal selection of δ is given elsewhere.

2.3 Statistical Mechanics

Machine reconstruction gives then a set of states, that will be associated with a set $\mathbf{V} = \{v\}$ of vertices, and a set of transitions, that will be associated with a set

$$\mathbf{E} = \left\{ e : e \sim v \underset{s}{\rightarrow} v', \quad v, v' \in \mathbf{V}, s \in \mathbf{A} \right\} \tag{9}$$

of labeled edges. The graphical depiction of $M = \{\mathbf{V}, \mathbf{E}, \mathbf{A}\}$ is a labeled directed graph as seen above in Fig. 5. The full probabilistic structure is described by a set of input-alphabet labeled transition matrices

$$\left\{ T^{(s)} : \left(T^{(s)}\right)_{vv'} = p_{v \underset{s}{\rightarrow} v'}, \quad v, v' \in \mathbf{V}, s \in \mathbf{A} \right\} \tag{10}$$

where $p_{v \underset{s}{\rightarrow} v'}$ denotes the conditional probability to make a transition to state v' from state v on observing symbol s.

Given the current state v_t at time t the future is conditionally independent of the past

$$\begin{aligned} Pr(\mathbf{s}) &= Pr\left(\mathbf{s}_t^{\leftarrow} \mathbf{s}_{t+1}^{\rightarrow}\right) \\ &= Pr\left(\mathbf{s}_{t+1}^{\rightarrow} | \mathbf{s}_t^{\leftarrow}\right) Pr(\mathbf{s}_t^{\leftarrow}) \\ &= Pr\left(\mathbf{s}_{t+1}^{\rightarrow} | v_t\right) Pr(v_t) \end{aligned} \tag{11}$$

By factoring the joint distribution over the observed subsequences, the discovered states vastly reduce the dimensionality of the process's representation. In the case of strings of length N, the representation changes from requiring an N-dimensional probability vector for the joint distribution to a set of $\|\mathbf{E}\|$ transition probabilities.

Rather than simply looking up the probability of a given sequence $\mathbf{s}^N = s_0 s_1 s_2 \ldots s_{N-1}, \quad s_i \in \mathbf{A}$, in the table of joint probabilities, its probability is recovered from the machine by computing the telescoping product of conditional transition probabilities

$$Pr\left(\mathbf{s}^N\right) = p_{v_0} p_{v_0 \underset{s_0}{\rightarrow} v_1} p_{v_1 \underset{s_1}{\rightarrow} v_2} \cdots p_{v_{N-1} \underset{s_{N-1}}{\rightarrow} v_N} \tag{12}$$

Here v_0 is the unique start state. By the nature of machine reconstruction it is the state of total ignorance: before the first measurement is made $p_{v_0} = 1$. The sequence $v_0, v_1, v_2, \ldots, v_{N-1}, v_N$ are those states through which the sequence s^N drives the machine.

Recall Chomsky's criterion for a scientific theory quoted at the beginning. A reconstructed machine is, in his sense, a theory of the source of the data from which it was inferred. In addition, a machine presumes to ascribe structure to unobserved sequences, such as whether or not they occur and their probabilities, by the above telescoping product (12). In fact, a machine typically does so for an indefinite number of unseen measurements. And this too is in accord with Chomsky's remark that a theory "predict an indefinite number of new phenomena".

A machine is a compact and very informative representation of the underlying source. In order to appreciate the properties that it captures, there are several statistics that can be computed from a given machine. Each of these is, naturally, a simplification in one way or another of the structure captured by the machine. Up to this point there has been no restriction on the number $\|\mathbf{V}\|$ of states. For the sake of simplicity in the following the number of states will be assumed to be finite. This restricts the general model class to that of stochastic finite automata.

One useful reduction of a machine M is to ask for its equivalent Markov process. This is described by the stochastic connection matrix

$$T = \sum_{s \in \mathbf{A}} T^{(s)} \tag{13}$$

where $(T)_{vv'} = p_{v \to v'}$ is the state to state transition probability, unconditioned by the measurement symbols. By construction every state has an outgoing transition. This is reflected in the fact that T is a stochastic matrix: $\sum_{v' \in \mathbf{V}} p_{vv'} = 1$. It should be clear from dropping the input-alphabet transition labels from the machine that the detailed, I call it "computational", structure of the input data stream has been lost. All that is retained in T is the state transition structure and this is a Markov chain. The interesting fact is that Markov chains are a proper subset of stochastic finite machines. Examples latter on will support this contention. But it is exactly at this step of unlabeling the machine that the "properness" relation between these two model classes appears.

The stationary state probabilities $\vec{p}_{\mathbf{V}} = \left\{ p_v : \sum_{v \in \mathbf{V}} p_v = 1, v \in \mathbf{V} \right\}$ are given by the left eigenvector of T

$$\vec{p}_{\mathbf{V}} T = \vec{p}_{\mathbf{V}} \tag{14}$$

The entropy rate of the Markov chain is then[12]

$$h_\mu(T) = -\sum_{v \in \mathbf{V}} p_v \sum_{v' \in \mathbf{V}} p_{v \to v'} \log_2 p_{v \to v'} \tag{15}$$

As it is an average of transition uncertainty over all the states, it measures the information production rate in bits per time step. It is also the growth rate of the Shannon information in subsequences

$$h_\mu = \lim_{L \to \infty} \frac{H(L)}{L} \tag{16}$$

where $H(L) = -\sum_{\omega \in \{s^L\}} Pr(\omega) \log_2 Pr(\omega)$. That is, $H(L) \underset{L \to \infty}{\propto} h_\mu L$. In general, subsequences are not in a one-to-one correspondence with the Markov chain's state-to-state transition sequences. Nonetheless, it is a finite-to-one relationship. And so, the Markov entropy rate is also the entropy rate of the original data source: $h_\mu(M) = h_\mu(T)$. More directly, this is given by

$$h_\mu(M) = -\sum_{v \in \mathbf{V}} p_v \sum_{v' \in \mathbf{V}} \sum_{s \in \mathbf{A}} p_{v \underset{s}{\to} v'} \log_2 p_{v \underset{s}{\to} v'} \tag{17}$$

Thus, once a machine is reconstructed from a data stream, its entropy is an estimate of underlying process's entropy rate.

The complexity* quantifies the information in the state-alphabet sequences

$$C_\mu(M) = H(\vec{p}_{\mathbf{V}}) = -\sum_{v \in \mathbf{V}} p_v \log_2 p_v \tag{18}$$

It measures the amount of memory in the source. For completeness, note that there is an edge-complexity that is the information contained in the asymptotic edge distribution $\vec{p}_{\mathbf{E}} = \left(p_e = p_v p_{v \underset{s}{\to} v'} : e \in \mathbf{E} \right)$

$$C_\mu^e(M) = -\sum_{e \in \mathbf{E}} p_e \log_2 p_e \tag{19}$$

These quantities are not independent. Conservation of information leads to the relation

$$C_\mu^e = C_\mu + h_\mu \tag{20}$$

Thus, there are only two independent quantities when modeling a source as a stochastic finite automaton. The entropy h_μ, as a measure of the diversity of patterns, and the complexity C_μ, as a measure of memory, have been taken as the two elementary "information processing" coordinates with which to analyze a range of sources.[19]

There is another set of quantities that derive from the skeletal structure of the machine. If we drop all probabilistic structure on the machine, the growth rate of the raw number of sequences it produces is the topological entropy

$$h = \log_2 \lambda(T_0) \tag{21}$$

where $T_0 = \sum_{s \in \mathbf{A}} T_0^{(s)}$ is called the connection matrix and $\lambda(T_0)$ is its principal eigenvalue. It is formed from the symbol matrices

$$\left\{ T_0^{(s)} : \left(T_0^{(s)} \right)_{vv'} = \left\{ \begin{array}{ll} 1 & p_{v \underset{s}{\to} v'} > 0 \\ 0 & \text{otherwise} \end{array} s \in \mathbf{A} \right\} \right. \tag{22}$$

* Within the reconstruction hierarchy this is actually the finitary complexity, since discussion is restricted to processes with a finite number of states. Although, I have not introduced this restriction in unnecessary places. Related forms of the finitary complexity have been considered before, outside of the context of reconstruction, assuming generating partitions and known equations of motion.[22–25]

The state and transition topological complexities are

$$\begin{aligned} C &= \log_2 \|\mathbf{V}\| \\ C^e &= \log_2 \|\mathbf{E}\| \end{aligned} \tag{23}$$

Although it falls somewhat outside of the present discussion, it is worthwhile noting that these entropies and complexities can be integrated into a single parametrized framework. The resulting formulation gives a thermodynamics of ϵ-machines.[26]

2.4 Complexity

It is useful at this stage to stop and reflect on some properties of the models whose reconstruction and structure have just been described. Consider two extreme data sources. The first, highly predictable, produces a stream of 1s; the second, highly unpredictable, is an ideal random source of binary symbols. The parse tree of the predictable source is a single path of 1s. And there is a single distinct subtree, at any depth. As a result the machine has a single state and a single transition on $s = 1$: a simple model of a simple source. For the ideal random source the parse tree, again to any depth, is the full binary tree. All paths appear in the parse tree since all binary subsequences are produced by the source. There is a single subtree, of any morph depth, at all parse tree depths: the full binary subtree. And the machine has a single state with two transitions: one on $s = 1$ and one on $s = 0$. The result is a simple machine, even though the source produces the widest diversity of binary sequences. Thus, these two zero entropy and maximal entropy sources have zero complexity.

A simple gedanken experiment serves to illustrate how (finitary) complexity is a measure of a machine's memory capacity. Consider two observers $\mathbf{A}$ and $\mathbf{B}$, each with the same model M of some process. $\mathbf{A}$ is allowed to start machine M in any state and uses it to generate binary strings that are determined by the edge labels of the transitions taken. These strings are passed to observer $\mathbf{B}$ which traces their effect through its own copy of M. On average how much information about M's state can $\mathbf{A}$ communicate to $\mathbf{B}$ via the binary strings? If the machine describes (say) a period three process, e.g. it outputs strings like $101101101\ldots$, $011011011\ldots$, and $110110110\ldots$; it has $\|\mathbf{V}\| = 3$ states. Since $\mathbf{A}$ starts M in any state, $\mathbf{B}$ can learn only the information of the process's phase in the period 3 cycle. This is $\log_2 \|\mathbf{V}\| = 1.584\ldots$ bits of information on average about the process's state, if $\mathbf{A}$ chooses the initial states with equal probability. However, if the machine describes an ideal random binary process, by definition $\mathbf{A}$ can communicate no information to $\mathbf{B}$, since there is no structure in the sequences to use for this purpose. This is reflected in the fact, as already noted above, that the corresponding machine has a single state and its complexity is $C_\mu(M) = \log_2 1 = 0$. In this way, a process's complexity is the amount of information that someone controlling its start state can communicate to another.

These examples serve to highlight one of the most basic properties of complexity, as I use the term.* Both predictable and random sources are simple in the sense that their models are small. Complex processes in this view have large models. In computational terms, complex processes have, as a minimum requirement, a large amount of memory as revealed by many internal states in the reconstructed machine. Most importantly, this memory is structured in particular ways that support different types of computation. The sections below on knowledge and meaning show several consequences of computational structure.

In the most general setting, I use the word "complexity" to refer to the amount of information contained in observer-resolvable equivalence classes.[17] This approach puts the burden directly on any complexity definition to explicitly state the representation employed by the observer. For processes with finite memory, the complexity is measured by the quantities labeled above by C. The general notion, i.e. without the finiteness restruction, has been referred to as the "statistical complexity" in order to distinguish it from the Chaitin-Kolmogorov complexity,[33,28] the Lempel-Ziv complexity,[34] Rissanen's stochastic complexity,[35] and others[36,37] which are all equivalent in the limit of long data streams to the process's Kolmogorov-Sinai entropy $h_\mu\left(\vec{X}\right)$. If the instrument Π_ϵ is generating and $\mu\left(\vec{X}\right)$ is absolutely continuous, these quantities are given by the entropy rate of the reconstructed ϵ-machine, i.e. (17).[38] Accordingly, I use the phrases "entropy" and "entropy rate" to refer to such quantities. They measure the diversity of sequences that a process produces. Implicit in their definitions is the restriction that the modeler must pay computationally for each random bit. Simply stated, the overarching goal is exact description of the data stream. In the modeling approach advocated here the modeler is allowed to flip a coin or to sample the heat bath to which it may be coupled. "Complexity" is reserved in my vocabulary to refer to a process's structural properties, such as amount of memory, syntactic and semantic properties, and other types of computational capacity.

---

* A number of authors have considered measures of complexity that have heuristically similar properties. One of the first computation theoretic notions along these lines was "logical depth".[27] It relies on obtaining a minimal universal Turing machine program. And so, as a consequence of Kolmogorov's theorem, it is uncomputable.[28] Apparently, the first constructive measure proposed, i.e. one that could be estimated from a data stream, was the excess entropy convergence rate.[29] The closely-related total excess entropy[14,30] was also suggested as a measure of information contained in a process. This was later recoined the "stored information"[31] and also the "effective measure complexity".[24] Using the kneading calculus for one-dimensional maps, algorithms were given to estimate the number of equivalent Markov states.[22] The size of deterministic finite automata describing patterns generated by cellular automata was used to measure the development of spatial complexity.[23] The present author proposed a complexity measure based on the size of the group of symmetries in an object; Los Alamos Workshop on Dimension and Entropy (1985). Finally, the diffusion rate on a hierarchical potential was shown to exhibit similar "complexity" properties.[32] There is clearly no lack of complexity measures. During the last five years yet more have been proposed, including the finitary and more general complexities considered here. All of these are seen to be similar or different according to (i) the computational model class selected, often implicitly, and (ii) whether or not the equations of motion are assumed known.

The present review is not the place to comment on the wide range of alternative notions of "complexity" that have been proposed recently for nonlinear physics. The reader is referred to the comments and especially the citations in [17,19,39]. It is important to point out, however, that the notion defined here does not require (i) knowledge of the governing equations of motion nor of the process's embedding dimension, (ii) the prior existence of exact conditional probabilities, (iii) Markov or even generating partitions of the state space, (iv) continuity and differentiability of the state variables, nor (v) the existence of periodic orbits. Within the framework of machine reconstruction such restrictions are to be viewed as verifiable assumptions. They can be given explicit form in terms of ϵ-machine properties. This is an essential criterion for any modeling framework that seeks to roll back the frontier of subjectivity to expose underlying mechanisms. Without it, there is little basis for appreciating what aspects of experimental reality are masked by a given set of modeling assumptions and, more importantly, for a study of the dynamics of modeling itself.

2.5 Compressing and Decompressing Chaos

A dynamical system, or any physical process for that matter, is a communication channel.[22,40] In the case of a chaotic physical process, the associated mechanism is viewed as being driven by a heat bath or a randomness source. The dynamics of the effective channel then connects the heat bath to the process's macroscopic, observable states.

This picture is particularly explicit in the case of stochastic automata. The machine model of the source can be reinterpreted as a transducer. A transducer is a machine that not only reads symbols from some input, taking the appropriate transitions, but also outputs a symbol determined by each transition taken. In the graphical representation the edges of a transducer are labeled with both input and output symbols. The latter may come from different alphabets. For example, the input symbols could be the binary strings considered up to this point, and the output symbols could be labels for each of the states reached after making a transition.

For now, consider both alphabets to be binary. Then the strings produced by a source are generated by the following communication channel that is built from the source's reconstructed machine. Starting from the start state, any deterministic transition, i.e. a state with only a single outgoing edge, is taken and that edge's output symbol is emitted. At states with more than one outgoing edge, a symbol is read from the heat bath and that edge is taken whose input symbol matches the symbol read. On taking the transition, the edge's output symbol is emitted. This defines a transformation from (random) strings to strings containing the constraints and structure that define the original machine.

The reverse procedure can be implemented to give a compression of a string. In this situation, as the string is read, no symbol is output when the transition is deterministic. When there is a branching, the corresponding string symbol is copied to the output. This transducer thereby removes all of the (topological) regularity of the string. The result is

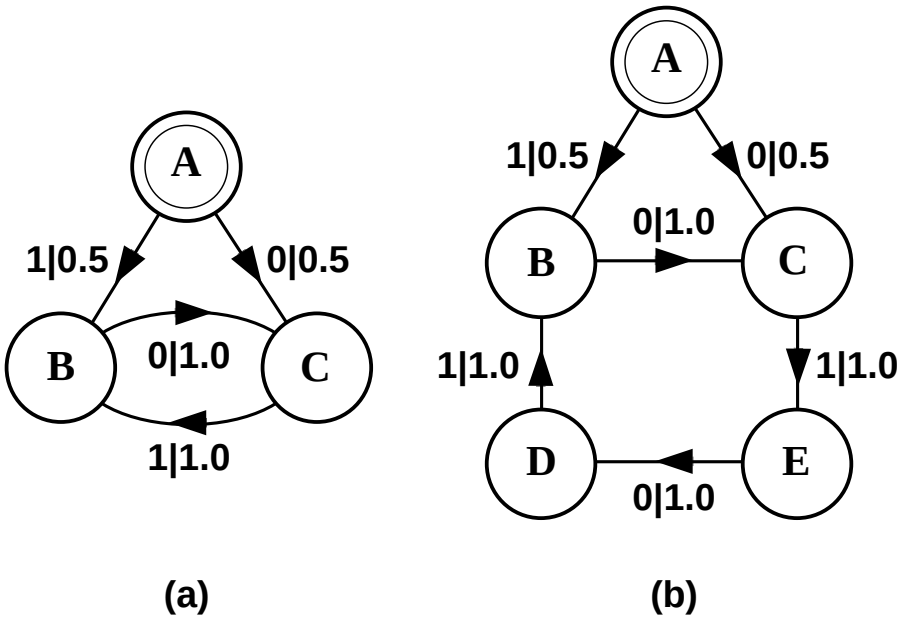

Fig. 6 Two consistent models of the period two process producing $\mathbf{s} = \ldots 10101010101010101\ldots$.
(a) The minimal machine. (b) A nonminimal machine.

a (partially) random string. Coupling this compression transducer with the preceding heat bath transducer gives a data transmission scheme for a string, if both transducers are built from a model of the string.

2.6 On the Importance of Being Minimal

Given a machine model of a process, the machine's complexity, or any other statistic for that matter, need not be that of the process. The inference that some model property also holds for the process generally depends on details of the reconstruction algorithm used and the attendant assumptions and restrictions it imposes. Nonetheless, something about the underlying process has been estimated by reconstructing the machine by the above method. In order to see just what this is, this section focuses on an important property of the reconstruction method and the notion of state on which it is based. The property is that machine reconstruction produces the unique machine with the smallest number of states. As the following will argue, due to uniqueness and minimality there are some properties of the underlying process that can be inferred from and are well-estimated by the machine.

Figure 6 shows two machines that are (statistically) consistent models of the period two process that produces $\mathbf{s} = \ldots 10101010101010101\ldots$. In fact, both machines are exact, since they describe the structure of the data stream without error.

Figure 7 shows two machines, from the other end of the entropy spectrum, that are (statistically) consistent models of the ideal random process $(1+0)^*$, where $+$ means to choose $s = 0$ or $s = 1$ with equal probability. Again, both machines are exact, describing as they do the source's structure without approximation.

Minimality of the reconstructed machine ensures that its complexity is a measure, actually a lower-bound estimate, of the process's memory capacity. As just noted, the latter is the

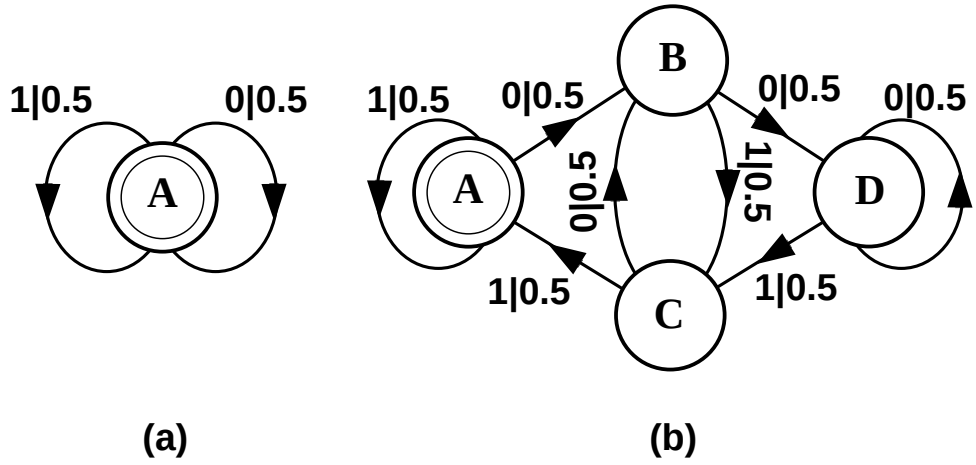

Fig. 7 Two consistent models of the ideal random process $(1+0)^*$. (a) The minimal machine. (b) A nonminimal machine.

maximal amount of state-information that can be transmitted using the process as a source and allowing one to select the source's initial state. The state information that can be conveyed for a periodic process is the phase of the periodic cycle. In the present case (Fig. 6), the period was two and so one bit of phase information can be communicated. The observer receives that bit at the moment it synchronizes to the data stream.* The period two machine with five states, Fig. 6(b), has a complexity of $\log_2 5 \approx 2.321$ bits. But this amount of information clearly cannot be transmitted with the period two source. The nonminimal machine has a complexity that is too large. The case of the ideal random process is even more extreme. The nonminimal machine has a complexity of $\log_2 4 = 2$ bits, but by definition the data stream and source have no structure that can be used for communication. The minimal machine, Fig. 7(a), has zero complexity, in accord with intuition.

In summary, then, even though there is a very large number of machines consistent with a given data stream, the minimal one is singled out not only for its uniqueness and compactness, but also in order to estimate and understand the source's properties. Minimality keeps the modeler from inferring structure in the source which is not justified by the given data.

3. KNOWLEDGE RELAXATION†

The next two sections investigate how models can be used by an observer. An observer's knowledge $\mathcal{K}_P$ of a process P consists of the data stream, its current model, and how the information used to build the model was obtained.‡ Here the latter is given by the measuring instrument $\mathcal{I} = \{\Pi_\epsilon(\mathrm{D}), \tau\}$. To facilitate interpretation and calculations, the following will assume a simple data acquisition discipline with uniform sampling interval τ and a time-independent measurement partition Π_ϵ. Further simplification comes from ignoring external

* Synchronization is the subject of a later section.

† This and the following section also appear elsewhere.[41]

‡ In principle, the observer's knowledge also consists of the reconstruction method and its various assumptions. But it is best to not elaborate this here. These and other unmentioned variables are assumed to be fixed.

factors, such as what the observer intends or needs to do with the model, by assuming that the observer's goal is optimal prediction with respect to the model class of finitary machines.

The totality of knowledge available to an observer is given by the development of its $\mathcal{K}_P$ at each moment during its history. If we make the further assumption that by some agency the observer has at each moment in its history optimally encoded the available current and past measurements into its model, then the totality of knowledge consists of four parts: the time series of measurements, the instrument by which they were obtained, and the current model and its current state. Stating these points so explicitly helps to make clear the upper bound on what the observer can know about its environment. Even if the observer is allowed arbitrary computational resources, given either finite information from a process or finite time, only a finite amount of structure can be inferred.

For the following assume that an observer's model of a process is an ϵ-machine. To see its role in the change in $\mathcal{K}_P$ consider the situation in which the model structure is kept fixed. Starting from the state v_0 of total ignorance about the process's state, successive steps through the machine lead to a refinement of the observer's knowledge as determined by a sequence of measurements. The average increase in $\mathcal{K}_P$ is given by a diffusion of information throughout the model. The machine transition probabilities, especially those connected with transient states, govern how the observer gains more information about the process with longer measurement sequences.

The average increase is governed by a diffusion of information throughout the given model. Consider for the moment the parse tree as a model. There is a flow of probability downwards, in increasing time, toward the leaves. This is a unidirectional diffusion of information on an ultrametric structure.[42]* The ultrametric distance on the tree is sequence length or, more simply, time itself. Taking the machine as the model, the distance between two state events, (say) $\mathrm{A} \in \mathbf{V}$ and $\mathrm{B} \in \mathbf{V}$, is the length of the shortest directed machine path between them. The direction is determined by the order in which A and B are observed to occur. This, in turn, translates back into the difference in parse tree levels at which the associated morphs are found. When viewed in terms of either the parse tree or the machine, the data stream and its joint distribution exhibit an ultrametric structure. That structure, in turn, determines those properties of time that can be inferred from the data stream.

In a more quantitative vein, a measure of knowledge relaxation on finitary machines is given by the time-dependent finitary complexity

$$C_\mu(t) = H(\vec{p}_{\mathbf{V}}(t)) \tag{24}$$

where $H(\{p_i\}) = \sum_{p_i \in P} p_i \log_2 p_i$ is the Shannon entropy of the distribution $\{p_i\}$ and

$$\vec{p}_{\mathbf{V}}(t+1) = \vec{p}_{\mathbf{V}}(t)T \tag{25}$$

* Unidirectional diffusion on a hierarchical structure is described by the theory of branching processes, and does not necessarily call into play the phenomena associated with ultrametric diffusion. If the direction of time is unknown, however, then the diffusion of the observer's information is bidirectional on the parse tree and so a diffusion on an ultrametric space.

is the probability distribution at time t beginning with the initial distribution $p_{\mathbf{V}}(0) = (1, 0, 0, \ldots)$ concentrated on the start state. This distribution represents the observer's condition of total ignorance of the process's state, i.e. before any measurements have been made, and correspondingly $C_\mu(0) = 0$. $C_\mu(t)$ is simply (the negative of) the Boltzmann H-function in the present setting. There is an analogous H-theorem for stochastic ϵ-machines: $C_\mu(t)$ converges monotonically when $\vec{p}_{\mathbf{V}}(t)$ is sufficiently close to $\vec{p}_{\mathbf{V}} = \vec{p}_{\mathbf{V}}(\infty)$: $C_\mu(t) \underset{t\to\infty}{\to} C_\mu$. That is, the time-dependent complexity limits on the finitary complexity. Furthermore, the observer has the maximal amount of information about the process, i.e. the observer's knowledge is in equilibrium with the process, when $C_\mu(t+1) - C_\mu(t)$ vanishes for all $t > t_{\text{lock}}$, where t_{lock} is some fixed time characteristic of the process.

For finitary machines there are two convergence behaviors for $C_\mu(t)$. These are illustrated in Fig. 8 for three processes: one P_3 which is period 3 and generates $(101)^*$, one P_2 in which only isolated zeros are allowed, and one P_1 that generates 1s in blocks of even length bounded by 0s. The first behavior type, illustrated by P_3 and P_2, is monotonic convergence from below. In fact, the asymptotic approach occurs in finite time. This is the case for periodic and recurrent Markov chains, where the latter refers to finite state stochastic processes whose support is a subshift of finite type (SSFT).[43] The convergence here is over-damped.

The second convergence type, illustrated by P_1, is only asymptotic; convergence to the asymptotic state distribution is only at infinite time. There are two subcases. The first is monotonic increasing convergence; the conventional picture of stochastic process convergence. The second subcase (P_1) is nonmonotonic convergence. Starting in the condition of total ignorance leads to a critically-damped convergence with a single overshoot of the finitary complexity. With other initial distributions oscillations, i.e. underdamped convergence, can be seen. Exact convergence is only at infinite time. This convergence type is associated with machines having cycles in the transient states or, in the classification of symbolic dynamics, with machines whose support is a strictly Sofic system (SSS).[43]* For these, at some point in time the initial distribution spreads out over more than just the recurrent states. $C_\mu(t)$ can then be larger than C_μ. Beyond this time, it converges from above. Much of the detailed convergence behavior is determined, of course, by T's full eigenvalue spectrum. The interpretation just given, though, can be directly deduced by examining the reconstructed machine's graph. One aspect which is less immediate is that for SSSs the initial distribution relaxes through an infinite number of Cantor sets in sequence space. For SSFTs there is only a finite number of Cantor sets.

This structural analysis indicates that the ratio

$$\Delta C_\mu(t) = \frac{|C_\mu - C_\mu(t)|}{C_\mu} \tag{26}$$

is largely determined by the amount of information in the transient states. For SSSs this quantity only asymptotically vanishes since there are transient cycles in which information

* SSS shall also refer, in context, to stochastic Sofic systems.[44]

Fig. 8 Temporal convergence of the complexity $C_\mu(t)$ for a period 3 process P_3 (triangles), a Markovian process P_2 whose support is a subshift of finite type (circles), and a process P_1 that generates blocks of even numbers of 1s surrounded by 0s (squares).

persists for all time, even though their probability decreases asymptotically. This leads to a general definition of (chaotic or periodic) phase and phase locking. The phase of a machine at some point in time is its current state. There are two types of phase of interest here. The first is the process's phase and the second is the observer's phase. The latter refers to the state of the observer's model having read the data stream up to some time. The observer has β-locked onto the process when $\Delta C_\mu(t_{lock}) < \beta$. This occurs at the locking time t_{lock} which is the longest time t such that $\Delta C_\mu(t) = \beta$. When the process is periodic, this notion of locking is the standard one from engineering. But it also applies to chaotic processes and corresponds to the observer knowing what state the process is in, even if the next measurement cannot be predicted exactly.

These two classes of knowledge relaxation lead to quite different consequences for an observer even though the processes considered above all have a small number of states (2 or 3) and share the same single-symbol statistics: $Pr(s = 1) = \frac{2}{3}$ and $Pr(s = 0) = \frac{1}{3}$. In the over-damped case, the observer knows the state of the underlying process with certainty after a finite time. In the critically-damped situation, however, the observer has only approximate knowledge for all times. For example, setting $\beta = 1\%$ leads to locking times shown in table 1. Thus, the ability of an observer to infer the state depends crucially on the process's

Locking Times at 1% Level	
Process	**Locked at time**
Period 3	2
Isolated 0s	1
Even 1 blocks	17

Table 1 β-locking times for the periodic P_3, isolated 0s P_2, and even 1s P_1, processes. Note that for the latter the locking time is substantially longer and depends on β. For the former two, the locking times indicate the times at which asymptotic convergence has been achieved. The observer knows the state of the underlying process with certainty at those locking times. For P_1, however, at $t = 17$ the observer is partially phase-locked with knowledge of 99% of the process's state information.

computational structure, viz. whether its topological machine is a SSFT or a SSS. The presence of extrinsic noise and observational noise modify these conclusions systematically.

It is worthwhile to contrast the machine model of P_1 with a model based on histograms, or look-up tables, of the same process. Both models are given sufficient storage to exactly represent the length 3 sequence probability distribution. They are then used for predictions on length 4 sequences. The histogram model will store the probabilities for each length 3 sequence. This requires 8 bins each containing an 8 bit approximation of a rational number: 3 bits for the numerator and 5 for the denominator. The total is 67 bits which includes an indicator for the most recent length 3 sequence. The machine model of Fig. 9 must store the current state and five approximate rational numbers, the transition probabilities, using 3 bits each: one for the numerator and two for the denominator. This gives a model size of 17 bits.

Two observers, each given one or the other model, are presented with the sequence 101. What do they predict for the event that the fourth symbol is $s = 1$? The histogram model predicts

$$Pr(1|101) \approx Pr(1|01) = \frac{Pr(011)}{Pr(11)} = \frac{1/6}{4/9} = \frac{3}{8} \tag{27}$$

whereas the machine model predicts

$$Pr(1|101) = p_{\mathrm{C}\to\mathrm{B}} = 1 \tag{28}$$

The histogram model gives the wrong prediction. It says that the fourth symbol is uncertain when it is completely predictable. A similar analysis for the prediction of measuring $s = 1$ having observed 011 shows the opposite. The histogram model predicts $s = 1$ is more likely, $p_{s=1} = 2/3$, when it is, in fact, not predictable at all, $p_{s=1} = 1/2$. This example is illustrative of the superiority of stochastic machine models over histogram and similar look-up table models of time-dependent processes. Indeed, there are processes with finite memory for which no finite-size sequence histogram will give correct predictions.

In order to make the physical relevance of SSSs and their slow convergence more plausible, the next example is taken from the Logistic map at a Misiurewicz parameter value. The Logistic map is an iterated mapping of the unit interval

$$x_{n+1} = f_r(x_n) = rx_n(1 - x_n), \quad r \in [0, 4], x_0 \in [0, 1] \tag{29}$$

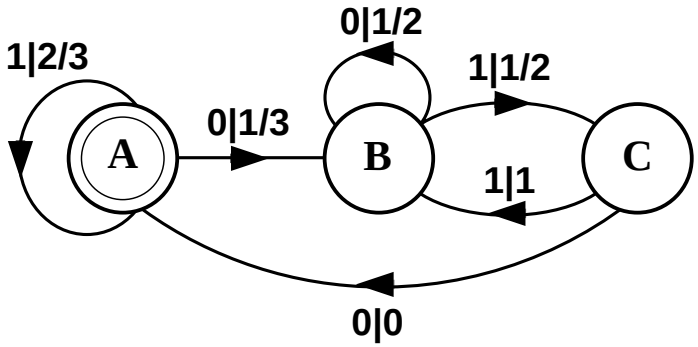

Fig. 9 The even system generates sequences $\left\{\ldots 01^{2n}0\ldots : n = 0, 1, 2, \ldots\right\}$ of 1s of even length, i.e. even parity. There are three states $\mathbf{V} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$. The state A with the inscribed circle is the start state v_0. The edges are labeled $s|p$ where $s \in \mathbf{A}$ is a measurement symbol and $p \in [0, 1]$ is a conditional transition probability.

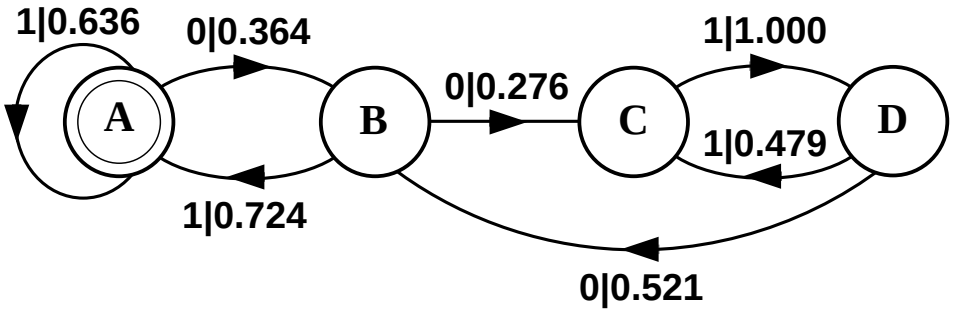

Fig. 10 The machine $M_{r'}^{\rightarrow}$ reconstructed by parsing in forward presentation order a binary sequence produced using a generating partition of the Logistic map at a Misiurewicz parameter value.

The control parameter r governs the degree of nonlinearity. At a Misiurewicz parameter value the chaotic behavior is governed by an absolutely continuous invariant measure. The consequence is that the statistical properties are particularly well-behaved. These parameter values are determined by the condition that the iterates $f^n(x_c)$ of the map's maximum $x_c = 1/2$ are asymptotically periodic. The Misiurewicz parameter value r' of interest here is the first root of $f_{r'}^4(x_c) = f_{r'}^5(x_c)$ below that at $r = 4$. Solving numerically yields $r' \approx 3.9277370017867516$. The symbolic dynamics is produced from the measurement partition $\Pi_{1/2} = \{[0, x_c], (x_c, 1]\}$. Since this partition is generating the resulting binary sequences completely capture the statistical properties of the map. In other words, there is a one-to-one mapping between infinite binary sequences and almost all points on the attractor.

Reconstructing the machine from one very long binary sequence in the direction in which the symbols are produced gives the four state machine $M_{r'}^{\rightarrow}$ shown in Fig. 10. The stochastic connection matrix is

$$T = \begin{pmatrix} 0.636 & 0.364 & 0.000 & 0.000 \\ 0.724 & 0.000 & 0.276 & 0.000 \\ 0.000 & 0.000 & 0.000 & 1.000 \\ 0.000 & 0.521 & 0.479 & 0.000 \end{pmatrix} \tag{30}$$

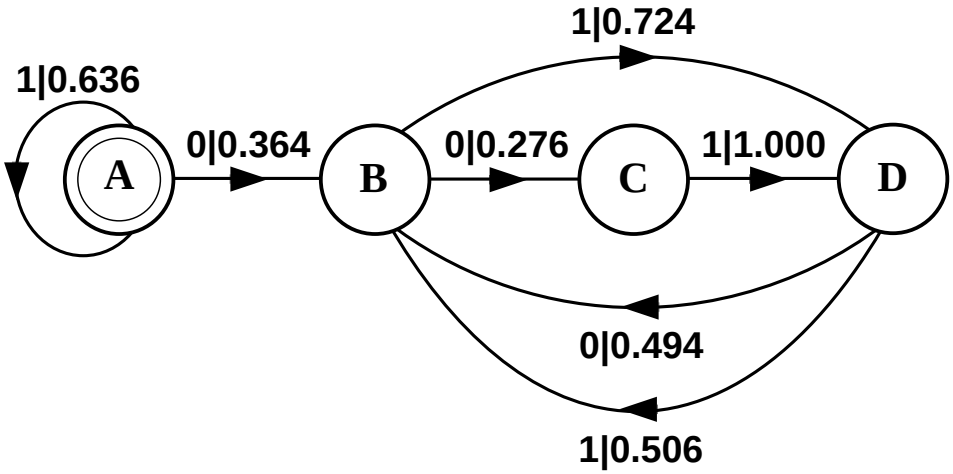

Fig. 11 The machine $M_{r'}^{\leftarrow}$ reconstructed by parsing in reverse presentation order a binary sequence produced using a generating partition of the Logistic map at a Misiurewicz parameter value.

Reconstructing the machine from the same binary sequence in the opposite direction gives the reverse-time machine $M_{r'}^{\leftarrow}$ shown in Fig. 11. It connection matrix is

$$T = \begin{pmatrix} 0.636 & 0.364 & 0.000 & 0.000 \\ 0.000 & 0.000 & 0.276 & 0.724 \\ 0.000 & 0.000 & 0.000 & 1.000 \\ 0.000 & 1.000 & 0.000 & 0.000 \end{pmatrix} \tag{31}$$

Notice that $M_{r'}^{\leftarrow}$ has a transient state and three recurrent states compared to the four recurrent states in $M_{r'}^{\rightarrow}$. This suggests the likelihood of some difference in complexity convergence. Figure 12 shows that this is the case by plotting $C_\mu(M_{r'}^{\rightarrow}, t)$ and $C_\mu(M_{r'}^{\leftarrow}, t)$ for positive and "negative" times, respectively. Not only do the convergence behaviors differ in type, but also in the asymptotic values of the complexities: $C_\mu(M_{r'}^{\rightarrow}) \approx 1.77$ bits and $C_\mu(M_{r'}^{\leftarrow}) \approx 1.41$ bits. This occurs despite the fact that the entropies must be and are the same for both machines: $h(M_{r'}^{\rightarrow}) = h(M_{r'}^{\leftarrow}) \approx 0.82$ bits per time unit and $h_\mu(M_{r'}^{\rightarrow}) = h_\mu(M_{r'}^{\leftarrow}) \approx 0.81$ bits per time unit. Although the data stream is equally unpredictable in both time directions, an observer learns about the process's state in two different ways and obtains different amounts of state information. The difference

$$\Delta C_{\leftarrow}^{\rightarrow} = C_\mu(M_{r'}^{\rightarrow}) - C_\mu(M_{r'}^{\leftarrow}) \approx 0.36 \text{ bits} \tag{32}$$

is a measure of the computational irreversibility of the process. It indicates the process is not symmetric in time from the observer's viewpoint. This example serves to distinguish machine reconstruction and the derived quantifiers, such as complexity, from the subsequence-based measures, such as the two-point mutual information and the excess entropy.

4. MEASUREMENT SEMANTICS

Shannon's communication theory tells one how much information a measurement gives. But what is the meaning of a particular measurement? Sufficient structure has been developed up to this point to introduce a quantitative definition of an observation's meaning. Meaning,

Fig. 12 What the observer sees, on average, in forward and reverse lag time in terms of the complexity convergence $C_\mu(t)$ for $M_{r'}^{\rightarrow}$ and $M_{r'}^{\leftarrow}$. Data for the latter are plotted on the negative lag time axis. Note that not only do the convergence characteristics differ between the two time directions, but the asymptotic complexity values are not equal.

as will be seen, is intimately connected with hierarchical representation.* The following, though, concerns meaning as it arises when crossing a single change in representation and not in the entire hierarchy.[11]

A universe consisting of an observer and a thing observed has a natural semantics. The semantics describes the coupling that occurs during measurement. The attendant meaning derives from the dual interpretation of the information transferred at that time. As already emphasized, the measurement is, first, an indirect representation of the underlying process's state and, second, information that updates the observer's knowledge. The semantic information processing that occurs during a measurement thus turns on the relationship between two levels of representation of the same event.

The meaning of a message, of course, depends on the context in which its information is made available. If the context is inappropriate, the observation will have no basis with which to be understood. It will have no meaning. If appropriate, then the observation will be "understood". And if that which is understood — the content of the message — is largely

* The following hopefully adds some specificity to approaches to the symbol grounding problem: How to physical states of nature take on semantic content?[45]

unanticipated then the observation will be more significant than a highly likely, "obvious" message.

In the present framework context is set by the model held by the observer at the time of a measurement. To take an example, assume that the observer is capable of modeling using the class of stochastic finite automata. And, in particular, assume the observer has estimated a stochastic finite automaton* and has been following the process sufficiently long to know the current state with certainty. Then at a given time the observer measures symbol $s \in \mathbf{A}$. If that measurement forces a disallowed transition, then it has no meaning other than that it lies outside of the contexts (morphs) captured in the current model. The observer clearly does not know what the process is doing. Indeed, formally the response is for the observer to reset the machine to the initial state of total ignorance. If, however, the measurement is associated with an allowed transition, i.e. it is anticipated, then the degree $\Theta(s)$ of meaning is

$$\Theta(s) = -\log p_{\underset{s}{\rightarrow} v} \tag{33}$$

Here $\underset{s}{\rightarrow} v$ denotes the machine state $v \in \mathbf{V}$ to which the measurement brings the observer's knowledge of the process's state. $p_{\underset{s}{\rightarrow} v}$ is the corresponding morph's probability which is given by the associated state's asymptotic probability. The meaning itself, i.e. the content of the observation, is the particular morph to which the model's updated state corresponds. In this view a measurement selects a particular pattern from a palette of morphs. The measurement's meaning is the selected morph† and the degree of meaning is determined by the latter's probability.

To clarify these notions, let's consider as an example a source that produces infinite binary sequences for the regular language[48] described by the expression $(0 + 11)^*$. We assume further that the choice implied by the "$+$" is made with uniform probability. An observer given an infinite sequence of this type reconstructs the stochastic finite machine shown in Fig. 9. The observer has discovered three morphs: the states $\mathbf{V} = \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$. But what is the meaning of each morph? First, consider the recurrent states B and C. State B is associated with having seen an even number of 1's following a 0; C with having seen an odd number. The meaning of B is "even" and C is "odd". Together the pair $\{\mathrm{B}, \mathrm{C}\}$ recognize a type of parity in the data stream. The machine as a whole accepts strings whose substrings of the form $01\ldots11\ldots10$ have even parity of 1s. What is the meaning of state A? As long as the observer's knowledge of the process's state remains in state A, there has been some number of 1's whose parity is unknown, since a 0 must be seen to force the transition to the

* Assume also that the estimated machine is deterministic in the sense of automata theory: the transitions from each state are uniquely labeled: $p_e = p(v, v', s) = p_v p_{v \underset{s}{\rightarrow} v'}$. This simplifies the discussion by avoiding the need to define the graph indeterminacy as a quantitative measure of ambiguity.[17] Ambiguity for an observer arises if its model is a stochastic nondeterministic automaton.

† I simplify here. The best formal representation of meaning at present uses the set-theoretic structure that the machine induces over the set of observed subsequences. This in turn is formulated via the lattice theory[46] of machines.[47]

parity state B. State A, a transient, serves to synchronize the recurrent states with the data stream. This indicates for this example the meaning content of an individual measurement in terms of the state to which it and its predecessors bring the machine.

Before giving a quantitative analysis the time dependence of the state probabilities must be calculated. Recall that the state probabilities are updated via the stochastic connection matrix

$$\vec{p}_{\mathbf{V}}(t+1) = \vec{p}_{\mathbf{V}}(t) \begin{pmatrix} \frac{2}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & 0 \end{pmatrix} \tag{34}$$

where $\vec{p}_{\mathbf{V}}(t) = (p_{\mathrm{A}}(t), p_{\mathrm{B}}(t), p_{\mathrm{C}}(t))$ and the initial distribution is $\vec{p}_{\mathbf{V}}(0) = (1, 0, 0)$. Using the z-transform, the time-dependent state probabilities are found to be

$$\begin{aligned} p_{\mathrm{A}}(t) &= \left(\frac{2}{3}\right)^t && t = 0, 1, 2, \ldots \\ p_{\mathrm{B}}(t) &= 2\left(\frac{2}{3}\right)^t - 2^{1-t} && t = 0, 1, 2, \ldots \\ p_{\mathrm{C}}(t) &= \begin{cases} \left(\frac{2}{3}\right)^t - 2^{1-t} & t = 1, 2, 3, \ldots \\ 0, & t = 0 \end{cases} \end{aligned} \tag{35}$$

Any time a disallowed transition is forced the current state is reset to the start state and $\vec{p}_{\mathbf{V}}(t)$ is reset to the distribution representing total ignorance which is given by $\vec{p}_{\mathbf{V}}(0)$.

What then is the quantitative degree of meaning of particular measurements? Let's consider all of the possibilities: all possible contexts, i.e. current states, and all possible measurements. t steps after a reset, the observer is

1. In the sync state and measures $s = 1$: $\Theta^t_{\mathrm{sync}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\mathrm{A}} = t(\log_2 3 - 1)$;
2. In the sync state and measures $s = 0$: $\Theta^t_{\mathrm{sync}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\mathrm{B}} = -\log_2 p_{\mathrm{B}}(t)$; e.g. $\Theta^1_{\mathrm{sync}}(0) = \log_2 3 \approx 1.584$ bits;
3. In the even state and measures $s = 1$: $\Theta^t_{\mathrm{even}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\mathrm{C}} = -\log_2 p_{\mathrm{C}}(t), t > 1$; e.g. $\Theta^2_{\mathrm{even}}(1) = \log_2 6 \approx 2.584$ bits;
4. In the even state and measures $s = 0$: $\Theta^t_{\mathrm{even}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\mathrm{B}} = -\log_2 p_{\mathrm{B}}(t)$; e.g. $\Theta^2_{\mathrm{even}}(0) = 1 + 2\log_2 3 - \log_2 7 \approx 1.372$ bits;
5. In the odd state and measures $s = 1$: $\Theta^t_{\mathrm{odd}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\mathrm{B}} = -\log_2 p_{\mathrm{B}}(t)$; e.g. $\Theta^3_{\mathrm{odd}}(1) = 2 + 3\log_2 3 - \log_2 37 \approx 1.545$ bits;
6. In the odd state and measures $s = 0$, a disallowed transition. The observer resets the machine: $\Theta^t_{\mathrm{odd}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\mathrm{A}} = -\log_2 p_{\mathrm{A}}(0) = 0$.

In this scheme states B and C cannot be visited at time $t = 0$ nor state C at time $t = 1$.

Assuming no disallowed transitions have been observed, at infinite time $\vec{p}_{\mathbf{V}} = \left(0, \frac{2}{3}, \frac{1}{3}\right)$ and the degrees of meaning are, if the observer is

1. In the sync state and measures $s = 1$: $\Theta_{\mathrm{sync}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\mathrm{A}} = \infty$;

Observer's Semantic Analysis of Parity Source				
Observer in State	**Measures Symbol**	**Interprets Meaning as**	**Degree of Meaning (bits)**	**Amount of Information (bits)**
A	1	Unsynchronized	Infinity	0.585
A	0	Synchronize	0.585	1.585
B	1	Odd number of 1s	1.585	1
B	0	Even number of 1s	0.585	1
C	1	Even number of 1s	0.585	0
C	0	Confusion: lose sync, reset to start state	0	Infinity

Table 2 The observer's semantics for measuring the parity process of Fig. 9.

2. In the sync state and measures $s = 0$: $\Theta_{\text{sync}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\text{B}} = \log_2 3 - 1 \approx 0.584$ bits;
3. In the even state and measures $s = 1$: $\Theta_{\text{even}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\text{C}} = \log_2 3 \approx 1.584$ bits;
4. In the even state and measures $s = 0$: $\Theta_{\text{even}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\text{B}} = \log_2 3 - 1 \approx 0.584$ bits;
5. In the odd state and measures $s = 1$: $\Theta_{\text{odd}}(1) = -\log_2 p_{\underset{1}{\rightarrow}\text{B}} = \log_2 3 - 1 \approx 0.584$ bits;
6. In the odd state and measures $s = 0$, a disallowed transition. The observer resets the machine: $\Theta_{\text{odd}}(0) = -\log_2 p_{\underset{0}{\rightarrow}\text{A}} = -\log_2 p_{\text{A}}(0) = 0$.

Table 2 summarizes this analysis for infinite time. It also includes the amount of information gained in making the specified measurement. This is given simply by the negative binary logarithm of the associated transition probability.

Similar definitions of meaning can be developed between any two levels in a reconstruction hierarchy. The example just given concerns the semantics between the measurement symbol level and the stochastic finite automaton level.[11] Meaning appears whenever there is a change in representation of events. And if there is no change, e.g. a measurement is considered only with respect to the population of other measurements, an important special case arises.

In this view Shannon information concerns degenerate meaning: that obtained within the same representation class. Consider the information of events in some set E of possibilities whose occurrence is governed by arbitrary probability distributions $\{P, Q, \ldots\}$. Assume that no further structural qualifications of this representation class are made. Then the Shannon self-information $-\log p_e, \quad p_e \in P$, gives the degree of meaning $-\log_2 p_{\underset{e}{\rightarrow}e}$ in the observed event e with respect to total ignorance. Similarly, the information gain $I(P;Q) = \sum_{e \in E} p_e \log_2 \frac{p_e}{q_e}$ gives the average degree of "meaning" between two distributions. The two representation levels are degenerate: both are the events themselves. Thus, Shannon information gives the degree of meaning of an event with respect to the set E of events and not with respect to an observer's internal model; unless, of course, that model is taken to be

the collection of events as in a histogram or look-up table. Although this might seem like vacuous re-interpretation, it is essential that general meaning have this as a degenerate case.

The main components of meaning, as defined above should be emphasized. First, like information it can be quantified. Second, conventional uses of Shannon information are a natural special case. And third, it derives fundamentally from the relationship *across* levels of abstraction. A given message has different connotations depending on an observer's model and the most general constraint is the model's level in a reconstruction hierarchy. When model reconstruction is considered to be a time-dependent process that moves up a hierarchy, then the present discussion suggests a concrete approach to investigating adaptive meaning in evolutionary systems: emergent semantics.

In the parity example above I explicitly said what a state and a measurement "meant". Parity, as such, is a human linguistic and mathematical convention, which has a compelling naturalness due largely to its simplicity. A low level organism, though, need not have such a literary interpretation of its stimuli. Meaning of (say) its model's states, when the state sequence is seen as the output of a preprocessor,* derives from the functionality given to the organism, as a whole and as a part of its environment and its evolutionary and developmental history. Said this way, absolute meaning in nature is quite a complicated and contingent concept. Absolute meaning derives from the global structure developed over space and through time. Nonetheless, the analysis given above captures the representation level-to-level origin of "local" meaning. The tension between global and local entities is not the least bit new to nonlinear dynamics. Indeed, much of the latter's subtlety is a consequence of their inequivalence. Analogous insights are sure to follow from the semantic analysis of large hierarchical processes.

5. CONCLUDING REMARKS

By way of summarizing the preceding discussion, there are a few points that can be brought out concerning what reconstructed machines represent. First, by the definition of future-equivalent states, they give the minimal information dependency between the morphs. In this respect, they represent the causality of the morphs considered as events. If state B follows state A then A is a cause of B and B is one effect of A. Second, the machines capture the information flow within the given data stream. Machine reconstruction produces minimal models up to the given approximation level; that is, up to the amount of data available for the estimation and up to the setting of the parameters (D, L, δ). This minimality guarantees that there are no other events (morphs) that intervene between successive states, at the given error level, to render A and B independent. In this and only this case, can one unambiguously say that information flows from A to B, under the chosen parsing direction. The amount of information that flows is given by the mutual information $I(\mathrm{A};\mathrm{B}) = H(\mathrm{B}) - H(\mathrm{B}|\mathrm{A})$ of observing the state-event A followed by state-event B. This criterion for information flow

* This preprocessor is a transducer version of the model that takes the input symbols and outputs strings in the state alphabet $\mathbf{V}$.

also extends to spatial systems. Finally, time is the natural ordering captured by machines. As noted in the section on knowledge relaxation changing the direction of parsing the data stream leads to a quantitative measure of a new type of irreversibility. This type of irreversibility is a consequence of the computational, in fact, semi-group, structure of the underlying process.

Causation appears then as an efficient organization of knowledge. It is a symmetry in the broadest sense of the word. It allows for the "factoring" of experience into lower dimensional representations. The conditional independence basis for machine reconstruction of states implies that ϵ-machines are the minimal causal representations of the source, up to the given approximation level. The hierarchical reconstruction procedure uses a slight, but significant extension of this in order to address the question of change model classes: An ϵ-machine for a process is the minimal causal representation reconstructed using the least powerful computational model class that yields a finite complexity.

At first glance the process of measurement, the elemental act of information acquisition, would seem to be an antecedent of causality. This impression is, though, a result of current nonphysical formulations of information theory. The essential physical contact during measurement calls into play the entire panoply of modeling and semantics laid out above. Information theory, as an engineering discipline, only considers the amount of information and rates of its production, loss, and transmission, as measured by various entropies. As such it ignores, as Shannon said it should, the question posed in the previous Woodward proceedings when considering the geometric structure of the space of information sources and the algebra of measurements: "what of the 'meaning' of ... information?"[49] The answer here, simply stated, is that it depends on the current state of knowledge of the observer. In restricting the class of models to stochastic finitary automata, a concrete answer was given in terms of the discovered morphs and the observer's anticipation of seeing them. A more general answer, though, lies in clearly delineating the resources, computational and observational, available to the modeler. Lending this large a context to a quantitative theory of meaning, though, pushes the boundaries especially of theoretical computer science, from which we must know how space and time complexities trade-off against one another. At present, these scalings appear to be largely, if not exclusively, studied as separate coordinates for the space of computational tasks.[48,50] Future progress demands a more detailed understanding of the complexity-entropy landscape over the space of computational tasks. There are some provocative hints from phenomenological studies that phase transitions organize some aspects of this space.[19,51,52]

What is the role of nonlinearity in all of this? I would claim at this point that it is much more fundamental than simply providing an additional and more difficult exercise in building good models and formalizing what is seen. Rather it goes to the very heart of genuine discovery. Let me emphasize this by way of a contrapositive point.

Linear systems are meaningless in the sense that the question of semantics need not arise within their universe of discourse. The implementations, artificial or natural, of mechanisms by which a linear system could be inferred are themselves nonlinear. Said another way there

is no finite linear system that learns stably. Therefore, (finite) linear systems as a class are not self-describing; the universe is open. This indirect observation suggests that there is an intimate connection between meaning and nonlinearity, since nontrivial semantics requires the ability to model.

The utility of complexity, when it is seen as the arbiter of order and randomness, within nonlinear physics can be illustrated by way of posing some final questions. Consider the evolution of scientific theories. In particular, focus on the theories of time-dependent physical processes, viz. Laplacian-Newtonian classical mechanics (LNCM) and Copenhagen quantum mechanics (CQM). In LNCM (local) determinism implies complete (local) predictability. In CQM (local) nondeterminism implies complete (local) unpredictability and so only statistical regularity can be present. Shrödinger's equation is, in fact, the equation of motion governing that statistical regularity. In light of the preceding discussion of modeling, it is rather curious, but probably no coincidence, that these two theories come from the two extremes of entropy.

The meaning of physical events is often couched only in terms of these two contenders via (say) Bohr's notion of complementarity. But what about the intermediate possibility: complexity? Could these two opposed views be incomplete aspects, or projections, of a complex nature? A nature too intricate and too detailed to be completely understood at any one time and with finite knowledge? Whose state is too complex to determine with any finite measurement?

It seems that such issues will only be given their proper framing when we understand how physical nature intrinsically computes and how subsystems would spontaneously take up the task of modeling. Then, of course, there is the nonlinear dynamics of this process. Could this natural complexity be a state to which a system evolves spontaneously? A true self-organized complexity?

Hopefully, the preceding made it clear that to ignore the central role of computation is to miss the point of these questions entirely.

Many thanks to the Santa Fe Institute, where the author was supported by a Robert Maxwell Foundation Visiting Professorship, for the warm hospitality during the writing of the present essay. Funds from NASA-Ames University Interchange NCA2-488 and the AFOSR also contributed to this work.

REFERENCES

1. N. Chomsky, "Three models for the description of language," *IRE Trans. Info. Th.*, vol. 2, p. 113, 1956.
2. A. O'Hear, *An Introduction to the Philosophy of Science*. Oxford: Oxford University Press, 1989.
3. D. M. MacKay, *Information, Meaning and Mechanism*. Cambridge: MIT Press, 1969.

4. S. Omohundro, “Modelling cellular automata with partial differential equations,” *Physica*, vol. 10D, p. 128, 1984.
5. J. P. Crutchfield, “Turing dynamical systems.” unpublished, 1987.
6. L. Blum, M. Shub, and S. Smale, “On a theory of computation over the real numbers,” *Bull. AMS*, vol. 21, p. 1, 1989.
7. C. Moore, “Unpredictability and undecidability in dynamical systems,” *Phys. Rev. Lett.*, vol. 64, p. 2354, 1990.
8. D. C. Dennett, *The Intentional Stance*. Cambridge: MIT Press, 1987.
9. J. P. Crutchfield and B. S. McNamara, “Equations of motion from a data series,” *Complex Systems*, vol. 1, p. 417, 1987.
10. J. Rissanen, *Stochastic Complexity in Statistical Inquiry*. Singapore: World Scientific, 1989.
11. J. P. Crutchfield, “Reconstructing language hierarchies,” in *Information Dynamics* (H. A. Atmanspracher and H. Scheingraber, eds.), (New York), p. 45, Plenum, 1991.
12. C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*. Champaign-Urbana: University of Illinois Press, 1962.
13. A. N. Kolmogorov, “A new metric invariant of transient dynamical systems and automorphisms in Lebesgue spaces,” *Dokl. Akad. Nauk. SSSR*, vol. 119, p. 861, 1958. (Russian) Math. Rev. vol. 21, no. 2035a.
14. J. P. Crutchfield and N. H. Packard, “Symbolic dynamics of noisy chaos,” *Physica*, vol. 7D, p. 201, 1983.
15. N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, “Geometry from a time series,” *Phys. Rev. Let.*, vol. 45, p. 712, 1980.
16. H. Poincaré, *Science and Hypothesis*. New York: Dover Publications, 1952.
17. J. P. Crutchfield and K. Young, “Inferring statistical complexity,” *Phys. Rev. Let.*, vol. 63, p. 105, 1989.
18. J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. New York: Morgan Kaufmann, 1988.
19. J. P. Crutchfield and K. Young, “Computation at the onset of chaos,” in *Entropy, Complexity, and the Physics of Information* (W. Zurek, ed.), vol. VIII of *SFI Studies in the Sciences of Complexity*, (Reading, Massachusetts), p. 223, Addison-Wesley, 1990.
20. A. Fraser, “Using hidden Markov models to predict chaos.” preprint, 1990.
21. L. R. Rabiner, “A tutorial on hidden Markov models and selected applications,” *IEEE Proc.*, vol. 77, p. 257, 1989.
22. J. P. Crutchfield, *Noisy Chaos*. PhD thesis, University of California, Santa Cruz, 1983. Published by University Microfilms Intl, Ann Arbor, Michigan.
23. S. Wolfram, “Computation theory of cellular automata,” *Comm. Math. Phys.*, vol. 96, p. 15, 1984.

24. P. Grassberger, "Toward a quantitative theory of self-generated complexity," *Intl. J. Theo. Phys.*, vol. 25, p. 907, 1986.

25. K. Lindgren and M. G. Nordahl, "Complexity measures and cellular automata," *Complex Systems*, vol. 2, p. 409, 1988.

26. J. P. Crutchfield and K. Young, "ϵ-machine spectroscopy." preprint, 1992.

27. C. H. Bennett, "Dissipation, information, computational complexity, and the definition of organization," in *Emerging Syntheses in the Sciences* (D. Pines, ed.), Redwood City: Addison-Wesley, 1988.

28. A. N. Kolmogorov, "Three approaches to the concept of the amount of information," *Prob. Info. Trans.*, vol. 1, p. 1, 1965.

29. J. P. Crutchfield and N. H. Packard, "Noise scaling of symbolic dynamics entropies," in *Evolution of Order and Chaos* (H. Haken, ed.), (Berlin), p. 215, Springer-Verlag, 1982.

30. N. H. Packard, *Measurements of Chaos in the Presence of Noise*. PhD thesis, University of California, Santa Cruz, 1982.

31. R. Shaw, *The Dripping Faucet as a Model Chaotic System*. Santa Cruz, California: Aerial Press, 1984.

32. C. P. Bachas and B. A. Huberman, "Complexity and relaxation of hierarchical structures," *Phys. Rev. Let.*, vol. 57, p. 1965, 1986.

33. G. Chaitin, "On the length of programs for computing finite binary sequences," *J. ACM*, vol. 13, p. 145, 1966.

34. A. Lempel and J. Ziv, "On the complexity of individual sequences," *IEEE Trans. Info. Th.*, vol. IT-22, p. 75, 1976.

35. J. Rissanen, "Stochastic complexity and modeling," *Ann. Statistics*, vol. 14, p. 1080, 1986.

36. W. H. Zurek, "Thermodynamic cost of computation, algorithmic complexity, and the information metric." preprint, 1989.

37. J. Ziv, "Complexity and coherence of sequences," in *The Impact of Processing Techniques on Communications* (J. K. Skwirzynski, ed.), (Dordrecht), p. 35, Nijhoff, 1985.

38. A. A. Brudno, "Entropy and the complexity of the trajectories of a dynamical system," *Trans. Moscow Math. Soc.*, vol. 44, p. 127, 1983.

39. J. P. Crutchfield, "Inferring the dynamic, quantifying physical complexity," in *Measures of Complexity and Chaos* (N. B. Abraham, A. M. Albano, A. Passamante, and P. E. Rapp, eds.), (New York), p. 327, Plenum Press, 1990.

40. R. Shaw, "Strange attractors, chaotic behavior, and information flow," *Z. Naturforsh.*, vol. 36a, p. 80, 1981.

41. J. P. Crutchfield, "Semantics and thermodynamics," in *Nonlinear Modeling and Forecasting* (M. Casdagli and S. Eubank, eds.), vol. XII of *Santa Fe Institute Studies in the Sciences of Complexity*, (Reading, Massachusetts), p. 317, Addison-Wesley, 1992.

42. R. Rammal, G. Toulouse, and M. A. Virasoro, “Ultrametricity for physicists,” *Rev. Mod. Phys.*, vol. 58, p. 765, 1986.
43. B. Marcus, “Sofic systems and encoding data,” *IEEE Transactions on Information Theory*, vol. 31, p. 366, 1985.
44. B. Kitchens and S. Tuncel, “Finitary measures for subshifts of finite type and sofic systems,” *Memoirs of the AMS*, vol. 58, p. no. 338, 1985.
45. S. Harnad, “The symbol grounding problem,” *Physica*, vol. 42D, p. 335, 1990.
46. G. Birkhoff, *Lattice Theory*. Providence: American Mathematical Society, third ed., 1967.
47. J. Hartmanis and R. E. Stearns, *Algebraic Structure Theory of Sequential Machines*. Englewood Cliffs: Prentice-Hall, 1966.
48. J. E. Hopcroft and J. D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. Reading: Addison-Wesley, 1979.
49. J. P. Crutchfield, “Information and its metric,” in *Nonlinear Structures in Physical Systems - Pattern Formation, Chaos and Waves* (L. Lam and H. C. Morris, eds.), (New York), p. 119, Springer-Verlag, 1990.
50. M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York: W. H. Freeman, 1979.
51. C. G. Langton, “Computation at the edge of chaos: Phase transitions and emergent computation,” in *Emergent Computation* (S. Forrest, ed.), p. 12, Amsterdam: North-Holland, 1990.
52. W. Li, N. H. Packard, and C. G. Langton, “Transition phenomena in cellular automata rule space,” *Physica*, vol. 45D, p. 77, 1990.