Finite statistical complexity for sofic systems

Nicolás Perry[1] and P.-M. Binder[1,2,*]
[1]*Departamento de Física, Universidad de Los Andes, Apartado Aéreo 4976, Bogotá, Colombia*
[2]*Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501*
(Received 19 May 1998; revised manuscript received 19 March 1999)

We propose a measure of complexity for symbolic sequences, which is based on conditional probabilities, and captures computational aspects of complexity without the explicit construction of minimal deterministic finite automata (DFA). Moreover, if the sequence is obtained from a dynamical system through a suitable encoding and its equations of motion are known, we show how to estimate the regions of phase space that correspond to computational states with statistically equivalent futures (causal states).
[S1063-651X(99)11707-0]

PACS number(s): 05.45.Gg, 02.50.Ga, 89.80.+h

Many approaches have been proposed to define and quantify the complexity of finite-alphabet strings which arise, for example, from observations of dynamical systems or spatially extended systems [1–16] through a suitable encoding [17]. These approaches have been carefully organized in a hierarchy [2] that has topological exponents and scaling dynamics measures at the highest level. In the present paper, we will propose a measure more related to Crutchfield's statistical complexity (C_μ) [6–8] and Grassberger's set complexity (SC) [10], both of which apply to systems that can be modeled by deterministic finite automata (DFA) also called sophic systems. This is a natural first step in the understanding of a complex system, since it corresponds to the simplest class of computational languages, but it does not apply to natural systems with intrinsically parallel dynamics (see Ref. [2], pp. 82 and 83). Both C_μ and SC define complexity with use of a minimal metric DFA (i.e., one in which transition probabilities between nodes are recorded), that represents a shift dynamical system, and each presents some problems in the construction of the DFA, as will be explained below. In this paper we offer a prescription which avoids the explicit construction of the DFA, and which yields finite complexity values for sofic systems; these coincide with regular languages: see Ref. [2], pp. 80 and 81. Instead of a DFA, we rely on equivalence classes of left strings of length l which lead to similar distributions of right strings of length r, defined to be equivalent within some tolerance ε. Our complexity measure $C_\phi(l,r,\varepsilon)$, which we call finite statistical complexity, is finite for systems with finite average internal memory. Moreover, it converges to the above-mentioned measures for systems properly described by a DFA. Our measure leads to representations of causal states (the states of the DFA) as sets of strings of length l, which correspond to the regions of phase space which have *statistically* the same futures for the following r time steps (or, equivalently, for r transitions of the DFA). If the dynamical system is known, we show how these regions can be estimated by running the system backwards for $l-1$ time steps, and keeping track of the boundaries.

To find C_ϕ, we begin with a stationary symbolic sequence of length $M \gg 1$ obtained from a suitable encoding of a trajectory of the system [17]. We then scan subsequences of length L. We further divide each such sequence into two parts, left and right, corresponding to the initial l and final r symbols, respectively, so that $l+r=L$. In the following discussion the term ''left'' will be interchanged freely with ''before,'' and so will ''right'' with ''after.'' We estimate the occurrence frequency of the left subsequences x_L, $P(x_L)$, and the occurrence frequency of right subsequences x_R that follow each left subsequence, $P(x_R|x_L)$ (conditional probability). Equivalence classes can be defined over the x_L subsequences by determining which produce the same distributions of x_R subsequences. Once the x_L subsequences are grouped into equivalence classes, that we represent by $\{x_L\}_i$, one can calculate the probabilities $P(\{x_L\}_1), P(\{x_L\}_2), \ldots, P(\{x_L\}_K)$ of the K equivalence classes of x_L [also equal to the probabilities of distributions of x_R that follow each equivalence class of x_L, $\rho_i(x_R|\{x_L\})$]. For example, if the strings 01010011, 01011001, 10000011, and 10001001 all appear with probability 0.01, and no other subsequence begins with 0101 or 1000, then the x_L subsequences 0101 and 1000 both lead to the x_R distribution (0011,1001), each right subsequence with probability 0.5, and therefore are in the same equivalence class. If no other x_L leads to the same right conditional distribution, the probability of this equivalence class is 0.04. The finite statistical complexity C_ϕ is then defined as

$$C_\phi = -\sum_{i=1}^{K} P(\{x_L\}_i)\log_2 P(\{x_L\}_i). \tag{1}$$

It must be noted that defining equivalence classes requires a regrouping of probability distributions of subsequences. The most natural choice defines two distributions to be related if they both have the same support, and the probabilities of each and every sequence in both distributions lie within an arbitrary difference ε, which we call tolerance. The value of C_ϕ obtained depends on ε [18], as expected. Other

*Author to whom correspondence should be addressed. Address correspondence to Departamento de Física, Universidad de Los Andes, A.A. 4976, Bogotá, Colombia. Electronic address: p@faoa.uniandes.edu.co

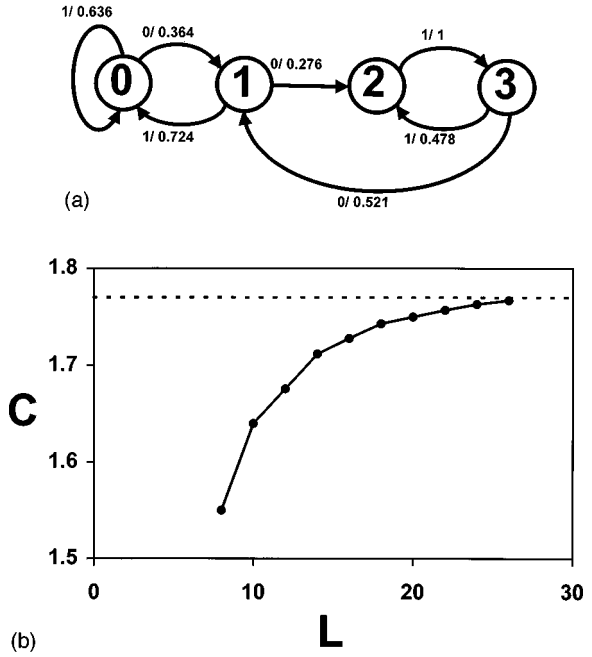

FIG. 1. (a) DFA without transient states; and (b) C_ϕ shown as a function of L, showing the convergence properties of C_ϕ. One million bits generated by the automaton in (a) were used to generate (b). The exact result for infinite L is $C_\mu = SC \sim 1.77$.

equivalence relations can be used for specific problems. For example, if one sequence appears in one distribution with a very small probability, it could be ignored, and considered as noise. Therefore, C_ϕ leaves the definition of equivalence relation relatively open. In our calculations we used the most tolerant equivalence relation: two distributions are considered equal if they have the same subsequences, regardless of their probabilities. This procedure for calculating C_ϕ does not rely on the construction of a DFA; instead, the measure

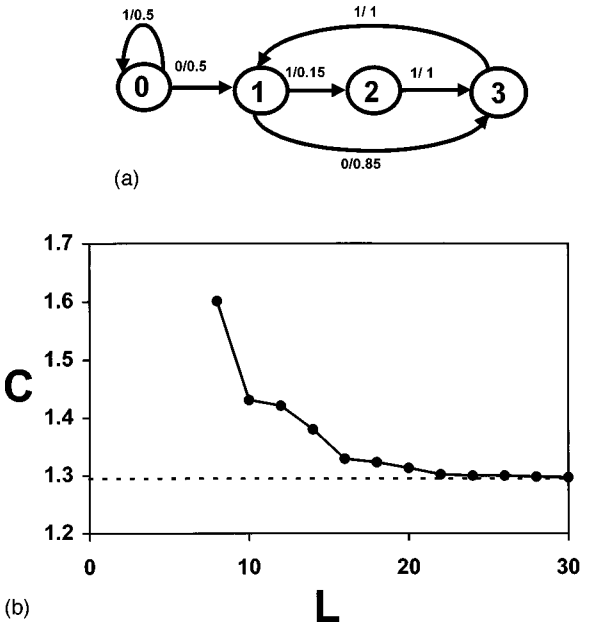

FIG. 2. (a) A DFA with a transient state and (b) C_ϕ vs L, showing convergence to the exact result, $C_\mu = SC \sim 1.295$.

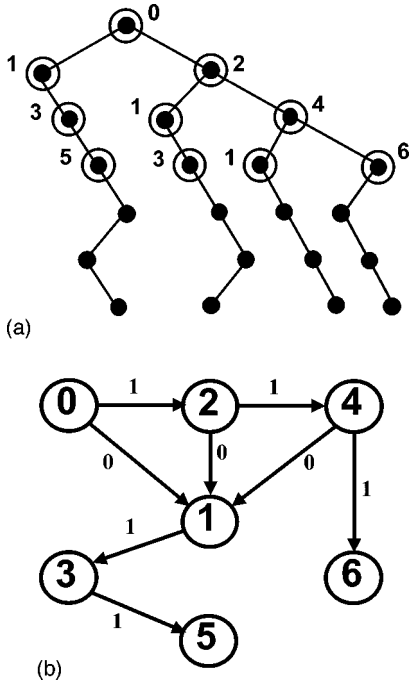

FIG. 3. (a) Binary tree and (b) ϵ machine constructed with the method of Crutchfield and Young for period 4 string given in the text, using sequences of $L = 6$ and subtrees of height 3. There are two dangling states (5 and 6). For $L \geq 8$ (and subtrees of height $L/2$) the reconstructed ϵ machine correctly describes the system.

is calculated directly from sequence statistics: this ensures that C_ϕ always yields a result.

In general, the number of equivalence classes inferred increases with increasing l, but too short an r reduces the number of sequences in a distribution, probably reducing

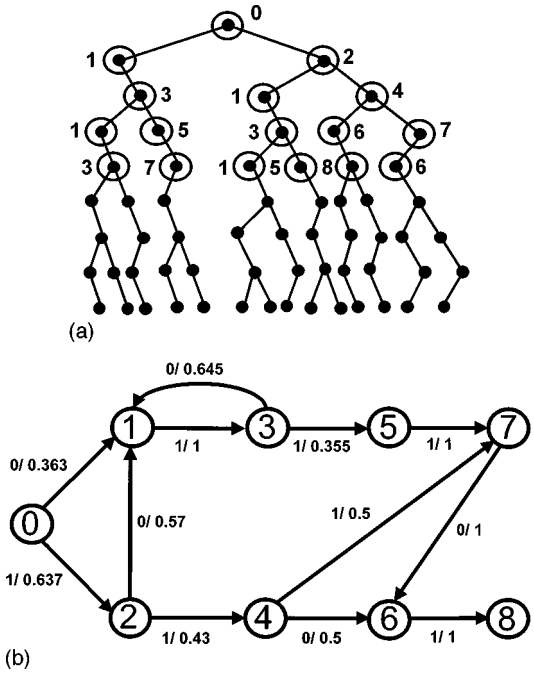

FIG. 4. (a) Binary tree and (b) ϵ machine with a dangling state for a language produced by the logistic map with $r = 3.6$, which is in the chaotic regime. We used $L = 8$ and subtrees of height 4. Note the dangling state [Eq. (8)].

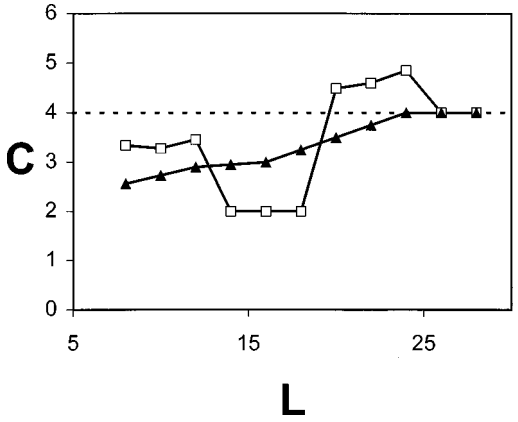

FIG. 5. Comparison of C_ϕ (triangles) and C_μ (squares) for a string of period 16, obtained with $L<32$. The latter were obtained by reconnecting dangling states to the initial node, whenever necessary. For $L\geqslant 32$, $C_\phi = C_\mu = 4$.

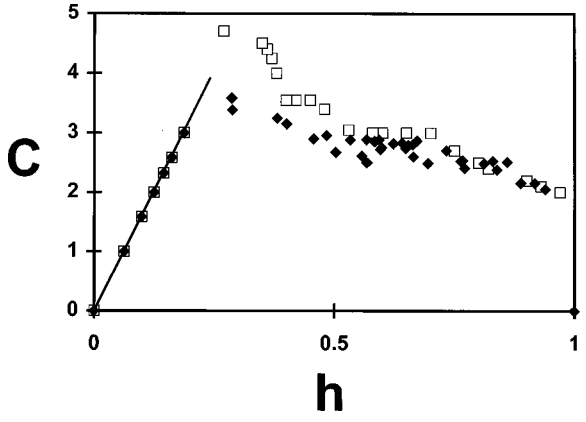

FIG. 6. Comparison of C_ϕ (diamonds) and C_μ (squares), both plotted against entropy density; points come from random samplings of the logistic map. The latter are shown only if no dangling states were present.

classes instead. In subsequent calculations we use $l=r=L/2$; compare the choice of subtree depth in Ref. [7], also discussed below.

To interpret statistical complexity, we recall that the information contained or generated by a source which produces J different symbols each with probability P_i is $I=-\Sigma_{i=1}^{J} P_i \log_2(P_i)$. In our case, each of K equivalent classes of left sequences gives rise to exactly one distribution of right sequences. Equation (1) gives the mean number of bits of internal memory from the left subsequences (before) needed to know which distribution we have in the right subsequences (after). As $L\to\infty$, the x_R distribution is for all the future; that is, each distribution of x_R sequences is analog to an infinite metric subtree (see Ref. [7]), and the corresponding equivalence class of x_L sequences is analog to a causal state. In this limit, C_ϕ becomes an intensive quantity [8], independent of r, l, and L, which measures the *average* embedded (internal) memory of the system, as long as the system is sofic. If the system is not sofic, C_ϕ will diverge with L. Consider the following case: if no regrouping is possible (for example, if ε is too small), there would be $\sim 2^{hl}$ equivalence classes, where h is the block entropy. If the system is not sofic, there would be no guarantee that the number of distributions of x_R (same number as equivalence classes) stops growing, because there are infinite follower sets (Ref. [2], p. 80, and references therein). Then, the number of equivalence classes scales as $2^{h_0 l}$, where h_0 is the Shannon entropy, and C_ϕ scales as l, i.e., it grows without bound.

We have studied two known DFA's, one with and one without transients, to compare finite statistical complexity with previous statistical complexity measures (SC and C_μ). The former are directly obtainable from sequence statistics in which we have used strings of length one million, while the latter can be calculated exactly from the configuration of the DFA. The results are shown in Figs. 1 and 2. There is very good convergence for large L of the finite statistical complexity to the value of the previous definitions, approaching from above and from below, respectively.

The limit cases of C_ϕ agree with the intuitive notion of complexity as internal memory: for a constant series (all zeros or all ones) there is only one right distribution, and hence one equivalence class in x_L. The memory requirement is zero. This is also the case for a totally random series, where no rules or patterns can be inferred, and in which the only right distribution is that all possible sequences are equally probable. For a system of period P one needs P distributions of right sequences and as many left sequences (causal states), which indicate the phase of the system. This corresponds to $\ln P$ bits of memory, as long as $L\geqslant 2P$; this indicates that nonzero information (e.g., about the phase of the system) is needed to predict its future [19]. However, nonzero complexity for periodic sequences does not meet the requirements for higher hierarchical definitions (see Ref. [2], p. 255), which can be a serious shortcoming.

These limits also agree with the definitions of Grassberger and of Crutchfield and Young (CY), which we review now. SC is defined as the information stored in the minimal DFA

TABLE I. Column 1: causal state; column 2: x_L sequences in each equivalence class; column 3: strings present in each $\{x_R\}$ distribution. The string was generated with the dynamical system $x_{n+1}=3.9x_n(1-x_n)$, scanned with $l=r=5$. The language presents three forbidden words up to length 10: 000, 0011, and 0010100101.

State	x_L sequences	x_R distribution
1	00101	1*1**,011**,*101*,*1001,00100,10010
2	**100	101**,10010
3	*1*11,*1101,101*1	1*1**,011**,*101*,*1001,0010*,10010
4	*1*10,10*10	1*1**,*101*,*1001,10010
5	*1001	011**,0101*,0010*,01001

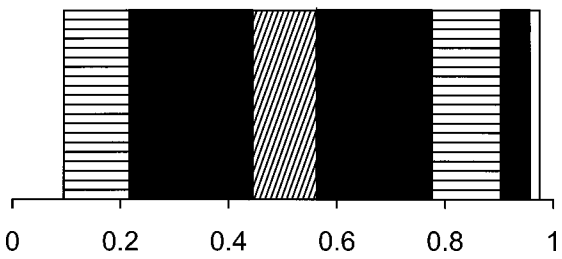

FIG. 7. Phase space of causal states for logistic map with $r=3.9$, sampled with $l=r=4$.

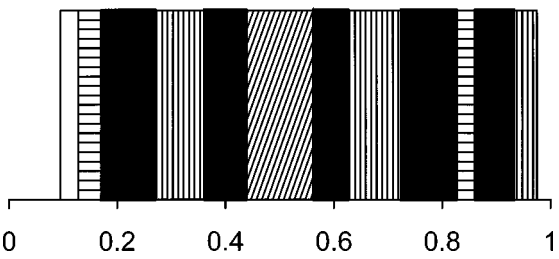

FIG. 8. Phase space of causal states for logistic map with $r=3.9$, sampled with $l=r=5$. The correspondence with Table I is as follows: state 1, white; state 2, horizontal stripes; state 3, black; state 4, vertical stripes; state 5, diagonal stripes.

that reproduces the behavior of a system, and is found [10] through the identification of irreducible forbidden words (see also Ref. [2]) in the language associated with the system. However, this algorithm may fail in constructing the minimal DFA, so its use is limited.

Instead of this, CY used the following method, which they claimed [7] yields the minimal DFA for a system: from the different subsequences of length L found in a long binary string, a binary tree of depth L is constructed in which each path from the root (top) to the bottom of the tree corresponds to an existing subsequence, in which a left (right) arch indicates a zero (one). Metric information on the transitions to lower branches is also kept. Then topologically and metrically equal subtrees of a given depth (typically $L/2$) are grouped into equivalence classes, called causal states. These, along with the transition probabilities between causal states, give rise to the DFA which describes the system (''ϵ machine'' in CY language). A stochastic connectivity matrix is constructed for the causal states, and the statistical complexity is defined as the entropy of the normalized eigenvector corresponding to the eigenvalue 1 of this matrix, whose elements correspond to the probabilities of being in each of the causal states.

This procedure has a problem. Often subtrees appear in the last scanned level of the binary tree, typically at a depth $L/2$, which do not belong to a previously known equivalence class. These give rise to terminal states in the DFA (dangling states in the CY language), which make the connectivity matrix singular, unless one makes the leap of faith of associating the new subtrees to an apparently similar equivalence class, or unless one reconnects them to the starting node [see Fig. 1(a) in Ref. [7]]. We have found that this problem happens very often, for both periodic and chaotic systems. A simple example is the analysis of a string of period P with subsequences of length $L<2P$. We illustrate this with the period 4 string 10111011 . . . , which we try to describe with subsequences of $L=6$. In Fig. 3 we show the binary tree and associated DFA obtained by this method, which is explained in more detail in Ref. [7]. Note that no arrows leave states 5 and 6, and therefore the connectivity matrix will have two empty columns, leading to a meaningless vector of state probabilities. The DFA for all $L\geqslant 8$ will look like Fig. 3(b), but with additional arrows pointing from state 5 to 6 (with output 1), and from state 6 to 1 (with output 0), correctly describing the period 4 of the sequence. A more complicated example is shown in Fig. 4 for a chaotic system. We stress that this is an inherent limitation of the CY method, which is trying to infer DFA's of arbitrary size with a finite microscope of length L.

Since our definition of complexity does not require the explicit construction of a DFA or a search for forbidden strings, it usually yields a reasonable estimation of statistical complexity. Again, as a simple example, in Fig. 5 we compare our method with the CY method, in which dangling states have been reconnected to the starting state in order to obtain a finite answer. We have used a string with period 16. The convergence properties of our method are clearly smoother. As a final example, Fig. 6 shows both C_ϕ (for $L=16$) and C_μ (for $L=32$, as shown in Ref. [7]) vs block entropy. We have taken as our dynamical system the logistic map, with many values of the nonlinearity parameter r. Our initial strings are typically several million bits long. The similarity of the two measures is evident for reasonably low complexity values. Unfortunately, for high complexity, the results depend strongly on L. We are unable to extend our results to $L=32$ because of our computer.

We now show how we can estimate the regions of phase space that correspond to each of the causal states, by finding sets of x_L strings that at least for r time steps lead statistically to the same futures (distributions of x_R); this only works if the equation of motion of the system happens to be known, and the estimates become better as one increases both l and r. We illustrate this with our canonical example, the logistic equation, for which we have chosen $r=3.9$. Table I shows the $l=5$ strings that lead to the $r=5$ x_R distributions shown in the last column. We have used the asterisk (*) to denote a wild card; that is, it can be replaced by either zero or 1. For example, causal state 5 has x_L strings 01001 or 11001.

To find an estimate to a causal state, we consider all the left strings in an equivalence class. We iterate backwards the map that governs the system $l-1$ times for each one of them, keeping track of the region of phase space consistent with the symbolic dynamics given by each and every x_L. Then the causal state representation in phase space is the union of all these regions. We show examples of where the causal states are located in phase space in Figs. 7 and 8. The figures correspond to $l=r=4$ and $l=r=5$, respectively. The two figures illustrate the size dependence of our estimate. The data in Table I were used to obtain Fig. 8. We note that representing causal states in phase space has been an open problem until now: see Figs. 4 and 5 of Ref. [20].

In this work we have presented a new way of calculating the statistical complexity of observed data sets representable with symbolic dynamics. Our method is based on conditional probabilities of sequences of bits in a long string obtained

from observations, and can be interpreted as the amount of embedded memory in the dynamical system. Our measure gives a result for any finite reconstruction size, and avoids problems present in the measures of Refs. [7] and [10], and is particularly useful for sofic systems, representable by a DFA. In the large sampling size (L) limit, our results agree with those obtained with these previous measures.

While our method does not yield explicitly the DFA for the system, our causal states and right distributions are expressed in terms of sets of strings, which has advantages. For example, if we know the equation of motion for the system, we can estimate which points of phase space correspond to specific causal states, i.e., share a statistically equivalent future. This leads us to believe that the string set representation can have other advantages, especially when combined with judicious use of the tolerance parameter ε, for example, in the study of noisy data sets or small data sets with large statistical fluctuations.

This work was supported by the IDB and Colciencias (Contract No. 259-96). We thank Centro MOX de Computación Avanzada at Universidad de Los Andes for the use of its Cray J916, and K. Young, D. P. Feldman, and J. P. Crutchfield for useful suggestions on this work.

[1] *Measures of Complexity and Chaos*, edited by N. B. Abraham, A. M. Albano, A. Passamante, and P. E. Rapp (Plenum, New York, 1989).

[2] R. Badii and A. Politi, *Complexity: Hierarchical Structure and Scaling in Physics* (Cambridge University Press, Cambridge, 1997).

[3] R. Badii and A. Politi, Phys. Rev. Lett. **78**, 444 (1997).

[4] C. H. Bennett, Found. Phys. **16**, 585 (1986).

[5] G. Chaitin, *Information, Randomness and Incompleteness* (World Scientific, Singapore, 1987).

[6] J. P. Crutchfield and N. H. Packard, Physica D **7**, 201 (1983).

[7] J. P. Crutchfield and K. Young, Phys. Rev. Lett. **63**, 105 (1989); J. P. Crutchfield, in *Nonlinear Modeling and Forecasting*, edited by M. Casdagli and S. Eubank (Addison-Wesley, Reading, MA, 1992), p. 317.

[8] J. P. Crutchfield and D. P. Feldman, Phys. Rev. E **55**, R1239 (1997); D. P. Feldman and J. P. Crutchfield, Phys. Lett. A **238**, 244 (1998).

[9] G. D'Alessandro and A. Politi, Phys. Rev. Lett. **64**, 1609 (1990).

[10] P. Grassberger, Int. J. Theor. Phys. **25**, 907 (1986); P. Grassberger, Z. Naturforsch. **43a**, 671 (1988).

[11] R. Landauer, Nature (London) **336**, 306 (1988).

[12] M. Li and P. M. B. Vitanyi, *An Introduction to Kolmogorov Complexity and its Applications* (Springer, New York, 1993).

[13] W. Li, Complex Syst. **5**, 381 (1991).

[14] K. Lindgren and M. G. Nordahl, Complex Syst. **2**, 409 (1988).

[15] S. Lloyd and H. Pagels, Ann. Phys. (N.Y.) **188**, 186 (1988).

[16] C. H. Papadimitriou, *Computational Complexity* (Addison-Wesley, Reading, MA, 1994).

[17] V. M. Alekseev and M. V. Yakobson, Phys. Rep. **75**, 287 (1981).

[18] We observe that more tolerance in the definition of equivalence classes results in lower values of complexity. This is reasonable, since higher tolerance leads to fewer equivalence classes; see N. Perry, undergraduate thesis, Universidad de Los Andes, 1998.

[19] Consider a string with a very large period P. When sampled with sequences of increasing L, the statistical complexity grows with L until $L>2P$, at which point the periodicity is recognized and complexity stabilizes at $\log_2 L$; see Fig. 5. Having zero complexity for periodic sequences would cause a discontinuity in a complexity vs L plot, located at $L=2P$.

[20] J. P. Crutchfield and C. R. Shalizi, Phys. Rev. E **59**, 275 (1999).