Eur. Phys. J. B (2014)
DOI: 10.1140/epjb/e2014-41018-5

THE EUROPEAN PHYSICAL JOURNAL B

Regular Article

Mixing Bandt-Pompe and Lempel-Ziv approaches: another way to analyze the complexity of continuous-state sequences

S. Zozor[1,2,a], D. Mateos[3], and P.W. Lamberti[3]

[1] Laboratoire Grenoblois d'Image, Parole, Signal et Automatique (GIPSA-Lab), CNRS, et Université de Grenoble, 961 rue de la Houille Blanche, 38402 Saint Martin d'Hères, France
[2] Instituto de Física de La Plata (IFLP), CONICET, and Departamento de Física, Facultad de Ciencias Exactas, Universidad Nacional de La Plata, C.C. 67, 1900 La Plata, Argentina
[3] Facultad de Matemática, Astronomía y Física (FaMAF), CONICET, and Universidad Nacional de Córdoba, Avenida Medina Allende, Ciudad Universitaria, X5000HUA, Córdoba, Argentina

Received 18 November 2013 / Received in final form 13 March 2014
Published online (Inserted Later) –

Abstract. In this paper, we propose to mix the approach underlying Bandt-Pompe permutation entropy with Lempel-Ziv complexity, to design what we call Lempel-Ziv permutation complexity. The principle consists of two steps: (i) transformation of a continuous-state series that is intrinsically multivariate or arises from embedding into a sequence of permutation vectors, where the components are the positions of the components of the initial vector when re-arranged; (ii) performing the Lempel-Ziv complexity for this series of 'symbols', as part of a discrete finite-size alphabet. On the one hand, the permutation entropy of Bandt-Pompe aims at the study of the entropy of such a sequence; i.e., the entropy of patterns in a sequence (e.g., local increases or decreases). On the other hand, the Lempel-Ziv complexity of a discrete-state sequence aims at the study of the temporal organization of the symbols (i.e., the rate of compressibility of the sequence). Thus, the Lempel-Ziv permutation complexity aims to take advantage of both of these methods. The potential from such a combined approach – of a permutation procedure and a complexity analysis – is evaluated through the illustration of some simulated data and some real data. In both cases, we compare the individual approaches and the combined approach.

1 Introduction

Many real signals result from very complex dynamics and/or from coupled dynamics of many dimensional systems. Various examples can be found in biology, such as the reaction-diffusion process in cardiac electrical propagation that provides electrocardiograms, and the collective actions of genes for the production of proteins in specific quantities [1–3]. In finance, there is the example of the variation in the price of an asset, which results from the collective actions of the buyers and sellers [4], while statistical physics and social sciences also have huge numbers of situations where 'complexity' emerges [5]. One of the challenges is to describe these complex signals in a simple way, to allow meaningful and relevant information to be extracted [6–10].

The complex origin of such signals has led researchers to analyze these signals through tools that come either from the 'probability world', or conversely, from 'nonlinear dynamics'. The purpose is to characterize the degree of information or the complexity of the signals under analysis as well as possible. The first approach is statistical, and the goal is to measure the spread of the distribution underlying the data, or to detect any changes in the statistics. The common tools that are used here come from information theory [8,9,11–13], or are correlation measures [3], or come from spectral analysis [14]. The second approach is devoted to signals that are produced by deterministic (generally nonlinear) mechanisms, even if the sequence under analysis can appear to be somewhat 'random'. The tools generally used for the description of such complex signals come often from the chaos world, like Lyapunov exponents, fractal dimensions, and others [6], or from the concept of complexity in the sense of Kolmogorov (e.g., Lempel-Ziv complexity) [7,10,15–17].

The measures from information theory are very powerful, in a sense that they allow the quantification of a degree of uncertainty (the rate) of a random sequence, or of a sequence considered as randomly generated. However, tools such as entropies can have some drawbacks when used in practice. One of these occurs when dealing with continuous-state data. In this case, the estimation of a differential entropy from the data is not always an easy task [18–20]. Some nonparametric estimators make use of nearest neighbors, or of graph lengths, although their properties are difficult to study [18–22]. More simple

[a] e-mail: steeve.zozor@gipsa-lab.grenoble-inp.fr

estimators are based on 'plug-in' approaches [18]; namely, the density is estimated using a Parzen-Rosenblatt approach [23,24], and the estimation is plugged into the mathematical expression of the entropy. The most simple density estimator is based on a histogram, which is equivalent to quantization of the data. The estimation performance depends on this quantization (e.g., number of thresholds, quantization intervals). To overcome this potential difficulty, Bandt and Pompe proposed (i) to construct the multivariate trajectories from the scalar series, i.e., an embedding; and (ii) to work with the so-called vectors of permutation, i.e., for each point of the trajectory, its components are sorted, and each component of the point is replaced by its position (rank) in the rearranged components [25]. Bandt and Pompe proposed then to estimate the discrete entropy of the permutation vector sequence, which led to the so-called *permutation entropy*, and later on, to some variations of this measure [26–28]. However, when dealing with sequences generated by a deterministic process, such statistical measures can be inappropriate because they measure an ensemble, or average, behavior.

Conversely, for deterministic sequences generated by dynamical systems, there are a huge number of analysis tools, like Lyapunov exponents, and fractal dimensions, among others [29–31]. In general, the quantities under study are relatively difficult to evaluate, and they require long times of computation. As an example, there can be the need to reconstruct a phase-space trajectory using several estimations to determine the embedding dimension and the optimal delay, and then, in a second step, to estimate some quantities from the reconstructed trajectory, such as the whole Lyapunov spectrum, or just some exponents (e.g., positive, max), or dimensions [31,32]. Moreover, these tools are generally designed specifically for the study of chaotic series. A more natural concept of 'uncertainty' of a time series, whether chaotic or not, is that of its complexity in the sense of Kolmogorov. Roughly speaking, this measures the *minimal size of a binary program that can generate the sequence* (i.e., the algorithmic complexity) [33,34]. Among these, there is the Lempel-Ziv complexity, which is based on simple recursive copy-paste operations, as will be seen later [35,36]. This kind of measure naturally finds applications in the compression domain [33,36,37], and it is also used for signal analysis [10,13,15,16]. A strength of this complexity is that as it deals with a random discrete-state and ergodic sequence, and when it is correctly normalized, it converges to the entropy rate of the sequence [33,35]. In a sense, the Lempel-Ziv complexity contains the concept of complexity both in the deterministic sense (Kolmogorov) and in the statistical sense (Shannon). This property led to the use of the Lempel-Ziv complexity for entropy estimation purposes [21,38]. A possible drawback of the Lempel-Ziv complexity is that it is defined for sequences that take their values on a discrete (finite sized) alphabet. If it can find natural applications that deal with discrete-state sequences, such as DNA sequences or sequences generated by logical circuits, while 'real-life' signals are generally continuous states[1]. Thus, to use the Lempel-Ziv complexity for signal characterization purposes, there is first the need to quantize the data, which introduces some parameters into the tuning. These parameters can influence the behavior of the complexity of the quantized signal, as can be seen, e.g., in reference [39], where for a logistic map, some bifurcations are not (completely) captured by the Lempel-Ziv complexity.

As can be imagined, there are many ways to overcome the drawbacks of purely statistical methods or purely deterministic approaches. Here, we concentrate on the Lempel-Ziv complexity, using first the idea that underlies the Bandt-Pompe entropy, to 'quantize' a sequence to analyze.

This report is organized as follows. In Section 2, we first define the notation we use in the following sections. Then we provide some basics on Bandt-Pompe entropy (or permutation entropy). In the same section, we also provide some basics on Lempel-Ziv complexity, proposing then to 'mix' both of these approaches in Section 3, to give what we call the *Lempel-Ziv permutation complexity*. In this same section, we provide some properties of the Lempel-Ziv permutation complexity, including in an Appendix the technical details and the description of a practical way to calculate this complexity when dealing with scalar sequences. We then illustrate in Section 4 how the Lempel-Ziv permutation complexity can be used for data analysis of both simulated sequences and biological signals, and we finish the paper by drawing up our concluding remarks.

2 Notation and recall

2.1 Bandt-Pompe permutation entropy

The starting point of the Bandt-Pompe approach [25] appears to take its origin from a study of chaos, and more specifically, through the famous Takens' delay embedding theorem [31,32]. The principle of this theorem is the reconstruction of the state trajectory of a dynamical system from the observation of one of its states. To fix the ideas, consider a real-valued discrete-time series $\{X_t\}_{t\geq 0}$ that is assumed to be a state of a multidimensional trajectory. Consider two integers $d \geq 2$ and $\tau \geq 1$, and from the series, let us then define a trajectory in the d-dimensional space as:

$$\boldsymbol{Y}_t^{(d,\tau)} = \left[X_{t-(d-1)\tau} \quad \dots \quad X_{t-\tau} X_t\right]^t, \quad t \geq (d-1)\tau \tag{1}$$

where the dimension d is known as the *embedding dimension*, and where τ is called the *delay*. Takens' theorem

[1] When performing the acquisition of a signal in a computer, for example, the (discrete time) series is intrinsically a discrete-state series due to the finite precision of the computer. However, this precision is generally high, so that the series can be assumed to be a continuous-state series. In particular, in general, the number of possible states is much higher that the number of samples to be analyzed.

gives conditions on d and τ such that $\boldsymbol{Y}_t^{(d,\tau)}$ preserves the dynamical properties of the full dynamic system (e.g., reconstruction of strange attractors) [31,32]. Many studies have dealt with 'optimal' reconstruction of this phase space; i.e., the choice of the correct embedding dimension, and more particularly, the 'optimal' delay.

In reference [25], Bandt and Pompe did not focus especially on chaotic signals, even if these signals serve as illustrations. Thus, they did not focus on the phase-space reconstruction problem. More precisely, they did not provide discussion on the parameters d and τ. The only ingredient they wished to conserve was the idea of taking into account the dynamics of the system underlying an observed signal. These questions of optimal reconstruction also go beyond the scope of our paper, so we do not discuss the choice of the embedding dimension and of the delay in the sequel anymore.

Starting with the phase-space trajectory $\boldsymbol{Y}_t^{(d,\tau)}$, instead of focusing on the real-valued vectors, Bandt and Pompe were interested in the order of the components of the vectors. The principle consists first of the sorting (in ascending order) of the components of $\boldsymbol{Y}_t^{(d,\tau)}$, and then the replacement of each component $X_{t-k\tau}$ by its rank/ position in the sorted vector. This so-called *permutation vector* is denoted as $\boldsymbol{\Pi}_{\boldsymbol{Y}_t^{(d,\tau)}}$ in the following. As an example, for a vector $\boldsymbol{Y} = [Y_0 \quad Y_1 \quad Y_2]^t$ such that $Y_2 \le Y_0 \le Y_1$, the permutation vector is $\boldsymbol{\Pi}_{\boldsymbol{Y}} = [1 \quad 2 \quad 0]^t$. Dealing with random processes, it is then possible to define the *permutation entropy* as the Shannon entropy H of the (random) permutation vector

$$H_{d,\tau}^{\pi}(X_t) \equiv H\left(\Pi_{\boldsymbol{Y}_t^{(d,\tau)}}\right). \tag{2}$$

For a stationary process, provided the size of the sequence is large enough in terms of $d!$, the entropy can be estimated via the frequencies of occurrence of any of the $d!$ possible permutation vectors in the sequence $\boldsymbol{Y}_t^{d,\tau}$. In their paper, Bandt and Pompe defined the permutation entropy as the Shannon entropy of the frequencies of the permutation vectors[2], which gives asymptotically the entropy $H_{d,\tau}^{\pi}(X_t)$ of equation (2) when dealing with a long-time (infinite) stationary and ergodic process, as indicated in reference [25]. Starting from a sequence of length T, $X_0 \dots X_{T-1}$, in the sequel we write $\widehat{H}_{d,\tau}^{\pi}(X_{0:T-1})$ for the entropy of the frequencies, to distinguish this from the entropy of the random process. Several quantifiers of information based on $\widehat{H}_{d,\tau}^{\pi}(X_{0:T-1})$ were proposed in reference [25], although such extensions go beyond the purpose of the present paper. Thus, we do not present these here.

The idea behind permutation entropy is that the $d!$ possible permutation vectors, also called *patterns*, might not have the same probability of occurrence, and thus, this probability might unveil knowledge about the underlying system. For a sequence of independent and identically distributed (iid) variables, whatever the distribution of the random variable, all of the patterns have the same probability $\frac{1}{d!}$ of occuring (whatever the delay τ), so that the permutation entropy is maximum and equal to $\log(d!)$ [25]. Conversely, an important situation is represented by the so-called *forbidden patterns*, which are patterns that do not appear at all in the analyzed time series [40–42]. As an example, it was shown in the logistic map $X_{t+1} = 4X_t(1-X_t)$ that whatever the initialization X_0, for $d=3$ and $\tau=1$, the permutation vector $[2 \quad 1 \quad 0]^t$ never appears. Such behavior shows how the use of permutation vectors allows the distinguishing between purely random sequences and deterministic sequences (e.g., when the last one is chaotic, and thus appears random): some authors have said that the presence of forbidden patterns is an indicator of deterministic dynamics [40–42]. This question remains, however, controversial, as it is possible to construct random series with forbidden patterns [43], and conversely, a chaotic series does not always show forbidden patterns [44].

Note that if we work on a multidimensional sequence $\{\boldsymbol{X}_t\}_{t\ge0}$, the permutation procedure can be performed on each vector $\boldsymbol{X}_t$, so that there are no embedding procedures. To distinguish this situation from that of Bandt and Pompe, we denote the permutation entropy and its estimate as H^{π} and $\widehat{H}^{\pi}$, respectively, without mention of any delay and embedding dimension.

2.2 Lempel-Ziv complexity

Consider a finite-size sequence $S_{0:T-1} = S_0 \dots S_{T-1}$ of symbols that take their values in an alphabet $\mathcal{A}$ of finite size $\alpha = |\mathcal{A}|$. In 1965, Kolmogorov introduced the concept of the complexity of such a sequence as the size of the smallest binary program that can produce the sequence [33]. In an algorithmic sense, the Kolmogorov complexity measures the minimal 'information' contained in the sequence, or the minimal information needed to generate the sequence. Several years later, the seminal work of Lempel and Ziv appeared [35], which dealt with the complexity of the Kolmogorov type of a sequence, restricting this concept to the 'programs' based only on two operations: recursive copy and paste operations. Their definition lies in the two fundamental concepts of reproduction and production:

- **Reproduction:** this consists of extending a sequence $S_{0:T-1}$ of length T, adding a sequence $Q_{0:N-1}$ via recursive copy-paste operations, which leads to $S_{0:T+N-1}$, i.e., the first letter Q_0 is in $S_{0:T-1}$, let us say $Q_0 = S_i$, the second one is the following one in the extended sequence of size $T+1$, i.e., $Q_1 = S_{i+1}$, etc.: $Q_{0:N-1}$ is a subsequence of $S_{0:T+N-2}$. In a sense, all of the 'information' of the extended sequence $S_{0:T+N-1}$ is in $S_{0:T-1}$.
- **Production:** the extended sequence $S_{0:T+N-1}$ is now such that $S_{0:T+N-2}$ can be reproduced by $S_{0:T-1}$.

[2] More precisely, in their paper, the permutation vector is defined as the time position of the component in the sorted vector, instead of the vector of the rank of the vector components. As there is a one-to-one mapping between the two ways of making, the entropy of the two vectors is the same.

The last symbol of the extension can also follow the recursive copy-paste operation, so that the production is a reproduction, but can be 'new'. Note thus that a reproduction is a production, but the converse is false.

Any sequence can be viewed as constructed through a succession of productions, called a history. As an example, a sequence can be 'produced' symbol by symbol. However, a given sequence does not have a unique history; several processes of productions can lead to the same sequence. In the spirit of the Kolmogorov complexity, Lempel and Ziv were interested in the optimal history; i.e., the minimal productions needed to generate a sequence: the so-called Lempel-Ziv complexity, denoted as $C(S_{0:T-1})$ in the following, is this minimal number of production steps needed for the generation of $S_{0:T-1}$. In a sense, C describes the 'minimal' information needed to generate the sequence by recursive copy-paste operations. Thus, the approach of Lempel and Ziv, and of several variations [36,37], naturally gave rise to various algorithms of compression (including the famous 'gzip'). It can intuitively be understood that in a minimal sequence of production, all of the productions are not reproductions, otherwise it would be possible to reduce the number of steps [35]. This allowed the development of simple algorithms for the evaluation of the Lempel-Ziv complexity of a sequence [39].

Surprisingly, although analyzing a sequence from a completely deterministic point of view, it appears that $C(S_{0:T-1})$ sometimes also contains the concept of information in a statistical sense. Indeed, it was shown in references [33,35] that for a random stationary and ergodic process, when correctly normalized, the Lempel-Ziv complexity of the sequence tends to the entropy rate of the process; i.e.,

$$\lim_{T\to+\infty} C\left(S_{0:T-1}\right)\frac{\log(T)}{T} = \lim_{T\to+\infty}\frac{H\left(S_{0:T-1}\right)}{T} \tag{3}$$

where $H(S_{0:T-1})$ is the joint entropy of the T symbols, and the righthand side is the entropy rate (entropy per symbol) of the process. Such a property gave rise to the use of the Lempel-Ziv complexity for entropy estimation purposes [21,38].

Note that using the Lempel-Ziv complexity for analysis purposes might not be envisaged if the size of the sequence is not large enough in terms of the size of the alphabet. Indeed, for small sequences compared to the size of the alphabet, except for very elementary situations (e.g., constant signals, periodic signals), the complexity of the sequence has a great probability of being close to the size of the sequence.

3 The Lempel-Ziv permutation complexity

As we have just seen, in a sense, the Lempel-Ziv complexity aims to capture a level of redundancy, or of regularity, in a sequence. Thus, this tool is interesting for the analysis of signals that appear to be random, but that hide some regularities, such as in chaotic sequences [39]. Conversely, viewing this complexity as an estimator of the Shannon entropy when dealing with random sequences, its use is also relevant in such a context. In some sense, it provides a bridge between the two above-mentioned contexts. However, a disadvantage of the Lempel-Ziv complexity is that it is defined only for sequences of symbols taken in a discrete (finite size) alphabet. Dealing with 'real-life' sequences, a quantization has to be performed before its use, as has been done in many of the studies dealing with data analysis via this complexity [10,15,16]. Quantizing a signal might have some consequences in the evaluation of the complexity, and the effects of the parameters of the quantizers appear difficult to evaluate.

Conversely, the permutation entropy also has some drawbacks due to its statistical aspects. To illustrate why sometimes it cannot capture the dynamics of a sequence, consider the example of an iid scalar noise, versus a periodic scalar sequence of period $T = 2$. For an embedding dimension $d = 2$ and a delay $\tau = 1$, in both cases the permutation vectors $[0 \quad 1]^t$ and $[1 \quad 0]^t$ appear with the same frequency $\frac{1}{2}$ (assuming the length of the sequence is large enough). Thus, the permutation entropy is equal in both cases, and in this example it is thus not sensitive enough to discriminate between the random iid sequence and the periodic sequence[3]. Several variants to avoid such a drawback can be imagined; e.g., taking into account the amplitudes when constructing the permutation vectors. The weighted-permutation entropy proposed in reference [28] shows its efficiency for the detection of abrupt changes in a sequence, but in the example given above, it will not be able to discriminate between the two situations. Moreover, when dealing with an intrinsic multidimensional sequence, the permutation vectors do not clearly reflect any dynamics.

To avoid the possible disadvantages of both methods, we propose here to mix the Bandt-Pompe and Lempel-Ziv approaches; i.e., to analyze the sequence of permutation vectors via the Lempel-Ziv complexity. In this way, it is expected that we can take advantage of both methods, and thus reduce their respective drawbacks. In the following, the so called *Lempel-Ziv permutation complexity* of a finite length scalar sequence $X_{0:T-1}$ or a finite length multivariate sequence $\boldsymbol{X}_{0:T-1}$ are respectively denoted as:

$$C^{\pi}_{d,\tau}\left(X_{0:T-1}\right) \equiv C\left(\boldsymbol{\Pi}_{\boldsymbol{Y}^{(d,\tau)}_{(d-1)\tau}} \ldots \boldsymbol{\Pi}_{\boldsymbol{Y}^{(d,\tau)}_{T-1}}\right) \tag{4}$$

where $\boldsymbol{Y}^{(d,\tau)}_t = [X_{t-(d-1)\tau} \quad \ldots \quad X_{t-\tau} \quad X_t]^t$ and $\boldsymbol{\Pi}_{\boldsymbol{Y}^{(d,\tau)}_{T-1}}$ is its permutation vector, and

$$C^{\pi}\left(\boldsymbol{X}_{0:T-1}\right) \equiv C\left(\boldsymbol{\Pi}_{\boldsymbol{X}_0} \ldots \boldsymbol{\Pi}_{\boldsymbol{X}_{T-1}}\right). \tag{5}$$

This way provides an answer to the necessity of working with data taking the values on a finite size alphabet (here,

[3] More rigorously, it is known that using the permutation entropy for data analysis, several embedding dimensions have to be tested. For $d = 3$ in this example, the permutation entropy makes the distinction between the iid noise and the periodic sequence.

the alphabet is $\mathcal{A} \equiv \{[\pi(0) \quad \dots \quad \pi(d-1)]^t : \pi \in \Pi^{(d)}\}$ of size $\alpha = d!$, where $\Pi^{(d)}$ is the ensemble of the $d!$ possible permutations on $\{0, \dots, d-1\}$). Moreover, viewing a permutation vector as quantization of the data, it is interesting to draw a parallel with dynamical quantization; namely, of the sigma-delta type [45]. Indeed, dealing with scalar real-state sequences, in the case where $\tau = 1$ and $d = 2$, for instance, the permutation vector is $[0 \quad 1]^t$ if the signal increases locally, and is $[1 \quad 0]^t$ otherwise. In other words, the two possible permutation vectors quantize the variations of the signal in one bit. Roughly speaking, a sigma-delta quantizer acts in a similar way[4]. For $d > 2$, the same parallel should be made in some sense with the so-called multi-stage sigma-delta quantizers [46]. This parallel is another motivation to use permutation vectors as a way to quantize a signal. Moreover, dealing with intrinsically multivariate sequences, the permutation vectors can be viewed as (vector) quantization of the real-valued vectors; this scheme does not need tuning parameters, contrary to standard vector quantization schemes [45].

Working on the permutation vectors maintains the idea of studying the occurrences of patterns in a sequence. By analyzing the permutation vectors via the Lempel-Ziv complexity, a step is added because how the patterns are temporarily organized is analyzed, rather than the frequency of occurrences. To stress this, let us come back to the example of the permutation vector sequences of an iid noise versus a periodic sequence of period $T = 2$. As previously explained, the patterns $[0 \quad 1]^t$ and $[1 \quad 0]^t$ appear with the same frequency in both cases. However, the difference between the permutation vector sequences in the two cases is that in the first case, the two patterns appear in a random sequence, while in the second case, they appear periodically: in the first case, the Lempel-Ziv complexity is then high, while it is low (and equal to 3) in the second case. With this very elementary example, it can be seen why the Lempel-Ziv permutation complexity of a sequence can provide more information on the dynamics; i.e., by analyzing how the patterns are organized temporarily, not only in terms of the frequency of occurrence.

Moreover, dealing with intrinsically multivariate sequences, the argument of capturing the dynamics that underlie the sequence fails, as there is no embedding prior to the quantization that is made with the construction of the permutation vector. At least this question is not clear. In essence, the Lempel-Ziv complexity will in a way capture the dynamics of such a multivariate sequence, which strengthens the interest for mixing both the Bandt-Pompe and Lempel-Ziv approaches in this context.

The Lempel-Ziv permutation complexity has some properties that have been inherited from the standard Lempel-Ziv complexity. The first is the link with the permutation entropy. Indeed, for a stationary ergodic process that is scalar or multivariate, the sequence of permutations remains stationary and ergodic, so that equation (3) applies to the Lempel-Ziv complexity and the entropy rate of this sequence, which can be written as:

$$\lim_{T\to\infty} C^{\pi}_{d,\tau}\left(X_{0:T-1}\right) \frac{\log T}{T} = \lim_{T\to\infty} \frac{H^{\pi}_{d,\tau}\left(X_{0:T-1}\right)}{T} \qquad (6)$$

and

$$\lim_{T\to\infty} C^{\pi}\left(\boldsymbol{X}_{0:T-1}\right) \frac{\log T}{T} = \lim_{T\to\infty} \frac{H^{\pi}\left(\boldsymbol{X}_{0:T-1}\right)}{T}. \qquad (7)$$

The second property is the invariance of the Lempel-Ziv permutation complexity to a given permutation applied to the components of the vector of the initial series; i.e., for any permutation matrix $\boldsymbol{P}$,

$$C^{\pi}\left(\boldsymbol{P}\boldsymbol{X}_0 \dots \boldsymbol{P}\boldsymbol{X}_{T-1}\right) = C^{\pi}\left(\boldsymbol{X}_0 \dots \boldsymbol{X}_{T-1}\right). \qquad (8)$$

In other words, if a sequence of vectors $\boldsymbol{X}_t$ is constructed from d scalar sequences, the choice of the order of the components does not modify the value of the complexity of the 'joint' sequence. This property arises because $\boldsymbol{\Pi}_{\boldsymbol{P}\boldsymbol{X}_t} = \boldsymbol{P}\boldsymbol{\Pi}_{\boldsymbol{X}_t}$ (permuting the components of a vector results in permuting the components of its permutation vector), together with the invariance of the Lempel-Ziv complexity to a one-to-one transformation [17].

As shown by reference [17] for the Lempel-Ziv complexity, it is possible to build measures associated with the Lempel-Ziv permutation complexity, although such possible extensions go beyond the scope of the present paper.

Before moving on to put the Lempel-Ziv permutation complexity into action, let us just note the following additional choices:

- To take into account a finite resolution in data acquisition or to counteract possible low noise in the data, we can introduce a radius of confidence δ; i.e., if the absolute value of the difference of two components is strictly lower than δ, then they are considered to be equal.
- Performing the permutation procedure, when two components of a vector are equal, we chose the 'smallest' one as that with the lowest index (the oldest one in the case of embedding).

(see Appendix for more details and further justification).

Once again, note that using the Lempel-Ziv permutation complexity for analysis purposes might not be feasible if the size of the sequence is not large enough in terms of the size of the alphabet $d!$.

4 Illustrations based on synthetic and real data

4.1 Characterizing the logistic map

To illustrate how the Lempel-Ziv permutation complexity can capture regularities in a signal, we consider here the example of the famous logistic map

$$X_{t+1} = k\, X_t\, (1 - X_t), \qquad t \geq 0, \qquad k \in (0; 4]. \qquad (9)$$

[4] More rigorously, it quantizes the difference between a sample and a prediction of this sample (the 'delta' part) in one bit. The prediction is made from all of the past samples, in general performing an integration or a summation (the 'sigma' part).

We initialize X_0 randomly in $[0;1]$ so that the sequence has a real value in the interval $[0;1]$. This map has already been taken as an illustration by both Bandt and Pompe in reference [25], and Kaspar and Schuster in reference [39].

The logistic map has been studied for a long time, and its behavior is well known and can be found in any textbook on chaos; e.g., [47,48]. Let us just recall that when k increases, it shows more and more complex regimes: there is an increasing sequence of values $k_{-1} = 0 < k_0 < \ldots < k_\infty \approx 3.56995$ such that, if $k \in (k_{n-1}; k_n]$, the output asymptotically oscillates between 2^n values, a phenomenon that is well known as *bifurcations*. For $k \geq k_\infty$, the system is in a chaotic (unpredictable) regime. Roughly speaking, it appears to behave randomly, although it is produced by an elementary deterministic system. However, in this zone, there remain some intervals, known as *islands of stability*, in which the behavior is nonchaotic. This briefly described behavior is summarized in the bifurcation diagram plotted in Figure 1A.

Let us now study the regimes of the logistic map versus k through the Lempel-Ziv permutation complexity proposed here. To this end, a sequence of size $T = 1000$ is drawn and only the second half of the sequence, which is assumed to be in the permanent regime, is analyzed. The behavior of $C^\pi_{(d,\tau)}$ versus k is depicted in Figure 1G, and this is compared to the permutation entropy (Figs. 1D−1F), to the Lempel-Ziv complexity performed on a static 2-level quantization of the signal $\mathbb{1}_{(.5;1]}(X_t)$, where $\mathbb{1}$ is the indicator function (Fig. 1C), and to the Lyapunov exponents (Fig. 1B). Roughly speaking, the Lyapunov exponent[5] measures the exponential convergence or divergence of two trajectories for two close initial conditions: a positive Lyapunov exponent is a signature of chaos [47,48].

The behavior of each descriptor can be interpreted as follows:

- *The Lyapunov exponent*: this exponent clearly describes the chaotic character of the logistic sequence (when it is positive) versus its non-chaotic character (when it is negative). However, as already mentioned in the literature, this is not precise enough to distinguish different types of behavior in nonchaotic regimes.
- *The Lempel-Ziv complexity* $c(\{\mathbb{1}_{(.5;1]}(X_t)\})$: as claimed by Kaspar and Schuster, this measure is more precise than the Lyapunov exponent. In particular, the complexity is very high in chaotic regimes, while it is low in nonchaotic regimes. However, it can be seen that the bifurcations are not detected very well. This is clearly due to the quantization threshold. Indeed, for $k < 3.237$, the system asymptotically oscillates between two values > 0.5, the threshold that was chosen by Kaspar and Schuster, which explains why the complexity fails to detect the bifurcations. The same phenomenon appears for the following bifurcations. Note that choosing a threshold of $2/3$ for this system leads to the detection of the first bifurcation, but the other bifurcations remain undetected.
- *The permutation entropies* $\widehat{H}^\pi_{(d,1)}$: in both cases of $d = 3$ and $d = 5$, the permutation entropy precisely characterizes the different regimes of the logistic map. In particular, it is high in chaotic regions, while it is low in nonchaotic regions; e.g., as is the case in the islands

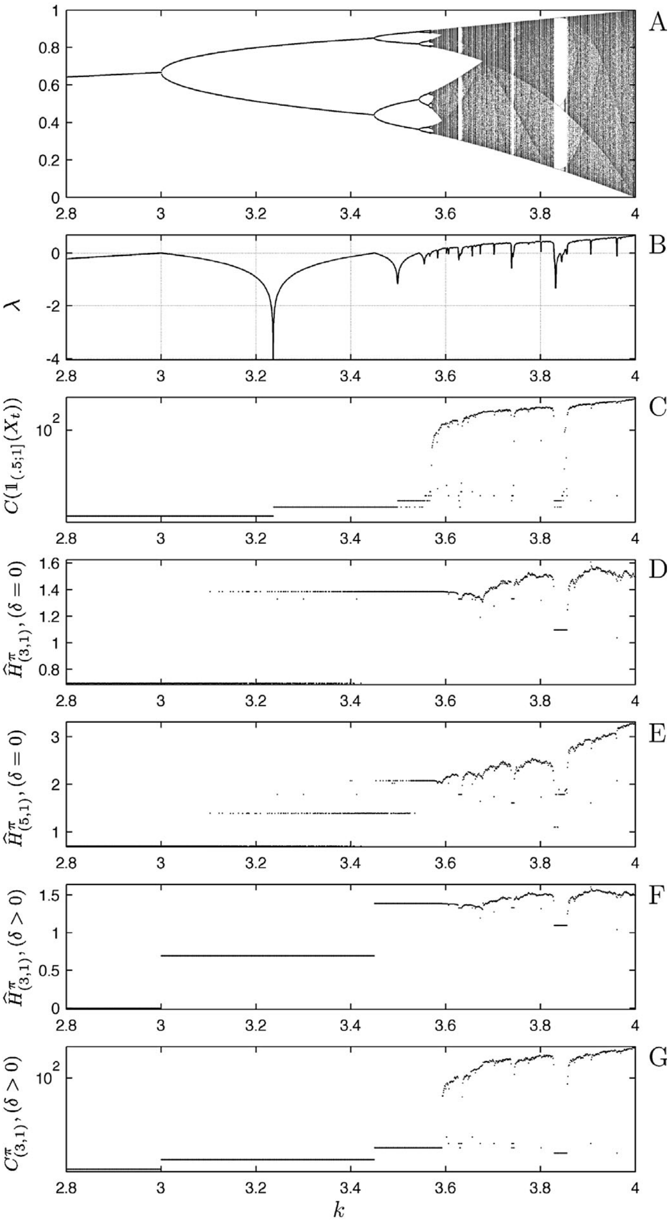


Fig. 1. Characterization of the logistic map versus k. (A) The bifurcation diagram; i.e., the values taken by the series in the permanent regime for each value of k. (B) The Lyapunov exponent λ. (C) The Lempel-Ziv complexity of the quantized signal $\mathbb{1}_{(.5;1]}(X_t)$ as in reference [39]. (D–F) The permutation entropy $\widehat{H}^\pi_{(d,\tau)}$ with a delay $\tau = 1$ when $(d,\delta) = (3,0)$ (D), $(d,\delta) = (5,0)$ (E), and $(d,\delta) = (3,10^{-3})$ (F). (G) The Lempel-Ziv permutation complexity $C^\pi_{(d,\tau)}$ for $(d,\tau,\delta) = (3,1,10^{-3})$.

[5] For a discrete map of the type $X_{t+1} = f(X_t)$, this coefficient is given by $\lambda = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \log f'(X_t)$ [47,48]. Practically speaking, this is calculated for a large T.

of stability. This is particularly true for the 'high' embedding dimension; e.g., $d = 5$. Note that for $\delta = 0$, the first bifurcation is not detected here. This is due to the small oscillations that remain around the limit value when $k \in (2; k_0]$. The consequence is that the permutation entropy fails to detect the first bifurcation, as the damped oscillatory behavior of the system for $k \in (2; k_0]$ is seen in the same manner as the sustained oscillations of the system when $k \in (k_0; k_1]$. Obviously, the permutation entropy ($\delta = 0$) detects this oscillatory behavior which is inherent to the system. However, if we are not really interested in the signal itself, but in its asymptotic regime, the small fluctuations can be viewed as perturbations. Choosing $\delta > 0$ allows the 'filtering' of these perturbations. In this case, even if the permutation entropy does not characterize the logistic sequence itself, it very precisely characterizes the asymptotic regimes of the sequence, as can be seen in Figure 1. Indeed, in this case, the bifurcations are very well detected, even for 'low' embedding dimensions.

– *The Lempel-Ziv permutation complexity* $C^{\pi}_{(d,1)}$: at a first glance, this measure behaves like the permutation entropy. In particular, the same effects of detection or not of the bifurcation occur if $\delta = 0$ (not plotted in Fig. 1) or $\delta > 0$. Note, however, that even in the low embedding dimension, the complexity appears to better characterize the constant, oscillatory or chaotic regimes. Indeed while $\widehat{H}^{\pi}_{(3,1)}$ is roughly constant when the chaos appears (for k slightly $> k_\infty$), the complexity greatly increases.

4.2 Detecting of a sudden change in a three-dimensional signal

To illustrate how the proposed measure can outperform the permutation entropy in assessing the degree of complexity of some signals, let us consider a multidimensional series $\boldsymbol{X}_t$ composed first of N_c points issued from a d-dimensional logistic series, followed by N_n points of both spatially and temporally iid noise. The d-dimensional logistic map we have chosen here for our purpose is described by the following equation:

$$\boldsymbol{X}_{t+1} = k \left(\boldsymbol{K}\, \boldsymbol{X}_t + \boldsymbol{1} \right) \odot \boldsymbol{X}_t \odot \left(\boldsymbol{1} - \boldsymbol{X}_t \right) \qquad (10)$$

where $\boldsymbol{X}_t$ is a d-dimensional vector, $\boldsymbol{1} = [1 \quad \ldots \quad 1]^t$, $\boldsymbol{K}$ is a $d \times d$ coupling matrix, and $\odot$ is the component-wise product ($t \geq 0$). When $\boldsymbol{K}$ is zero, the d logistics are decoupled. For the opposite, when $\boldsymbol{K} = 3\boldsymbol{P}$ with $\boldsymbol{P}$ as the cyclic permutation matrix of one place to the left, or when $\boldsymbol{K} = \boldsymbol{1}\boldsymbol{1}^t$, the map corresponds to the models proposed by Lopez-Ruiz and Fournier-Prunaret in the 2-dimensional and 3-dimensional contexts to model symbiotic interactions between species, where parameter k represented the growth rate of the species [49,50]. In both the cases of $d = 2$ and $d = 3$, according to the value of k, these maps show regular orbits or chaotic orbits. We do not describe here the richness of these maps, but instead direct the reader to [49,50].

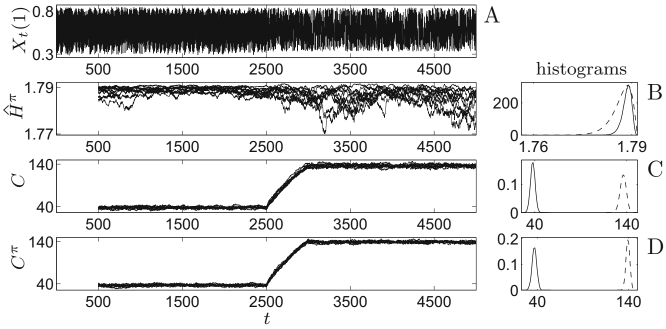


Fig. 2. Detection of a sudden change in a 3-dimensional sequence composed of $N_c = 2500$ points of a coupled 3-dimensional logistic map given by equation (10) followed by $N_n = 2500$ points of pure random noise (uniform). (A) A snapshot of the first component of such a sequence, with 2000 sequences then analyzed through a sliding window of $N_w = 500$ points, moving sample by sample. (B–D) Ten snapshots of the permutation entropies (B), the Lempel-Ziv complexities of a quantized version of the vectors (C), and Lempel-Ziv permutation complexity (D) are shown. Right: the corresponding histograms of the values taken by the measure, showing the windows in the chaotic part (solid line) and in the noise part (dashed line). The chaotic map is here strongly coupled, with $\boldsymbol{K} = 3\boldsymbol{P}$ and $k = 1.01$.

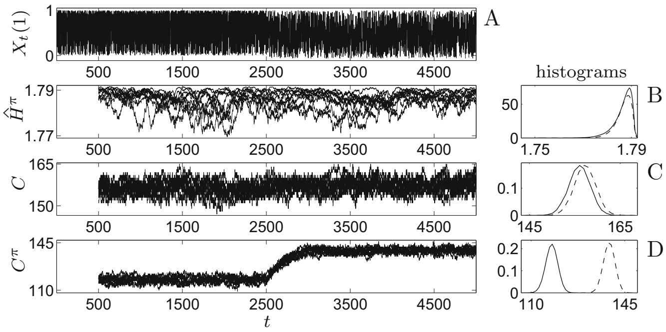


Fig. 3. Same as for Figure 2 for a weakly coupled chaotic map, with $\boldsymbol{K} = .01\boldsymbol{P}$ and $k = 3.96$.

For our purposes, we have chosen to study what happens when the N_c first points of the sequence are generated by the 3-dimensional map ($d = 3$) showing chaotic behavior. We considered two cases: in the first, the coupling is $\boldsymbol{K} = 3\boldsymbol{P}$ and $k = 1.01$; and in the second, $\boldsymbol{K} = .01\boldsymbol{P}$ and $k = 3.96$. In the first case, the components are strongly coupled, while they are weakly coupled in the second case. A snapshot of these logistic map sequences followed by pure noise is shown in Figures 2A and 3A. Visually, it is relatively difficult to detect the instant where the nature of the signal changes. Let us then analyze the signal through sliding windows of size N_w, moving point by point. In each window of the analysis $(\boldsymbol{X}_{t-N_w+1}, \ldots, \boldsymbol{X}_t)$, $t = N_w - 1, \ldots,$ we evaluate the permutation entropy, the Lempel-Ziv complexity of a quantized version of the components (by $\mathbb{1}_{[.5;+\infty)}$), and the Lempel-Ziv permutation complexity. The results versus t are plotted in Figures 2B–2D and 3B–3D, where 10 realizations are shown. On the right of Figures 2B–2D and 3B–3D, the corresponding histograms are shown[6] for the values taken by each measure using 4×10^6 snapshots of the chaotic map (solid lines) and the noise (dashed lines).

In these examples, the interpretations are the following:

- *The permutation entropy*: this index cannot detect the change in the nature of the signal, as can be seen in the snapshots for both the strong and weak coupling (Figs. 2B and 3B). This is because, in these examples, the patterns obtained in the permutation vectors performed on the components appear with similar frequencies to the chaotic regime and in the noise regime. By statistically analyzing these patterns, the dynamics underlying the data are lost. The difficulty in the discrimination between chaos and noise is also illustrated by the probabilities taken by the values of H^{π}: roughly speaking, the probability of error in a discrimination task is a function of the surface shared by the two distributions.
- *The Lempel-Ziv complexity*: when looking at the case of the strong coupling between the components of the logistic, the Lempel-Ziv complexity performed on the basic quantized version of the vector clearly discriminates between chaos and noise. However, when the components are weakly coupled, this is no more the case. This is clearly seen in the histograms that overlap in the weak coupling case (Fig. 3C) while they are separated in the strong coupling situation (Fig. 2C). Our interpretation of this effect is that, in a sense, the Lempel-Ziv analyzes the components almost individually: in the weak coupling case, it does not 'see' that the components follow exactly the same dynamics and are, in a sense, linked by these common dynamics.
- *The Lempel-Ziv permutation complexity*: in both types of coupling, this measure unambiguously detects the change in the nature. This can be viewed both in the snapshots and in the probability distributions of the values taken by this measure (Figs. 2D and 3D). Clearly, there is no overlap between the two histograms, which confirms that there is no probability of error in the discrimination between the chaos and noise in this illustration. From the curves, it would appear that for both cases, the Lempel-Ziv permutation entropy shows a weaker dispersion around its mean value than does the standard Lempel-Ziv complexity.

These illustrations show that in spite of the power of the permutation entropy to discriminate between chaos and randomness, for instance, there are situations in which this tool fails in this task. Using the Lempel-Ziv complexity of a basic quantized version of the sequence can be an alternative, but this remains dependent on the quantification. Moreover, in this example, when there is no coupling or there is weak coupling between the components, the permutation vector takes into account that the components follow exactly the same dynamics, which is what the standard Lempel-Ziv complexity appears not to do. For these interpretations, basically, we believe that dealing with an intrinsic multidimensional sequence, the Lempel-Ziv permutation complexity should be preferred to the permutation entropy and the standard Lempel-Ziv complexity.

[6] In the case of the Lempel-Ziv complexities, as these values can only take on discrete values between 2 and 500, their probability distributions are discrete. By misuse of representation, we have plotted them as continuous distributions to make their reading easier.

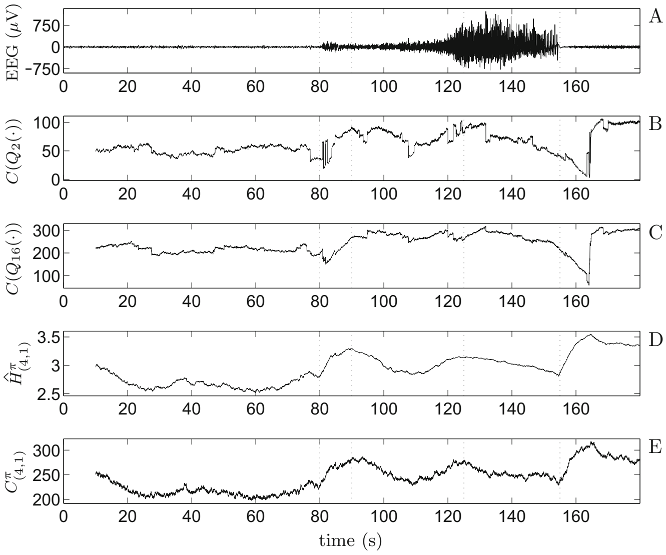


Fig. 4. Electroencephalogram records of the analysis of a secondary generalized tonic-clonic epileptic seizure. The analysis was performed with a sliding window of 10 s (1024 points) moving sample by sample. (A) The original EEG. (B, C) The Lempel-Ziv analysis was performed on a 2-level quantization (B) and a 16-level quantization (C), and the quantizers were uniform over the dynamics of the analyzed window. (D, E) For both the permutation entropy (D) and the Lempel-Ziv permutation complexity (E), the permutation vectors were evaluated from a reconstructed phase-space trajectory with an embedded dimension $d = 4$ and a delay $\tau = 1$. The confidence radius was chosen as zero. The vertical dotted lines denote the characteristic times of T_1, T_2, T_3 and T_4.

4.3 Epileptic electroencephalogram analysis

The electroencephalogram (EEG) signal analyzed in this illustration corresponds to a scalp EEG record of a secondary generalized tonic-clonic epileptic seizure, recorded from a central right location ($C4$) of the scalp. This EEG record is one of the EEGs studied by Rosso et al. in references [51–53]. It was obtained from a 39-year-old female patient with a diagnosis of pharmaco-resistant epilepsy (temporal lobe epilepsy), and no other accompanying disorders. The EEG signal is shown in Figure 4A. The epileptic seizure started at $T_1 = 80$ s, with a *discharge* of slow waves that are superposed by fast waves with a lower amplitude. This discharge lasts beyond $\Delta T = 8$ s, and has a mean amplitude of 100 μV. During the tonic-clonic epileptic seizure, there are very high amplitudes that contaminate the seizure recording, and the patient had to be treated with an inhibitor of muscle responses. After a short period, a desynchronization phase, known as the *epileptic recruiting rhythm*, appears in a frequency band centered at about 10 Hz, and it rapidly increases in amplitude. After approximately 10 s, a progressive increase of the lower frequencies (0.5–3.5 Hz) was observed [54]. For the EEG studied here, this phase appears at $T_2 = 90$ s. It is also possible to establish the beginning of the clonic phase, at around $T_3 = 125$ s, and the end of the seizure at $T_4 = 155$ s, where there is an abrupt decay of the signal amplitude.

The recorded signal has a duration of 180 s, and the sampling frequency was 102.4 Hz (1024 samples/10 s) so that we dispose of 18 432 samples. To analyze the signal, we again consider the methodology proposed in this paper; namely, the evaluation of the Lempel-Ziv permutation complexity. This result is compared to that given by the standard Lempel-Ziv performed on a static quantized version of the signal, and with the permutation entropy. The analysis was performed with sliding windows of size $N_w = 1024$ points (10 s), which moved sample by sample. Here, two quantized version are considered: a 2-level Q_2 and a 16-level Q_{16}, both of which are uniform over the range of the signal in the window of analysis. For the permutation measures, the permutation vectors were constructed with an embedding dimension and a delay, of $d = 4$ and $\tau = 1$, respectively. We chose here a radius of confidence of zero. The results are shown in Figures 4B−4E.

The interpretations of these analyses are the following:

- *The Lempel-Ziv complexity*: for both the 2-level and 16-level quantization, this measure cannot detect any change in the analyzed series. Although not plotted here, we also tested 4-level and 8-level uniform quantizers, which leads to the same conclusion.
- *The permutation entropy*: in this signal, the permutation entropy detects the appearance of the epileptic seizure at $T_1 = 80$ s, which is visible in the signal. The increase in the entropy measures a change in the nature of the signal; it is not just a change in amplitude, otherwise the nature of the sequence of the permutation vectors would not have been changed, and nor would its entropy. Similarly, the characteristic times $T_2 = 90$ s (not very visible in the signal), $T_3 = 125$ s (the clonic phase) and $T_4 = 155$ s (end of seizure) that are visible in the signal are also detected (as decreases and an increase in the permutation entropy, respectively). However, the characteristic time T_3 is not well detected by the permutation entropy.
- *The Lempel-Ziv permutation complexity*: it can be seen that the characteristic times detected by the permutation entropy are also clearly detected by the Lempel-Ziv permutation complexity. The shape of this complexity is very similar to that of the permutation entropy. In particular, the Lempel-Ziv permutation complexity detects a modification of the signal after the time $T_2 = 90$ s, a change that is not particularly detectable visually: at the peak, the analyzed window is completely inside the 'complex part' of the crisis, but the decrease indicates that the signal becomes more and more organized. Finally, the Lempel-Ziv permutation complexity better detects the modification of the signal after the time $T_3 = 125$ s than the permutation entropy.

Note that both the permutation entropy and the Lempel-Ziv permutation complexity appear to indicate the appearance of an event at time 110 s, as seen by their increases. We have no interpretation yet as to this possible event. Finally, the abrupt change that was detected by the standard Lempel-Ziv complexity at time 165 s is only a consequence of the abrupt change in the dynamics.

We can see in this example that the measure of complexity introduced in this paper increases steeply and very precisely in time when the patient starts the seizure, and even more, it can detect the different states of the tonic-clonic epileptic seizure. Note also the high level of the complexity at the end of the signal compared to that at the beginning. This level indicates that the signal remains 'disorganized'. A possible interpretation of such high complexity is that even if the epileptic sequence is apparently ended, complex activity remains consequent to the crisis. A longer post-epilepsy sequence would be needed to verify whether the complexity decreases to the low value observed before the crisis.

As this signal serves essentially as an illustration, and as our goal here is not to carry out deep EEG analyses, we will not go further with this analysis. We also do not compare our result here to those obtain in references [51–53], which merits a study in itself.

5 Discussion

Data analysis has a long history and still gives rise to a huge amount of research. Among the challenges, especially for the analysis of natural signals such as biomedical signals, there is the need to characterize the degree of organization or the degree of complexity of signal sequences, the problem of detecting sudden sequence changes that are not detectable visually, and the problem of characterization of the nature of specific changes in a sequence. The literature on information theory on the one hand, and on dynamical systems analysis on the other, provides an important number of tools and methods to solve these challenges.

In this paper, we propose a tool that mixes two very well known approaches: the permutation entropy and the Lempel-Ziv complexity. The idea is to try to take the advantage of both of these approaches, the first of which is statistical, and the second of which is deterministic.

The Lempel-Ziv complexity has long been known and was initially introduced in the compression domain. However, it has been shown to be powerful for data analysis. On the other hand, the permutation entropy allows a part of the dynamics of a signal underlying data to be captured when it is performed on reconstructed phase-space signals. Moreover, in some sense, it is based on a kind of quantization of the data, by considering only the tendencies rather than the values of the sequence. From this last, it appears natural to quantize data, as has been done via the permutation vectors of a vector sequence (natural or reconstructed) followed by the evaluation of the the complexity of such a quantized sequence. The association of these two approaches has here ‘given birth’ to what we have named the Lempel-Ziv permutation complexity, which is at the heart of our proposal.

In this paper, in particular, we have shown how the Lempel-Ziv permutation complexity of a sequence can precisely capture the degree of organization of such series. When dealing with scalar sequences, the Lempel-Ziv permutation complexity appears to give similar results to those of the permutation entropy, even if one measure is statistical while the other is purely deterministic. However, when dealing with intrinsic multidimensional signals, without procedures of phase-space reconstruction, the entropy performed on the permutation vectors built from the vector sequences cannot capture the dynamics that underlie the data. Indeed, the calculation of the frequency of occurrence of such permutation vectors is then a point-by-point analysis, and the links between successive points are lost. Conversely, as the Lempel-Ziv complexity aims to detect regularities in a sequence by analyzing how the symbols (numerical scalar samples, vectors, or any kind of symbol) can be predicted algorithmically from the past symbols, it captures the dynamics of the signal. Doing this analysis for the permutation vector sequences allows the natural solving of the question of quantization of the data, as by definition, the Lempel-Ziv complexity works with sequences of symbols lying on a discrete finite size alphabet. As shown in our illustration, we can imagine many situations for which the Lempel-Ziv permutation complexity can capture a degree of organization, while the permutation entropy fails, especially when dealing with multidimensional signals; i.e., without phase-space (re)construction.

S. Zozor is grateful to the Région Rhône-Alpes (France) for the grant that enabled this work. D. Mateos is a Fellowship holder of SeCyT, UNC.

Appendix: Technical details

Before detailing a possible practical implementation, we should point out that when two components of a vector are equal, an ambiguity remains when performing the permutation procedure. Such a situation appears with a probability of zero for continuous state iid random sequences, but it can appear in constant or periodic sequences, for instance. To avoid such an ambiguity, Bandt and Pompe proposed to add a small perturbation to the values, which is equivalent to choosing randomly the ‘smallest’ value between two equal values. For instance, in the example of a constant sequence, in this way, the permutation vectors reflect only the behavior of the perturbation, and thus both the permutation entropy and the Lempel-Ziv permutation complexity are of the noise and not of the signal under analysis. To overcome such a difficulty, we chose here to consider that the ‘smallest’ of two equal values as the ‘oldest’ one, as has also been done in the literature. In the example of a constant signal, the sequence of permutation vectors will be constant, which can then capture the low complexity of the sequence.

Conversely, an observed sequence can be corrupted by a low noise. This corrupting noise can hide the complexity of the sequence when the permutation vectors are evaluated. The example of a constant signal again illustrates such an impact of the noise. To counteract perturbations, a way to denoise or filter the observed sequence can consist of choosing a value $\delta \ge 0$ so that for two components $Y(i)$ and $Y(j)$ of a (phase-space) vector, if $|Y(i) - Y(j)| \le \delta$ then $Y(i)$ and $Y(j)$ are interpreted as equal. In a sense, δ is a *radius of confidence* in the measured data. If $\delta = 0$, this means that we have perfect confidence in the measured data, while for $\delta > 0$ we take into account possible perturbations in the measures. In other words, δ can be chosen to be equal to the resolution of the acquisition.

Practically, to evaluated $C^{\pi}_{d,\tau}(X_t)$, and to avoid two passes through the sequence, this can be done recursively, by alternating the calculation of the permutation vectors and the up-dating of the complexity:

Step 0. Construction of the first d-dimensional vector $\boldsymbol{Y} = \boldsymbol{Y}_t^{(d,\tau)}$ and evaluation of the first permutation vector $\boldsymbol{\Pi}_t = \boldsymbol{\Pi}_{\boldsymbol{Y}}$, $t = 0$; storage of this permutation vector in a stack, and initialization of the Lempel-Ziv algorithm (implicitly, the first production step).

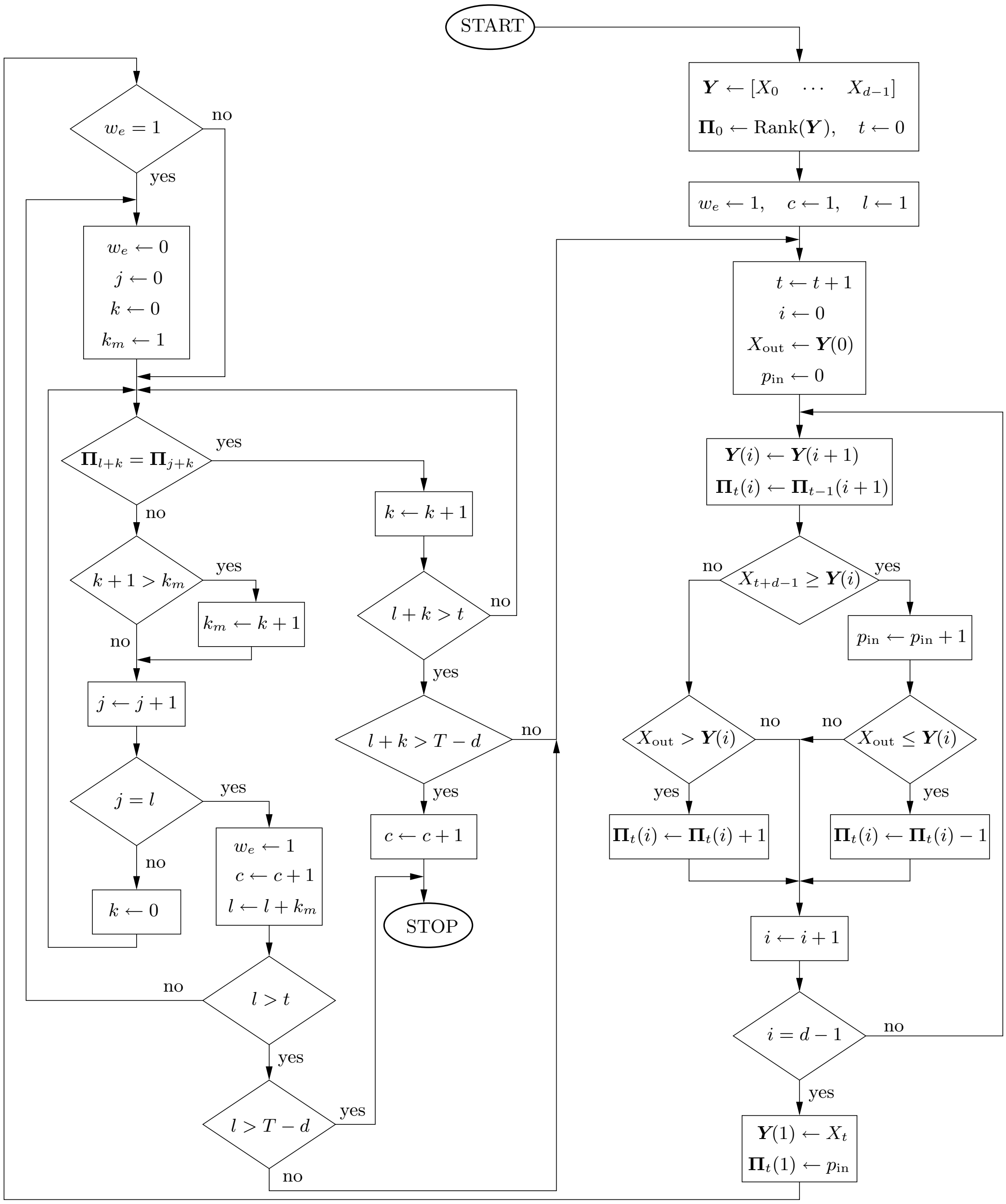
Step 1. $t \leftarrow t+1$: replacement of $\boldsymbol{Y}$ by the new vector of the trajectory, evaluation of the new permutation vector $\boldsymbol{\Pi}_t$ to be stored in the stack.

Step 2. Up-dating of the Lempel-Ziv complexity using this permutation vector, and go to step 1.

In the case where $\tau = 1$, the evaluation of the permutation vector $\boldsymbol{\Pi}_t$ at time t can be simplified by using $\boldsymbol{\Pi}_{t-1}$. Indeed, in the constructed trajectory vector $\boldsymbol{Y}$, the first point $X_{\text{out}} = Y(0)$ disappears, the other $d-1$ components are shifted, and the next point of the scalar sequence X_t appears as the last component of $\boldsymbol{Y}$. The permutation of component i (previously $i+1, i = 1, \ldots, d-1$) changes only if either $X_t \ge Y(i)$ and $X_{\text{out}} \le Y(i)$ (the rank decreases) or $X_t < Y(i)$ and $X_{\text{out}} > Y(i)$ (the rank increases). This up-dating of the rank can thus be made with d doublet of comparisons (seeking also the rank of the new point X_t).

For the Lempel-Ziv complexity, when beginning a new production step, the algorithm of [39] consists of testing all of the letters of the already constructed history as possible pointers of a production step, and retaining the letter that gives the greatest production step: this pointer gives what is then called an *exhaustive* production step.

The global recursive algorithm is described in detail by the diagram flow shown in Figure A.1; in this simple case, $\tau = 1$. For $\tau > 1$, the same scheme holds, except that we have to first store the τ permutation vectors, then store the τ vectors $\boldsymbol{Y}$, let us say $\boldsymbol{Y}_0, \ldots, \boldsymbol{Y}_{\tau-1}$, and use both $\boldsymbol{Y}_{t \,\mathbf{mod}\, \tau}$ and $\boldsymbol{R}_{t-\tau}$ to recursively evaluate $\boldsymbol{\Pi}_t$. For a non-zero radius of confidence, in the algorithm described in Figure A.1, $x > y$ (and respectively, $x \ge y$) is then



Lempel–Ziv complexity   Bandt–Pompe permutation

Fig. A.1. Diagram flow of the algorithm evaluating the Lempel-Ziv permutation complexity $C^{\pi}_{d,\tau}$ for a scalar sequence. In this diagram, $\tau = 1$ (see text for the extension to any τ), and the size of the sequence is denoted as T. w_e marks when a word is exhaustive or not, l is the beginning of an exhaustive word, j is the tested pointer, and k_m is the size of the current exhaustive word [35,39].

replaced by $x > y + \delta$ (respectively, $x \geq y + \delta$) and $x < y$ (respectively, $x \leq y$) by $x < y - \delta$ (respectively, $x \leq y - \delta$).

Note that there are various fast algorithms that rank a vector [55,56]. In general, these work by recursively partitioning the points to be ranked in a partially ordered manner (through a tree), performing a brute-force sorting in the last partitions, and coming back to the overall ensemble. In general, the computational cost is in $O(d \log d)$, instead of $O(d^2)$ for a totally brute force method. Such approaches should be used in our algorithm, using the partitions at step $t-1$ to determine that at step t, expecting a computational cost in $O(\log d)$ instead of d. However, in practice, the Bandt-Pompe entropy (and here the Lempel-Ziv permutation complexity) is studied in low dimensions, so that the computational cost of a brute force approach is relatively close to that of fast approaches.

Thus, we will not go deeper into such possible improvements of the proposed algorithm.

Finally, note that contrary to the permutation entropy, the Lempel-Ziv complexities can be evaluated online, i.e., up-dated acquisition by acquisition.

References

1. S. Zozor, O. Blanc, V. Jacquemet, N. Virag, J.M. Vesin, E. Pruvot, L. Kappenberger, C. Henriquez, IEEE Trans. Biomed. Eng. **50**, 412 (2003)
2. S.A. Kauffman, *The Origins of Order: Self-Organization and Selection in Evolution*, 1st edn. (Oxford University Press, Oxford, 1993)
3. G.W. Botteron, J.M. Smith, IEEE Trans. Biomed. Eng. **42**, 548 (1995)
4. F.F. Ferreira, G. Francisco, B.S. Machado, P. Muruganandam, Physica A **321**, 619 (2003)
5. W.B. Arthur, Am. Econ. Rev. **84**, 406 (1994)
6. M. Rajković, Physica A **287**, 383 (2000)
7. M. Rajković, Z. Mihailović, Physica A **325**, 40 (2003)
8. C. Cysarz, H. Bettermann, P.V. Leeuwen, Am. J. Physiol. Heart Circ. Physiol. **278**, H2163 (2000)
9. R.Q. Quiroga, J. Arnhold, K. Lehnertz, P. Grassberger, Phys. Rev. E **62**, 8380 (2000)
10. X.S. Zhang, Y.S. Zhu, N.V. Thakor, Z.Z. Wang, IEEE Trans. Biomed. Eng. **46**, 548 (1999)
11. G.A. Darbellay, D. Wuertz, Physica A **287**, 429 (2000)
12. W. Ebeling, L. Molgedey, J. Kurths, U. Schwarz, Entropy, complexity, predictability and data analysis of time series and letter sequences, in *Theory of Disaster*, edited by A. Bundle, H.-J. Schellnhuber (Springer Verlag, Berlin, 2000)
13. M.E. Torres, L.G. Gamero, Physica A **286**, 457 (2000)
14. T.H. Evrett, J.R. Moorman, L.C. Kok, J.G. Akar, D.E. Haines, Circulation **103**, 2857 (2001)
15. N. Radhakrishnan, IEEE Trans. Biomed. Eng. **49**, 1371 (2002)
16. J. Szczepański, J.M. Amigó, E. Wajnryb, M.V. Sanchez-Vives, Network: Computation in Neural Systems **14**, 335 (2003)
17. S. Zozor, P. Ravier, O. Buttelli, Physica A **345**, 285 (2005)
18. J. Beirlant, E.J. Dudewicz, L. Györfi, E.C. van der Meulen, Int. J. Math. Stat. Sci. **6**, 17 (1997)
19. N. Leonenko, L. Pronzato, V. Savani, Ann. Statist. **36**, 2153 (2008)
20. P.O. Amblard, S. Zozor, O. Michel, A.M. Cuculescu, On the estimation of the entropy using kth nearest neighbours, in *International Conference on Mathematics in Signal Processing (IMA'08)* (Circenter, United Kingdom, 2008)
21. T. Schürmann, P. Grassberger, Chaos **6**, 414 (1996)
22. T. Schürmann, J. Phys. A **36**, L295 (2004)
23. M. Rosenblatt, Ann. Math. Statist. **27**, 832 (1956)
24. E. Parzen, Ann. Math. Statist. **33**, 1065 (1962)
25. C. Bandt, B. Pompe, Phys. Rev. Lett. **88**, 174102 (2002)
26. K. Keller, M. Sinn, Physica A **356**, 114 (2005)
27. C. Bian, C. Qin, Q.D.Y. Ma, Q. Shen, Phys. Rev. E **85**, 021606 (2012)
28. B. Fadlallah, B. Chen, A. Keil, J. Príncipe, Phys. Rev. E **87**, 022911 (2013)
29. P. Grassberger, I. Procaccia, Physica D **9**, 189 (1983)
30. A. Lasota, M.C. Mackey, *Chaos, Fractals, and Noise; Stochastic Aspects of Dynamics*, Applied Mathematical Sciences 97, 2nd edn. (Springer Verlag, New York, 1994)
31. J.C. Robinson, *Dimensions, Embeddings, and Attractors* (Cambridge University Press, Cambdrige, UK, 2011)
32. F. Takens, Detecting strange attractors in turbulence, in *Dynamical Systems and Turbulence of Lecture Notes in Mathematics*, edited by D. Rand, L.S. Young (Springer Verlag, Warwick, 1981), Vol. 898, pp. 366–383
33. T.M. Cover, J.A. Thomas, *Elements of Information Theory*, 2nd edn. (John Wiley & Sons, Hoboken, New Jersey, 2006)
34. P. Gács, J.T. Tromp, P.M.B. Vitányi, IEEE Trans. Inf. Theory **47**, 2443 (2001)
35. A. Lempel, J. Ziv, IEEE Trans. Inf. Theory **22**, 75 (1976)
36. J. Ziv, A. Lempel, IEEE Trans. Inf. Theory **23**, 337 (1977)
37. T.A. Welch, IEEE Trans. Comput. **17**, 8 (1984)
38. G. Hansel, Lecture Notes in Computer Science (Electronic Dictionaries and Automata in Computational Linguistics) **377**, 51 (1989)
39. F. Kaspar, H.G. Schuster, Phys. Rev. A **36**, 842 (1987)
40. J.M. Amigó, L. Kocarev, J. Szczepanski, Phys. Lett. A **355**, 27 (2006)
41. J.M. Amigó, S. Zambrano, M.A.F. Sanjuán, Europhys. Lett. **79**, 50001 (2007)
42. J.M. Amigó, *Permutation Complexity in Dynamical Systems* (Springer Verlag, Heidelberg, 2010)
43. O.A. Rosso, L.C. Carpi, P.M. Saco, M. Gómez Ravetti, Physica A **391**, 42 (2012)
44. O.A. Rosso, F. Olivares, L. Zunino, L.D. Micco, A.L.L. Aquino, A. Plastino, H.A. Larrondo, Eur. Phys. J. B **86**, 116 (2013)
45. A. Gersho, R.M. Gray, *Vector quantization and signal compression* (Kluwer, Boston, 1992)
46. W. Chou, P.W. Wong, R.M. Gray, IEEE Trans. Inf. Theory **35**, 784 (1989)
47. K.T. Alligood, T.D. Sauer, J.A. Yorke, *Chaos: An Introduction to Dynamical Systems* (Springer Verlag, New-York, 1996)
48. S.H. Strogatz, *Nonlinear Dynamics and Chaos* (Westview Press (Perseus Books Publishing), Cambridge, MA, USA, 1994)
49. D. Fournier-Prunaret, R. López-Ruiz, A. Taha, Route to chaos in three-dimensional maps of logistic type, in *Proceedings of the European Conference on Iteration Theory (ECIT '04)* (Batschuns, Austria, Grazer Mathematische Berichte, Institut für Matematik Karl-Franzens Universität Graz, 2004), Vol. 350, pp. 82–95
50. R. López-Ruiz, D. Fournier-Prunaret, Math. Biosci. Eng. **1**, 307 (2004)
51. O.A. Rosso, S. Blanco, A. Rabinowicz, Signal Process. **83**, 1275 (2003)
52. O.A. Rosso, M.T. Martín, A. Figliola, K. Keller, A. Plastino, J. Neurosci. Methods **153**, 163 (2006)
53. M. Pereyra, P.W. Lamberti, O.A. Rosso, Physica A **379**, 122 (2007)
54. H. Gaustaut, R.J. Broughton, *Epileptic Seizure: Clinical and Electrographic Feature, Diagnosis and Treatment* (Charles C. Thomas Publisher, Springfield, IL, 1972)
55. C.A.R. Hoare, Comput. J. **5**, 10 (1962)
56. R. Sedgewick, Programming Techniques **21**, 847 (1978)