Detecting determinism in time series with complex networks constructed using a compression algorithm

Débora C. Corrêa
Department of Mathematics and Statistics,
University of Western Australia, Nedlands, WA 6009 Australia
debora.correa@uwa.edu.au

David M. Walker
Department of Mathematics and Statistics,
University of Western Australia, Nedlands, WA 6009 Australia
david.walker@uwa.edu.au

Michael Small
Department of Mathematics and Statistics,
University of Western Australia, Nedlands, WA 6009 Australia
Mineral Resources, CSIRO, Kensington, Perth WA 6151 Australia
michael.small@uwa.edu.au

Received (to be inserted by publisher)

The properties of complex networks derived from applying a compression algorithm to time series subject to symbolic ordinal-based encoding is explored. The information content of compression codewords can be used to detect forbidden symbolic patterns indicative of nonlinear determinism. The connectivity structure of ordinal-based compression networks summarized by their minimal cycle basis structure can also be used in tests for nonlinear determinism, in particular, detection of time irreversibility in a signal.

Keywords: Chaos; Complex Networks; Compression Algorithm; Time irreversibility; Ordinal encoding

1. Introduction

We present in this paper an investigation of properties of a compression network [Walker *et al.*, 2018] constructed using an ordinal-based symbolic encoding [Amigo, 2010]. A compression network is a complex network constructed with the codewords of a data compression algorithm [Welch, 1984] applied to scalar time series data. A complex network is a mathematical graph $G = (V, E)$ consisting of a set of vertices V and a set of edges E describing the connectivity of the vertices. In a compression network the vertices are data compression codewords and the edges indicate which codewords succeed others in a compressed time series. Previously, we showed that the properties of the structure of compression networks capture the information content of time series [Walker *et al.*, 2018]. Here we are interested in what useful information towards detecting determinism in times series can be extracted from these complex network representations of compression algorithms.

In earlier work [Walker *et al.*, 2018], we discovered that the number of unused codewords in the compression dictionary — those not emitted in the compressed time series and therefore corresponding to degree zero vertices in the compression network — in a data compression dictionary of percentile-based encoded symbolic time series is useful as a discriminating statistic in the standard surrogate data tests [Theiler *et al.*, 1992]. Here, we show that the structure of the largest connected component of a compression network, i.e., the used codewords — those emitted in the compression time series — can be used to test for time irreversibility, provided the symbolic encoding of the time series is ordinal-based.

Chaotic dynamics are necessarily nonlinear and so it is useful to develop further methods for detecting nonlinearity from time series measurements. Theiler *et al.* [1992] proposed a suite of hypothesis tests using surrogate data to reject the possibility of the observed dynamics being consistent with linear processes. Despite these tests being able to distinguish a wide range of linear processes — that the observed data is consistent with a static monotonic nonlinear transformation of Gaussian noise — it is not difficult to find a situation incompatible with the hypothesis, e.g., consider a non-monotonic nonlinear transformation. A test which can potentially deal with the aforementioned generality is to test a time series for reversibility, also referred to as testing for time irreversibility, time symmetry or time asymmetry.

In Kennel's words *"Time symmetry, often called statistical time reversibility, in a dynamical process means that any segment of time series output has the same probability of occurrence in the process as its time reversal."* [Kennel, 2004]. It is known that independently identically distributed (iid) linear Gaussian noise is time reversible [Diks *et al.*, 1995] and thus the ability to detect irreversibility in a time series is an indication of nonlinear determinism. Donges *et al.* [2013] point out, however, that testing the above time symmetry condition explicitly is practically infeasible due to the difficulty of estimating high dimensional probability distributions from limited data. Instead they advise comparing empirical distributions of certain statistical characteristics obtained from a time series and its time reversal. In [Donges *et al.*, 2013] they construct visibility graphs and compare properties of these graphs to identify time irreversibility. Kennel [2004] considered the difference in compression achieved from data compression dictionaries from forward and reverse scans of the time series, e.g., can the data compression dictionary obtained from a forward pass of a segment of the data compress a reverse pass of the time series as well as a data compression dictionary constructed from a reverse sweep and vice-versa? In the context of cardiovascular data analysis, Humeau-Heurtier *et al.* [2012] have also employed data compression ideas to test for time irreversibility. Here we investigate the connectivity structure of the largest connected component of a compression network to detect time irreversibility of time series.

In particular, we study the distribution of cycle lengths in a minimal cycle basis [Mehlhorn & Michail, 2006] of a binary reduction of the largest connected component of forward and reverse compression networks and test if there is statistically significant difference for chaotic sources. Important, however, is the process of symbolic encoding of the time series before applying the compression algorithm. We discovered that a straightforward percentile-based encoding did not provide enough resolution to detect differences between forward and reverse compression networks. Instead we find that an ordinal-based symbolic encoding endows the resulting compression networks with a cyclic structure that can detect time irreversibility.

We also note that the symbolic form of the codewords in a data compression dictionary provides a further indicator of nonlinear determinism in a fashion similar to the way forbidden patterns of symbols in ordinal partitions indicate determinism [Amigo, 2010]. In a sequential data compression scheme a symbolic codeword can only appear in the compression dictionary, i.e., be a compression network vertex, if all of its prefixes have first appeared. If the dynamics of the system forbids a particular pattern, or symbolic codeword, then longer codewords with the pattern as a prefix are also forbidden. Thus, we argue that the absence of particular codewords of a given length in the compression dictionary indicates forbidden patterns. Since iid noise should eventually generate all possible symbolic patterns, the symbolic form of (absent) codewords provides an indication of determinism.

Although there are other simpler methods to detect determinism, a benefit of the compression algorithm is its scalability to larger time series. Thus, the fact that the information retained by the compression algorithm and the corresponding compression network is capable of performing such tests provides evidence towards its more general applicability.

The remainder of this paper is outlined as follows: in Section 2.1 we describe in detail the ordinal-based symbolic encoding employed before applying the compression algorithm (Section 2.2) to the encoded time series. In Section 2.3 we describe how the data compression dictionary is converted to an (ordinal) compression network. In Section 3 we demonstrate the usefulness of the compression network approach by showing how the methods can detect changes from periodic to chaotic behaviour through forbidden codewords; how forbidden codewords can be used to indicate time irreversibility; and how the connectivity structure of compression networks can be used as a test for irreversibility of time series. We close the paper with a summary of the main results.

2. Methods

2.1. *Encoding*

Our encoding strategy is based on the ordinal partition method [Amigo, 2010] used for the complex network representation of time series, which is a simple and computationally efficient technique that embeds the temporal information present in time series into a network structure [McCullough *et al.*, 2015]. It works by finding the ordinal patterns—a simple rank-ordering of data values in a segment which maps to a permutation of an alphabet A of symbols—in an embedding phase space and mapping them to vertices in a network, while edges connect ordinal patterns that occur consecutively in the symbolized time series.

Here we do not generate complex network representations directly from the ordinal partitions as has been traditionally done [McCullough *et al.*, 2015]. Instead, our motivation is to use the temporal ordering of the ordinal patterns to have a symbolized version of the time series, and then to use the resulting symbolic time series within our compression network approach [Walker *et al.*, 2018]. As in the traditional ordinal partition technique, we select a window length $w = |A|$, the size of the alphabet, and time lag τ. Then we construct the set of ordinal patterns by segmenting the time series and mapping the segments to permutations $\{1, 2, ..., w\}$. Each segment is then mapped to an ordinal pattern by finding the rank of each element in the sequence.

To illustrate our encoding strategy, consider twenty time steps of a time series[1] as presented in Figure 1. For $w = 4$ and $\tau = 1$, the first three ordinal patterns in the forward case are $\{3, 0, 2, 1\}$, $\{0, 2, 1, 3\}$ and $\{2, 1, 3, 0\}$, obtaining by ranking the elements in each segment with w elements. The forward ordinal encoding starts with the first ordinal pattern and concatenates the index corresponding to the last element of subsequent ordinal patterns. The reverse ordinal encoding, on the other hand, starts with the reversed version of the last ordinal pattern and concatenates the first element of consecutive ordinal patterns. The forward and reverse ordinal encoding will be further used to test time irreversibility of time series. We note that the forward ordinal encoding and the reverse ordinal encoding produce different symbolic time series, that is, the reverse encoding is not simply a reversed version of the forward encoding as in the case of percentile-based encoding.

2.2. *Compression algorithm*

To transform the time series to a complex network we use a compression algorithm which is a Lempel-Ziv-Welch-like method [Welch, 1984]. We work with the symbolic time series resulting from the ordinal encoding as previously described and generate a dictionary of codewords (CD) together with an emitted time series which will be used to construct the compression network. An example is presented in Figure 2 for the forward sequence s = 30213030302130213030 depicted in Figure 1. The codeword dictionary is initiated with the symbolic alphabet, i.e., $CD = \{0 : 0, 1 : 1, 2 : 2, 3 : 3\}$, where 0 is the label for symbol 0, 1 is the label for symbol 1 and so on. Now let p be the first symbol in the time series s, that is $p = 3$ and let q be the next symbol in the time series, i.e., $q = 0$. Next, p and q are concatenated to form $pq = 30$. The next step is to check if pq is already included in the codeword dictionary.

As pq is not in the dictionary it is considered a novel codeword, and the dictionary is extended to include pq. Thus, the codeword dictionary becomes $CD = \{0 : 0, 1 : 1, 2 : 2, 3 : 3, 4 : 30\}$ where the codeword 30 receives the label 4. The emitted time series (denoted by, say, TS) is started by emitting p,

[1] Obtained from iterating a logistic map, $x_{t+1} = \lambda x_t(1 - x_t)$, with $\lambda = 4$ and $x_0 = 0.01$.

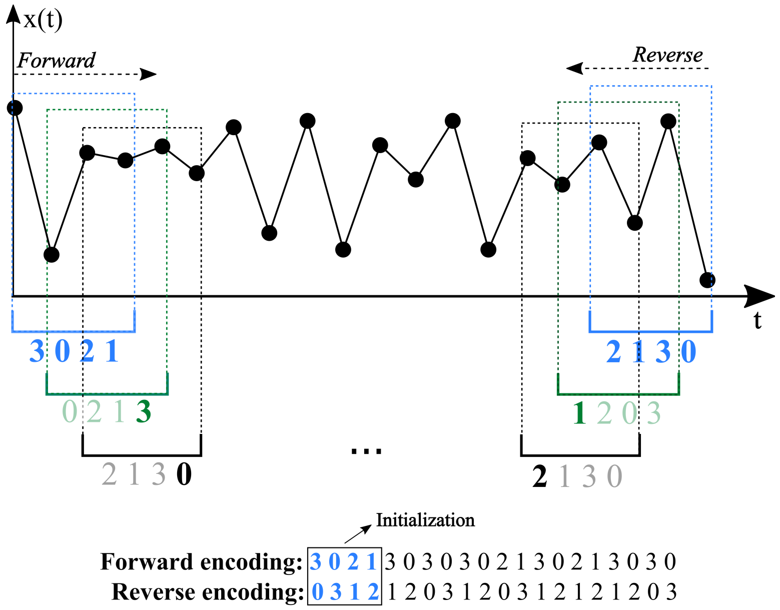

x(t)
Forward
Reverse
t
3 0 2 1
0 2 1 3
2 1 3 0
...
2 1 3 0
1 2 0 3
2 1 3 0
Initialization
Forward encoding: 3 0 2 1 3 0 3 0 3 0 2 1 3 0 2 1 3 0 3 0
Reverse encoding: 0 3 1 2 1 2 0 3 1 2 0 3 1 2 1 2 1 2 0 3

Fig. 1. Example of ordinal encoding from a time series. After initialization, which assigns all components of the first ordinal pattern to the sequence, the forward encoding sequence is generated by concatenating the last component of the temporal ordinal patterns, while the reverse encoding is constructed by concatenating to the sequence the first component of the ordinal patterns.

that is, $TS = \{3\}$, since $p = 3$ is a codeword with label 3. Now we step along the symbolic time series by setting $p = q$ and setting q as the next symbol in the time series. Thus, at this step $p = 0$, the second symbol in the time series, and $q = 2$, the third symbol in the time series. Again, we concatenate pq and check if the resulting codeword $pq = 02$ is already in the dictionary. For this example, the three next steps will generate the dictionary $CD = \{0 : 0, 1 : 1, 2 : 2, 3 : 3, 4 : 30, 5 : 02, 6 : 21, 7 : 13\}$ and $TS = \{3, 0, 2, 1\}$. After appending the codeword 13 to the dictionary, we have $p = 3$, the fifth symbol in the time series, and $q = 0$, the sixth symbol in the time series. Now the concatenated codeword $pq = 30$ is not a novel codeword, i.e., it is already included in the dictionary.

Codewords which are not novel require a different strategy. We set $p = pq$ and let q be the next symbol in the time series. That is, $p = 30$ and $q = 3$ the seventh symbol in the symbolic time series. Now we proceed similarly as before, we concatenate p and q and check if $pq = 303$ is a novel codeword. In this case $pq = 303$ is not in the dictionary, so the codeword 303 is added to the dictionary and we emit the codeword label corresponding to $p = 30$, which is 4. At this point, $CD = \{0 : 0, 1 : 1, 2 : 2, 3 : 3, 4 : 30, 5 : 02, 6 : 21, 7 : 13, 8 : 303\}$, and $TS = \{3, 0, 2, 1, 4\}$. We also note an important feature of the content of symbolic codewords in the compression dictionary. The codeword 303 only appears in the dictionary if all of its prefixes are already in the dictionary. That is, 30 must appear in the dictionary before 303, which it does, and, similarly, 3 must appear before 30 which it also does being a symbol of the alphabet. To continue, p is updated to $p = q = 3$ and $q = 0$, the eighth symbol in the time series.

This process continues until we reach the end of the symbolic time series. At each step, the dictionary of codewords is updated whenever a novel codeword is observed and the codeword labels are emitted to form the new compressed (emitted) time series. At the end of the time series, we emit the codeword label corresponding to the dictionary codeword that matches our final pq symbol sequence. Figure 2 presents the final codewords and emitted time series for the time series resulting from the forward and backward ordinal encoding as presented in the previous section (Figure 1).

An inspection of the final dictionary revels that some codewords introduced to the dictionary are never emitted. They represent the codewords which are seen only once in the symbolic time series, meaning that they are not a prefix code for another novel sequence. The compression of longer symbolic time series with this Lempel-Ziv-Welch-like algorithm is achieved from the compressed time series of the emitted codeword labels. As the codewords, independently of their variable length, are replaced by smaller labels in the emitted time series, the emitted time series is shorter than the original time series. Moreover, given that the codeword dictionary is communicated only once, it is plausible to think that the code length of [dictionary + emitted time series] will be shorter than the code length of [original symbolic time series]. Indeed for binary encoded iid noise time series, it can be shown that the amount of compression achieved is related to the entropy of the underlying source [Cover & Thomas, 2006]. In [Walker *et al.*, 2018] the size of the compression network in terms the number of vertices V tracked the sample entropy of time series as it changed with respect to a bifurcation parameter.

Forward time series:
s = 3 0 2 1 3 0 3 0 3 0 2 1 3 0 2 1 3 0 3 0

Codeword dictionary:

Codeword label	Codeword
0	0
1	1
2	2
3	3
4	30
5	02
6	21
7	13
8	303
9	3030
10	021
11	130
12	0213

Emitted time series:
{3,0,2,1,4,8,5,7,10,9}

Reverse time series:
s = 0 3 1 2 1 2 0 3 1 2 0 3 1 2 1 2 1 2 0 3

Codeword dictionary:

Codeword label	Codeword
0	0
1	1
2	2
3	3
4	03
5	31
6	12
7	21
8	120
9	031
10	1203
11	312
12	212
13	2120

Emitted time series:
{0,3,1,2,6,4,8,5,7,12,4}

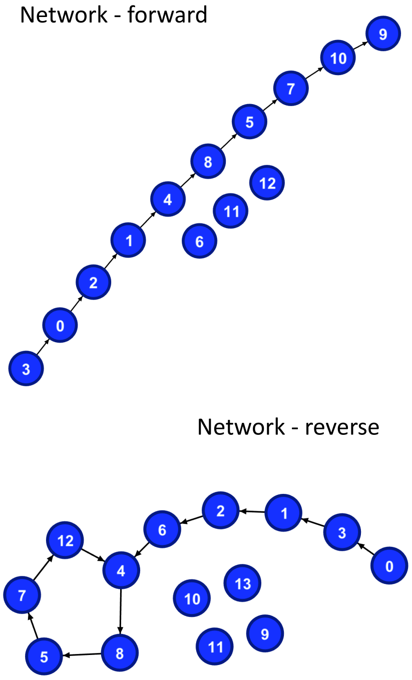


Fig. 2. The final dictionaries of codewords and the emitted time series of codeword labels for the symbolic sequences representing the forward and reverse ordinal encoding of the time series in Figure 1. We see that the forward ordinal compression network has 12 vertices while the reserve network has 13 vertices. We also observe that the reverse network contains a connected component represented by one 5-cycle including the vertices $\{4, 8, 5, 7, 12\}$. We show in the following that this difference in structure is informative of the underlying dynamics of the time series.

2.3. *Ordinal compression networks*

We convert the original time series to a complex network by examining the emitted time series produced by the compression algorithm applied to the symbolic sequence resulting from the forward or reverse ordinal encoding. Specifically, the network vertices represent the codewords, or their labels, and we connect network vertices with (directed) network edges if two codewords appear successively in the emitted time series. We refer to such a network, or its undirected binary reduction, as an ordinal compression network.

The compression network is, by construction, a directed network as it respects the sequence of successive codewords and hence also obeys causality in the original time series. It is also possible for two successive codewords to appear more than once in the emitted time series, whence the compression network can also be regarded as a weighted network. Indeed the directed version of the compression network is a form of transition network [McCullough *et al.*, 2015]. Many properties, however, of undirected complex networks translate to the directed case, and so it is often convenient to convert a directed network to an undirected network and study the properties of the transformed network. In particular, studying a binary reduction of the transformed weighted directed network expands the number of properties we can employ to help characterize compression networks and understand the underlying source system. This is the strategy we employ when we study properties of the largest connected component of the compression networks.

We observed that, for the symbolic sequence resulting from the forward ordinal encoding we used to describe the compression algorithm, the dictionary of codewords contains thirteen codewords (see, Figure 2). Thus, the forward ordinal compression network for this symbolic time series has thirteen vertices labelled from 0 to 12. The emitted time series was $TS = \{3, 0, 2, 1, 4, 8, 5, 7, 10, 9\}$ which corresponds to the edge set $E = \{3 \to 0, 0 \to 2, 2 \to 1, 1 \to 4, 4 \to 8, 8 \to 5, 5 \to 7, 7 \to 10, 10 \to 9\}$. For the reverse ordinal encoding, the dictionary of codewords contains fourteen codewords and the emitted time series resulted from the compression algorithm is $TS = \{0, 3, 1, 2, 6, 4, 8, 5, 7, 12, 4\}$. A rendering of the (directed) forward and reverse ordinal compression networks is shown in Figure 2.

As expected, the forward and reverse emitted time series resulted in ordinal compression networks with distinct structures and features (see, Figure 2). The forward ordinal compression network has three isolated or zero-degree vertices, while the reverse compression network has four isolated vertices. In both cases, the isolated vertices represent the codewords that are never revisited during the execution of the compression algorithm. In earlier work [Walker *et al.*, 2018] we demonstrated the usefulness of using a census count of these nodes for surrogate data hypothesis testing. Moreover, the connected component of the reverse directed network contains one 5-cycle comprising the vertices $\{4, 8, 5, 7, 12\}$; while the forward network has no cycles.

As we can see from this small example the structure of the compression network is informative of the complexity underlying the original time series. The purpose of the rest of this paper is to examine the usefulness of topological properties of ordinal compression networks that arise from applying the compression algorithm to symbolic encodings of nonlinear deterministic time series.

3. Results

3.1. *Forbidden patterns*

The data compression algorithm we use to construct the compression network is sequential and creates a dictionary of codewords and an emitted time series detailing which codeword succeeds another. In the compression network we assign a vertex to each codeword and a (directed) edge captures the successiveness. The codewords themselves, however, exhibit complexity and contain additional information. In addition to the alphabet of the symbolic encoding which initializes the data compression dictionary, each codeword in the dictionary is a word of the alphabet. Furthermore, since the compression algorithm is sequential, by construction, a particular codeword only appears in the data compression dictionary if all of its prefixes are already in the dictionary. For example, the codeword 011 will only appear in the dictionary if codewords 01 and 0 are already there. Or equivalently, if codeword 011 is in the dictionary then so to must 01 and 0. Similarly, if the system dynamics forbids the sequence 00 then all longer sequences which contain 00 are also forbidden and will not appear as a codeword in the data compression dictionary.

It is expected that for a binary encoding of iid noise all sequences of 0's and 1's are permitted and so eventually for a long enough time series all sequences would appear as codewords. In contrast, a chaotic dynamical system possesses its own grammar encapsulating the prohibition of certain sequences. For example, consider a chaotic system which exhibits a first return map where a binary encoding geometrically-based on the maximum of the map can be assigned. Suppose we find 00 is a forbidden pattern then a data compression dictionary formed by applying the compression algorithm to such a symbol sequence will not contain codewords with two or more consecutive 0 symbols. We can examine the codewords present in data compression dictionaries and form a statistic which measures absence of these (potentially) forbidden patterns. This has been the approach taken to detect determinism in time series using an ordinal partition framework [Amigo, 2010].

One such statistic is the percentage of forbidden patterns of a given length, say m, that are absent from the data compression dictionary. For example, consider a binary alphabet and words of length 2. There are four possible patterns 00, 01, 10 and 11. If, as for the description above, 00 is forbidden then only three of the four possible patterns will appear in the data compression dictionary (provided the time series is sufficiently long). Thus the percentage of forbidden patterns of length 2 is $(1 - 3/4) \times 100\% = 25\%$.

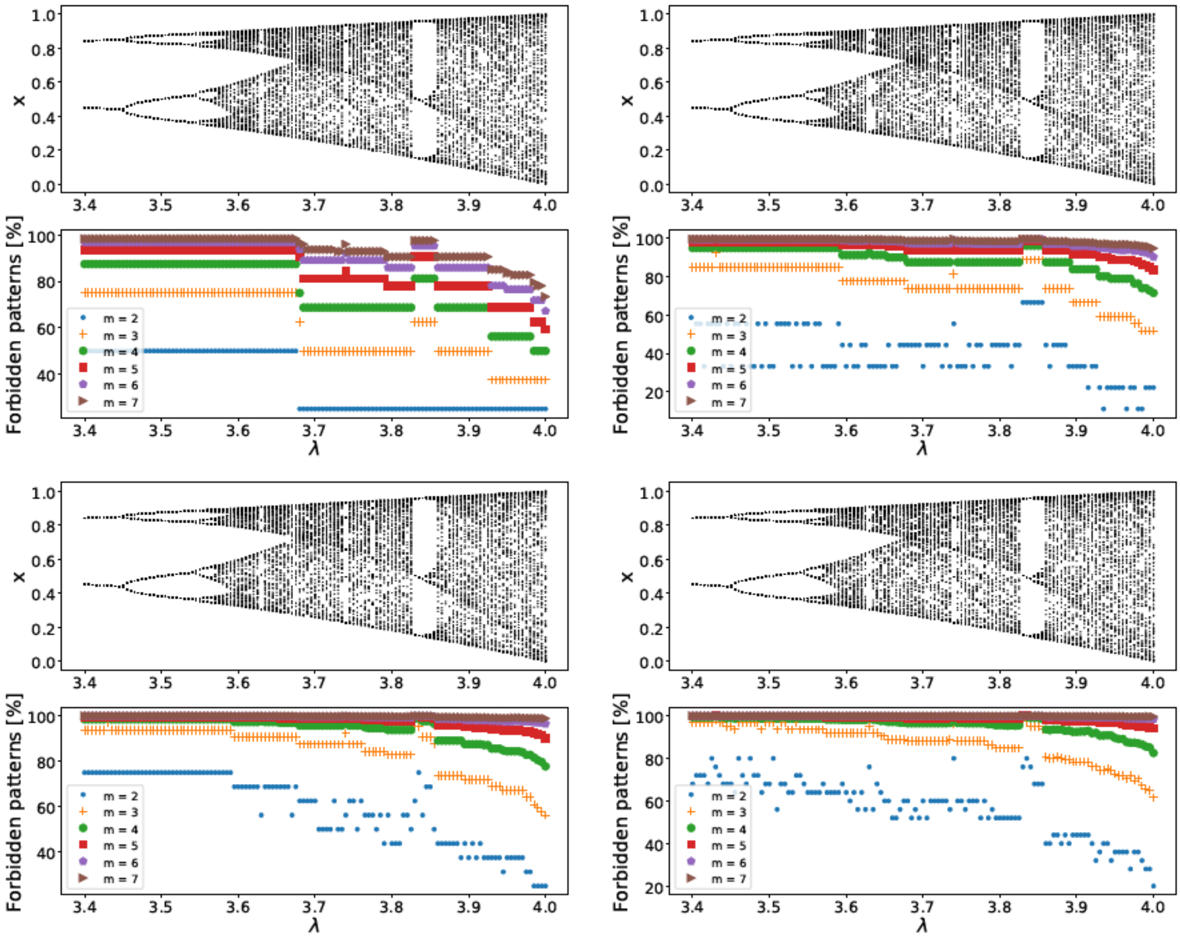


Fig. 3. The percentage of forbidden patterns obtained for the logistic map for codewords of length $m = 2$ to $m = 7$. Symbolic encoding was ordinal-based with window size (left to right, top to bottom from top left panel): 2, 3, 4 and 5. The ability to detect changes in dynamics increases with longer codeword lengths, although the limit here imposed by the length of the time series (2000 points) makes the statistics of longer codewords more volatile.

In general, for codewords of length m with an alphabet of $|A|$ symbols we define the forbidden pattern percentage, say $P(m, |A|)$, as

$$P(m, |A|) = \left(1 - \frac{|C(m)|}{|A|^m}\right) \times 100\% \tag{1}$$

where $|C(m)|$ denotes the number of codewords in the dictionary of length m.

In Figure 3 we show how the percentage of forbidden patterns in data compression dictionaries obtained from simulated time series of the logistic map, $x_{t+1} = \lambda x_t(1 - x_t)$, changes with respect to the bifurcation parameter λ, for $\lambda \in [3.4, 4.0]$ and time series of length 2000 points (after an initial transient was discarded). The type of symbolic encoding used was ordinal-based as described in Sec. 2.1, with windows of length 2, 3, 4, and 5, with $\tau = 1$. The statistic $P(m, |A|)$ was calculated for $m = 2, 3, \ldots, 7$. We can see that the forbidden pattern statistic for all symbolic encodings and codeword lengths examined is capable of detecting the transition from periodic behaviour to chaotic behaviour and also the onset of crises. The resolution or ability to detect such changes increases with increasing codeword length for each encoding, however, for longer codeword lengths the limit imposed by the length of the time series (2000 points) introduces too much volatility. We also note that, typically, periodic windows register a higher percentage of forbidden patterns as would be expected since only a restricted part of phase space is visited by such orbits. In chaotic regimes, the percentage of forbidden patterns falls according to the grammar inherent in the behaviour; for binary ordinal encoding we see that the percentage of forbidden patterns of length 2 drops to 25% as expected for $\lambda = 4$, since an ordinal 00 pattern is forbidden.

In general, for short codeword lengths (so that $|A|^m$ is not too large) we would expect iid noise to register zero percent according to this statistic—all patterns are permitted—chaotic behaviour to register a non-zero percentage between 0 and 100—some patterns are forbidden—but typically lower than the percentage registered by periodic behaviour—only a few patterns are generated. Thus, as for the ordinal partition framework the absence of codewords of a given length in a data compression dictionary compared to all codewords that could arise is a practicable method of detecting determinism, or at the very least, distinguishing dynamics from iid noise.

3.2. *Forbidden patterns and time irreversibility*

To further test the ability of a sequential data compression algorithm subject to ordinal-based encoding to expose forbidden patterns in time series data, we consider an example of time-asymmetrical data given in Kennel [2004]. We consider two independent samples of length N from the logistic map, restated here,

$$x_{t+1} = \lambda x_t(1 - x_t) \tag{2}$$

with $\lambda = 4$, i.e., $\{x_{i,1}\}_{i=1}^N$ and $\{x_{i,2}\}_{i=1}^N$. By themselves $x_{i,1}$ and $x_{i,2}$ exhibit time-asymmetrical chaotic dynamics with the set of forbidden patterns for any reasonable encoding being non-empty. Consider the mixture

$$y_i = x_{i,1} + \alpha x_{N-i,2} \tag{3}$$

with $\alpha \in [0, 1]$, i.e., y_i is formed from a mixture of an independent sample of chaotic dynamics and a time-reversed independent sample of chaotic dynamics. We can tune, via α, the level of time-asymmetry and the size of the set of forbidden patterns. When $\alpha = 0$ we simply have an independent chaotic orbit with non-empty set of forbidden patterns. When $\alpha = 1$, by construction, y_i is statistically reversible and all patterns are permissible but all may not appear (modulo length of time series and ordinal window size). Reducing α from 1 to 0 thus increases the time-irreversibility of the series y_i and increases the population of forbidden patterns.

In Figure 4 we show how the percentage of forbidden patterns changes for varying values of α from $\alpha = 1$ to $\alpha = 0$. Samples of length 98000 points were used in an attempt to ensure time series long enough to capture all permissible patterns up to length $m = 7$. That is, for an encoding with $|A|$ symbols there are a possible $|A|^7$ codewords of length 7 and so we require longer time series so each permitted codeword has a chance to appear in the data compression dictionaries. We used ordinal-based encoding of lengths $|A| = 2$ and $|A| = 4$. We see that for $\alpha = 1$ no forbidden patterns are detected (for low m, $m \leq 5$) with the percentage of forbidden patterns of the specified lengths gradually increasing to the expected values for $\alpha = 0$. Increasing the symbolic ordinal window length from 2 symbols to 4 symbols (left panel to right panel) does not appear to have a noticeable effect (for low m) for such long time series—we still see more forbidden patterns as time irreversibility increases for decreasing α—but the increased alphabet size serves to make the behaviour appear more graceful. We note that reducing the length of the time series or increasing the codeword length would naturally introduce more volatility in the forbidden pattern detection abilities of the compression algorithm.

The above example has demonstrated that, in addition to the approach of Kennel [2004] in using compression algorithms to detect time irreversibility, data compression dictionaries and the precise form of the compression codewords can also detect the absence of forbidden patterns in data and hence be able to detect nonlinear determinism. A question that arises in consequence is can the structure of the largest connected component of the compression network, which captures the sequential transitions of codewords in a time series, also be used to detect determinism or time irreversibility?

3.3. *Time irreversibility*

The structure of the largest connected component of (a binary reduction of) a compression network constructed from an ordinal-based symbolic encoding can be used to discern time irreversibility of a system from time series. The structural property of the largest connected component that we use to demonstrate

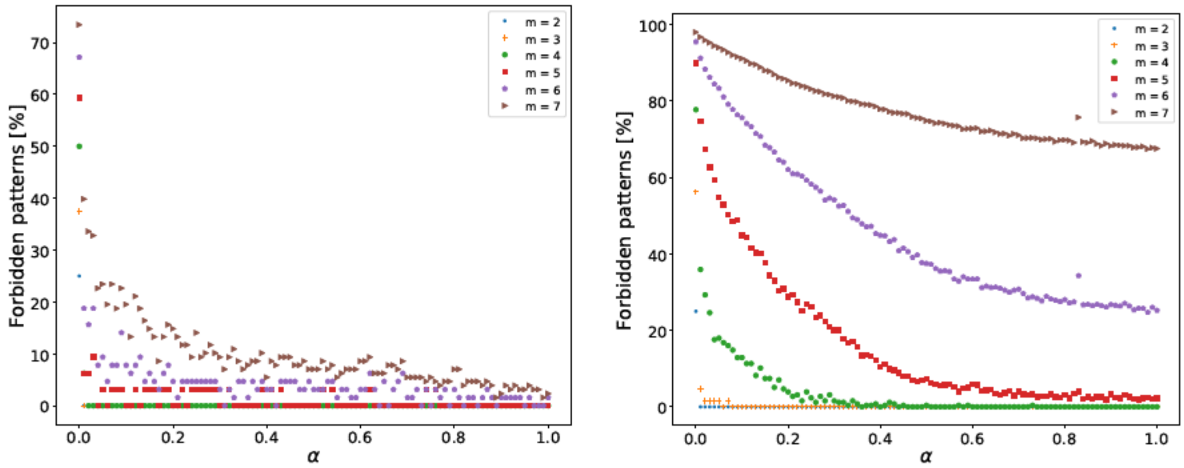
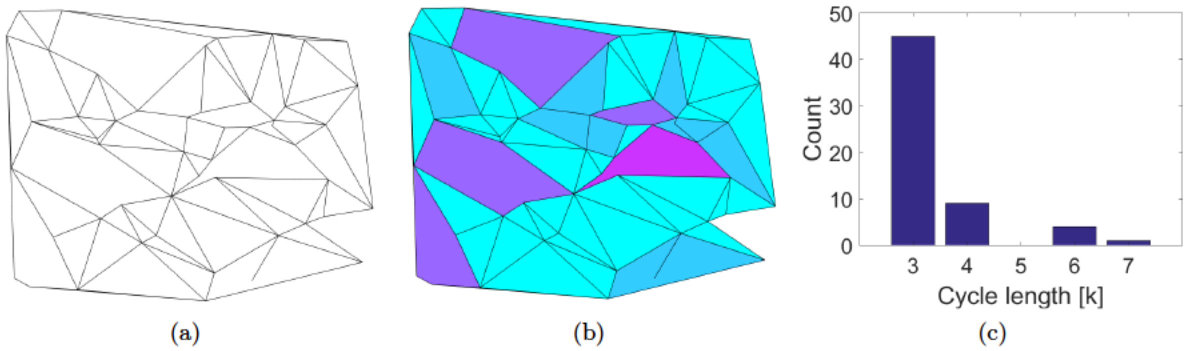


Fig. 4. The percentage of forbidden patterns of word length m detected in a mixture consisting of an independent sample of the chaotic logistic map and a second independent time-reversed logistic map sample. Ordinal-based symbolic encoding was used within the data compression algorithm. (Left panel) ordinal window of length 2. (Right panel) ordinal window of length 4.

this capability is the distribution of cycle lengths in a minimal cycle basis [Mehlhorn & Michail, 2006] of the compression network[2].



(a) (b) (c)

Fig. 5. Example of a minimum cycle basis of a graph. (a) A random planar graph (b) one solution of a minimum cycle basis, where colours indicate cycle of different lengths k (c) the empirical distribution of the cycle lengths

In a graph[3] G, a cycle is any subgraph in which each vertex has even degree. The incident vectors of cycle edges generates a vector space over $GF(2)$ called the cycle space of G. A maximal set of linearly independent cycles is called a cycle basis. If the edges of G have non-negative weights, which is the case for undirected graphs where each edge is given unit weight, then we assign each cycle a weight as the sum of the weights of its edges. The weight of a cycle basis is the sum of the weights of the cycles and a minimal cycle basis is a cycle basis with minimum weight. A minimum cycle basis is not necessarily unique, i.e., different sets of cycles forming a cycle basis can have the same minimum weight. In Figure 5 we present as an example the cycle length distribution of a planar random network. An algorithm for calculating a minimal cycle basis of an undirected graph is given in [Mehlhorn & Michail, 2006].

In [Walker *et al.*, 2018] we showed that the differences in the distribution of cycle lengths of minimal cycle bases of compression networks constructed from time series of iid noise compared to a chaotic system

[2] We adapted the algorithm to produce a minimal cycle basis to directed networks but did not find any difference to our conclusions drawn from examining the undirected binary reduction compression network.

[3] We assume one connected component.

were significant for low-order cycles, i.e., cycles of lengths 3, 4, 5, etc. In that study we used a percentile-based symbolic encoding to drive the compression algorithm. Although not reported, we investigated if the structure summarized by a minimal cycle basis of the largest connected component of compression networks could detect time irreversibility. In that instance, percentile-based symbolic encodings of forward and reverse scans of a chaotic time series were considered. We found no evidence for differences in the cycle length distribution of compression networks of forward and reverse scans of time series that are irreversible. That is, subject to percentile-based symbolic encoding the minimal cycle basis structure of the largest connected component of compression networks is unable to detect time irreversibility. This conclusion changes if we apply ordinal-based encoding. An explanation may be, that for coarse-graining based on percentiles, the symbolization depends on the global behaviour and so detailed local dynamics are lost. In contrast, the ordinal approach can capture local fluctuations and therefore codewords obtained by the compression algorithm are sensitive enough to provide accurate results for such tests.

To demonstrate we consider two case examples. The first one considers time series of iid noise which are reversible, and time series of orbits of the chaotic logistic map ($\lambda = 4$) which are irreversibile. For each class of time series we consider 100 simulations of length 2000 points. For each time series, we symbolically encode using an ordinal-based encoding of window length 4 both forward and reverse scans. The resulting symbolic sequences are compressed using the sequential compression algorithm to construct two compression networks; one for the forward scan and one for the reverse scan. A minimal cycle basis is calculated for the largest connected component of each compression network and the distribution of cycle lengths stored. Once every simulation is processed in this way we obtain for each cycle length k, $k = 3, 4, \ldots$, an empirical distribution of the number of 3-cycles, 4-cycles, etc. in the compression networks for forward scans and reverse scans. For each of these empirical distributions we compare if there are significant differences between those obtained for forward scans and those obtained for reverse scans. A Kolmogorov-Smirnov (KS) test is used to assess significance at p-value equal to 0.05.

In Figure 6 we show the result of the above procedure. For iid noise the empirical distribution of the cycle lengths for forward and reverse scans of the time series do not show any visual differences (top left panel). This is confirmed in the right panel which shows the p-values obtained for each KS-test of cycle length k. In contrast, the empirical distribution of cycle lengths for the chaotic logistic map orbits suggests differences in the two distributions (forward and reverse). The p-values presented in the corresponding right panel show that for most low-order cycles there are significant differences in these distributions. Thus, with an *ordinal-based* symbolic encoding the structure (of binary reductions) of compression networks summarized by a minimal cycle basis is capable of detecting evidence for time irreversibility. We considered window lengths longer than 4 and produced similar results for the chaotic logistic map. Longer window lengths also produced similar results for iid noise, i.e., no evidence for time irreversibility was detected as expected. We also considered ordinal windows of length 2 together with increasing lengths of time series, 2000, 4000, 8000 and 16000 points, but were unable to detect irreversibility for the logistic map. It appears that longer ordinal windows are important for successfully detecting time irreversibility using minimal cycle basis of binary reductions of compression networks.

Our second example considers scalar observations (x_t) from the chaotic Ikeda map:

$$x_{t+1} = a + b(x_t \cos\theta_t + y_t \sin\theta_t) \tag{4}$$

$$y_{t+1} = b(x_t \sin\theta_t + y_t \cos\theta_t) \tag{5}$$

where $\theta_t = k - \eta/(1 + x_t^2 + y_t^2)$ and $(a, b, k, \eta) = (1.0, 0.9, 0.4, 6.0)$. Figure 7 shows the KS tests for time irreversibility for the Ikeda map considering an ordinal-based encoding with window length 4. In this case, we used time series with $N = 16000$ data points — there was no evidence for time irreversibility for shorter time series. Comparing the distribution of cycles over forward and reverse scans of the time series with the KS test, we find p-values lower than 0.05 for cycle lengths in the range $k = 5 - 7$ and for cycle lengths $k = 12$, $k = 13$ and $k = 16$. The statistical evidence indicates that the topological properties of the largest connected components of the compression networks can be used towards detecting time irreversibility of time series for more complicated dynamics.

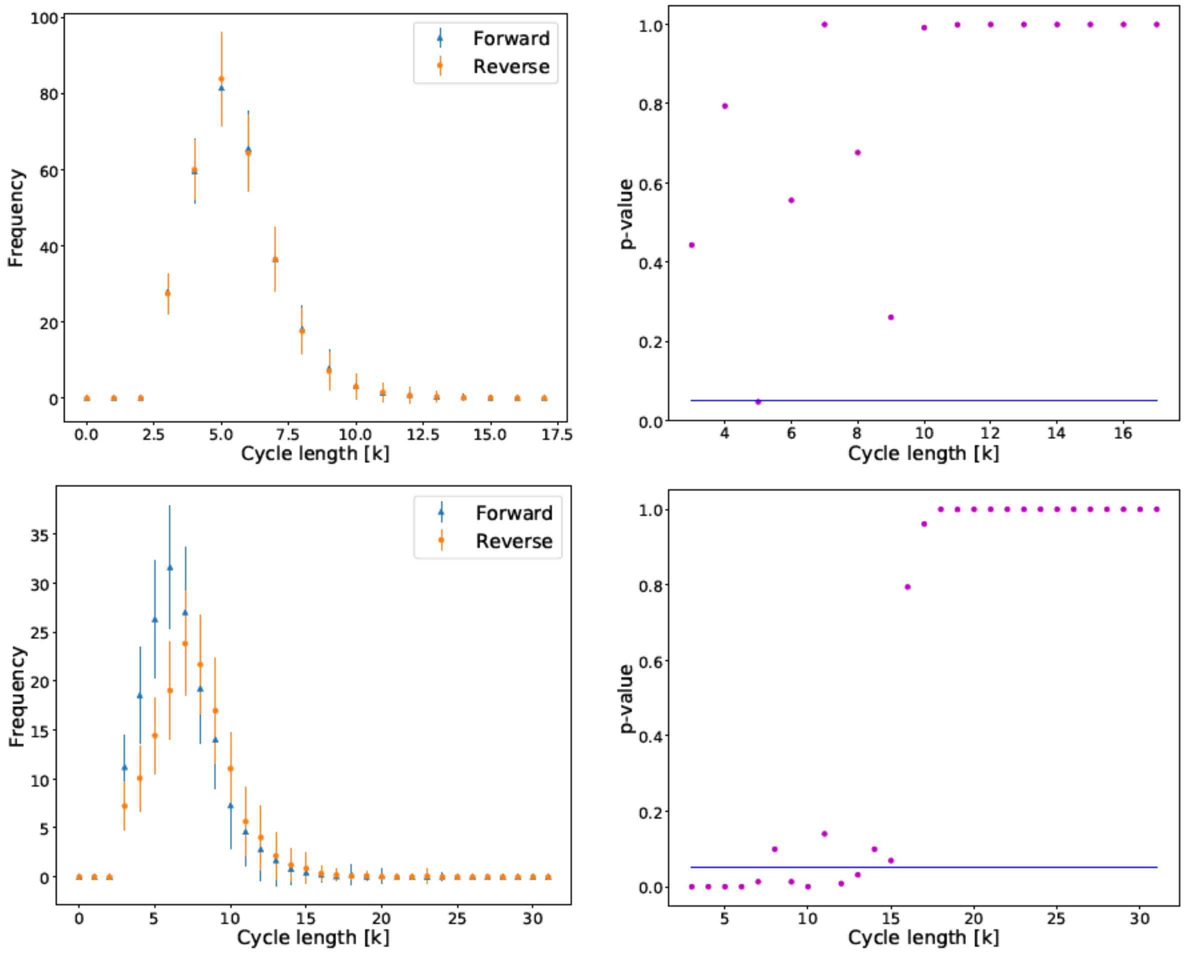


Fig. 6. Testing time irreversibility using an ordinal-based window length 4 symbolic encoding. Minimal cycle distributions for 100 realizations of $N(0,1)$ iid noise (top panels) and 100 simulations of chaotic logistic map with random initial conditions (bottom panels). Each time series has length 2000 points. The left panels show the cycle length distributions for forward and reverse scans (mean value plotted and bars extend one standard deviation). The right panels, plotting the p-values of the statistical test (solid line at p-value 0.05), exhibit evidence for time irreversibility for low-order cycles of compression networks of logistic data (most cycle lengths from $k = 3$ to $k = 13$). There is no such evidence for iid data as expected.

4. Conclusion

We have introduced the combined use of an ordinal-based symbolic encoding of time series together with a sequential data compression algorithm to construct a complex network referred to as a ordinal compression network. The compression network edge structure captures the sequential succession of data compression codewords. The compression network vertices represent these data compression codewords. We have shown that the format of these codewords, in particular those codewords absent from the compression dictionary, can be considered forbidden patterns of the dynamics. As such, the content of data compression codeword dictionaries can be used to distinguish noise from chaos. Moreover, the structure of the largest connected component of a binary reduction of an ordinal compression network, summarized by a minimal cycle basis, can be used to detect time irreversibility of time series thus further extending the capabilities of compression networks to detecting chaos. Crucial to the ability of the minimal cycle basis of a compression network to detect determinism is the use of an ordinal-based symbolic encoding; no such evidence was obtained via a percentile-based symbolic encoding. Compression networks are a new approach to consider the information content of data compression algorithms which provide a succinct way of capturing system behaviour. We have shown that by viewing a compression algorithm as a complex network, structures and properties of the network can be used for important tests of a signals' determinism. We will report on further properties

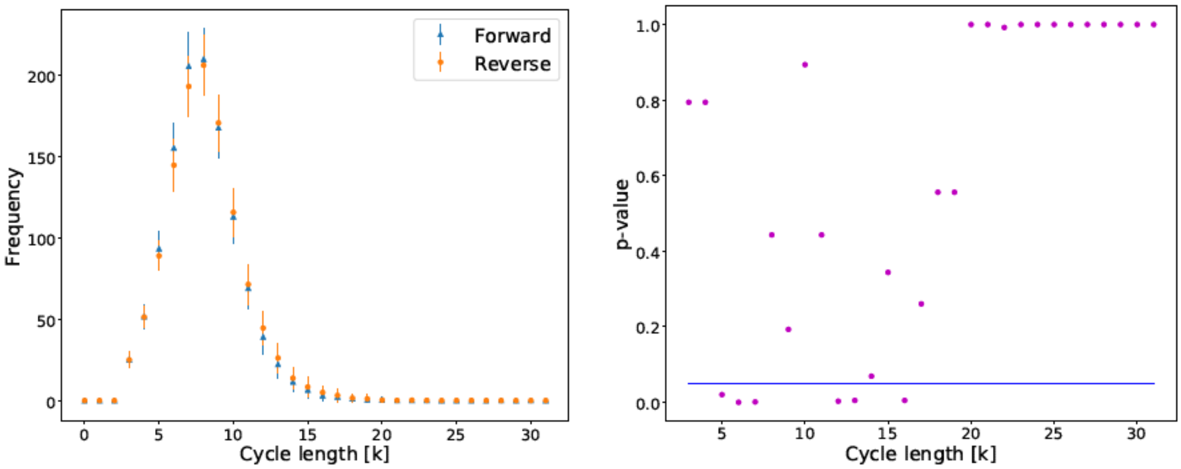


Fig. 7. Testing time irreversibility for 100 realizations of the Ikeda map. Each time series has 16000 points, and the procedure used an ordinal-based window length 4. The left panel shows the cycle length distributions for forward and reverse scans (in terms of the mean and one standard deviation). The right panel shows the p-values of the KS test, which shows evidence for time irreversibility for the cycle range $k = 5 - 7$ and for cycle lengths $k = 12$, $k = 13$ and $k = 16$. Shorter time series didn't exhibit evidence for time irreversibility. As expected, for more complicated systems, the length of the time series is an important factor.

and tests of compression networks and their robustness to noise in future work.

Acknowledgments

MS is supported by the Australian Research Council through Discovery Project DP140100203.

References

Amigo, J. M. [2010] *Permutation Complexity in Dynamical Systems: Ordinal Patterns, Permutation Entropy and All That* (Springer-Verlag Berlin).

Cover, T. M. & Thomas, J. A. [2006] *Elements of Information Theory* (John Wiley & Sons Inc, New Jersey).

Diks, C., van Houwelingen, J. C., Takens, F. & DeGoede, J. [1995] "Reversibility as a criterion for discriminating time series," *Physics Letters A* **201**, 221–228.

Donges, J. F., Donner, R. V. & Kurths, J. [2013] "Testing time series irreversibility using complex network methods," *EPL* **102**, 10004.

Humeau-Heurtier, A., Mahe, G., Chapeau-Blondeau, F., Rousseau, D. & Abraham, P. [2012] "Study of time reversibility/irreversibility of cardiovascular data: theoretical results and application to laser doppler flowmetry and heart rate variability signals," *Phys. Med. Biol.* **57**, 4335–4351.

Kennel, M. B. [2004] "Testing time symmetry in time series using data compression dictionaries," *Physical Review E* **69**, 056208.

McCullough, M., Small, M., Stemler, T. & Iu, C. [2015] "Time lagged ordinal partition networks for capturing dynamics of continuous dynamical systems," *Chaos* **25**, 053101.

Mehlhorn, K. & Michail, D. [2006] "Implementing minimum cycle basis algorithms," *J. Exp. Algorithm.* **11**, 1.

Theiler, J., Eubank, S., Longtin, A., Galdrikian, B. & Farmer, J. D. [1992] "Testing for nonlinearity in time series: the method of surrogate data," *Physica D: Nonlinear Phenomena* **58**, 77–94.

Walker, D. M., Correa, D. C. & Small, M. [2018] "On system behaviour using complex networks of a compression algorithm," *Chaos* **28**, 013101.

Welch, T. [1984] "A technique for high-performance data compression," *Computer* **17**, 8–19.