Journal of Biological Physics (2021) 47:215–235
https://doi.org/10.1007/s10867-021-09577-6

ORIGINAL PAPER

Fitness landscapes for coupled map lattices

Noelle Driver[1] · Michael Frame[2]

Received: 21 April 2021 / Accepted: 7 June 2021 / Published online: 8 September 2021

Abstract
Our goal is to match some dynamical aspects of biological systems with that of networks of coupled logistic maps. With these networks we generate sequences of iterates, convert them to symbol sequences by coarse-graining, and count the number of times combinations of symbols occur. Comparison of this with the number of times these combinations occur in experimental data—a sequence of interbeat intervals for example—is a measure of the fitness of each network to describe the target data. The most fit networks provide a cartoon that suggests a decomposition of the experimental data into a component that may be produced by a simple dynamical subsystem, and a residual component, the result of detailed, particular characteristics of the system that generated the target data. In the space of all network parameters, each point corresponds to a particular network. We construct a fitness landscape when we assign a fitness to each point. Because the parameters are distributed continuously over their ranges, and because fitnesses are estimated numerically, any plot of the landscape involves a finite sample of parameter values. We'll investigate how the local landscape geometry changes when the array of sample parameters is refined, and use the landscape geometry to explore complex relations between local fitness maxima.

Keywords Coupled logistic maps · Fractals · Chaotic dynamics · Fitness landscape

1 Introduction

The geometrical notion of landscapes has been an effective tool for visualization of many issues in biology. Examples include Sewall Wright's fitness landscapes [1, 2] and Chapter 3 of [3] to study changes of genotype frequencies, Conrad Waddington's epigenetic landscapes [4, 5] to study developmental biology (René Thom credited epigenetic landscapes as playing a pivotal role in the development of catastrophe theory [6]), and Hans Frauenfelder's energy landscapes [7–10] to study protein dynamics.

Here we'll take a different approach. Following H. Joel Jeffrey's geometric representation of nucleotide sequences [11, 12], we take time series (our examples include interbeat intervals from electrocardiographic (ECG) data and a lead from an electroencephalogram

This article belongs to the Topical Collection: The Revolutionary Impact of Landscapes in Biology
Guest Editors: Robert Austin, Shyamsunder Erramilli, Sonya Bahar

✉ Michael Frame
framemichael0@gmail.com

[1] School of Graduate Medical Education, Mayo Clinic, Rochester, MN, USA

[2] Mathematics Department, Yale University, New Haven, CT, USA

(EEG)), partition the values into four disjoint bins (in Sect. 2 we explain why we use four bins), and then note the number of consecutive pairs, triples, quadruples, and quintuples of bin numbers that occur. By converting a data sequence into bin numbers as shown in Fig. 1, we move the analysis into the field of symbolic dynamics [13–15]. The method we develop here is a way to visualize some aspects of symbolic dynamics.

For networks of coupled logistic maps, we assemble a dictionary of coupled maps, indexed by the coupling strength, logistic map parameters, and number of logistic maps in the network; the dictionary entry is the numbers of consecutive pairs, triples, quadruples, and quintuples of bin numbers. Comparison of these numbers with the corresponding numbers for data sequences generates a fitness landscape of dictionary entries with respect to the data set.

The landscape base has as coordinates the logistic map parameters and the coupling strength. For networks of M maps these coordinates are $(r_1, \dots, r_M, c)$, where r_i is the parameter of the ith logistic map $L_i(x) = r_i x(1-x)$, and c is the coupling strength. See Eq. (3). That is, the dimension of the base is $M + 1$. We can visualize the landscape as a surface over a 2-dimensional base only for a "network" of one map, that is, $M = 1$. For more maps we'll plot cross sections of the landscapes to study its properties.

We do not interpret a close match of experimental data to a dictionary entry as evidence the particulars of the dictionary entry coupled maps correspond to subsystems of the processes generating the experimental data. Rather, coarse properties of the coupled maps estimate the complexity of the natural process that accounts for some fraction of dynamics of the natural system. We have no expectation of discovering a *model* of the system, but a *cartoon* of part of its workings. Any model requires detailed knowledge of the system mechanics. We look for a simple formal guide to illuminate investigation of some major aspects of the experimental dynamics.

2 Driven IFS

Iterated function systems (IFS) were presented by John Hutchinson [16] and popularized by Michael Barnsley [17] as a simple method to generate fractal images. Suppose a self-similar shape A is composed of pieces $A_1, \dots, A_n$, each similar to A. If the transformations T_i

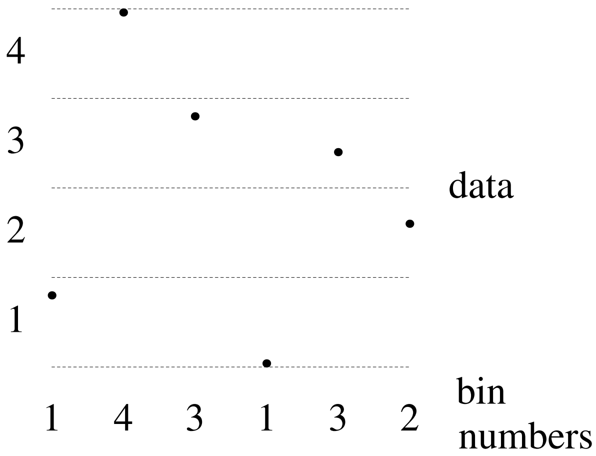


Fig. 1 Bin numbers

satisfy $T_i(A) = A_i$ for $i = 1, \ldots, n$, then $\{T_1, \ldots, T_n\}$ is the IFS for the shape A. If we apply the T_i in random order from an initial point (x_0, y_0),

$$(x_1, y_1) = T_{i_1}(x_0, y_0),\ (x_2, y_2) = T_{i_2}(x_1, y_1),\ (x_3, y_3) = T_{i_3}(x_2, y_2), \ldots$$

then A is the limit set of $\{(x_1, y_1), (x_2, y_2), \ldots\}$. This process is the *random IFS algorithm.*

Jeffrey's mechanism to visualize nucleotide sequences is a variant to the random IFS algorithm. Following Jeffrey, we use these transformations

$$\begin{aligned} T_1(x, y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(0, 0\right) & T_2(x, y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(\frac{1}{2}, 0\right) \\ T_3(x, y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(0, \frac{1}{2}\right) & T_4(x, y) &= \left(\frac{x}{2}, \frac{y}{2}\right) + \left(\frac{1}{2}, \frac{1}{2}\right) \end{aligned} \tag{1}$$

Applied to these transformations, the random IFS algorithm produces the filled-in unit square.

Any departure from uniform fill of the unit square reveals a departure from uniform randomness of the symbol sequence $i_1, i_2, i_3, \ldots$. So far as we know, Ian Stewart first questioned how to interpret the pattern when transformations are selected in non-random order [18]. There Stewart used a three transformation IFS that generates a Sierpinski gasket and consequently divides the data into three bins. Departures from uniform randomness in the data are reflected in departures from uniform fill of the gasket, difficult to see because the gasket already has many holes. Departures from the uniform fill of a square are much easier to see. Because the order in which the transformations are applied is driven by a data sequence, this method is called *driven IFS*.

To interpret driven IFS images, we use the *address* of a region on the unit square. Denote the (filled-in) unit square by S. We see these decompositions:

$$\begin{aligned} S &= T_1(S) \cup T_2(S) \cup T_3(S) \cup T_4(S) \\ &= (T_1T_1(S) \cup \cdots \cup T_1T_4(S)) \cup \cdots \cup (T_4T_1(S) \cup \cdots \cup T_4T_4(S)) \end{aligned}$$

and so on. For example, the address of the region $T_iT_jT_k(S)$ is ijk. Note the order of the address digits is the order of the composition of the transformations that determine the region. The number L of digits in an address is the *length* of the address. The $L = 1$ and $L = 2$ addresses are shown in Fig. 2.

For example, in Fig. 3 we see the IFS driven by the nucleotide sequence for the enzyme amylase, which converts starches into sugars. As the amylase sequence is read, apply T_1 when the symbol C (for cytosine) is read, T_2 for A (adenine), T_3 for T (thymine), and T_4 for G (guanine). Note the driven IFS points are not at all uniformly spread across

Fig. 2 The $L = 1$ addresses (left), $L = 2$ addresses (right)

3	4
1	2

33	34	43	44
31	32	41	42
13	14	23	24
11	12	21	22

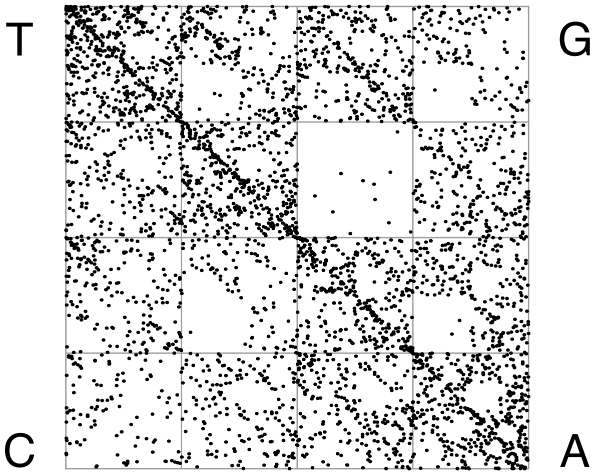


Fig. 3 Amylase driven IFS

the square. For example, the sparseness of the points in address 41 represents the fact that guanine only rarely follows cytosine in this sequence.

Nucleotide sequences occur in an alphabet of four symbols, so a map from nucleotides to transformations is straightforward. For numerical sequences, including intervals between heartbeats, some choices must be made. Suppose the data sequence is $x_1, x_2, \ldots, x_n$. The range of the sequence is $[\min(\{x_i\}), \max(\{x_i\})]$, which has length $R = \max(\{x_i\}) - \min(\{x_i\})$. To divide the range into four bins, one mapped to each T_i, we need three bin boundaries. While many choices are can be made, we'll focus on two families of bins:

equal-size bins. Set $B_1 = \min(\{x_i\}) + R/4$, $B_2 = \min(\{x_i\}) + R/2$, and $B_3 = \min(\{x_i\}) + 3R/4$.

equal-weight bins. Set the bin boundaries so the four bins $[\min(\{x_i\}), B_1)$, $[B_1, B_2)$, $[B_2, B_3)$, and $[B_3, \max(\{x_i\})]$ have about the same number of points.

Equal-size bins are sensitive to the metric relations between data sequence entries, but a few values that are much larger than, or much smaller than, most of the others can result in only a few points in bin 4 or in bin 1. This reduces the discriminatory power of the driven IFS. Typically, equal-weight bins, also called a maximum-entropy partition, do not suffer from this reduction of discriminatory power; they are sensitive to the ordering relations, not the magnitudes, of the data. Whether we want to avoid the visual consequence of extreme data values depends on whether we think an extreme value could be the result of an external force unrelated to the system dynamics, or represents an important feature of the system dynamics.

To find the boundaries of equal-weight bins, let $\{y_i\}$ be the list made by sorting the list $\{x_i\}$ in increasing order. Then $B_1 = y_{[N/4]}$, $B_2 = y_{[N/2]}$, and $B_3 = y_{[3N/4]}$, where N is the number of data points and $[q]$ is the integer part of q. If the sequence contains many equal values, and a bin boundary happens to be one of those values, then no set of boundary values may produce bins with equal point counts.

We will use driven IFS to visualize patterns generated by data sequences and the top matches from the dictionary entries. Mostly, though, we'll focus on the construction of fitness landscapes by comparing address occupancies.

3 Logistic maps

As Robert May [19] pointed out in 1976, the logistic map

$$L_r(x) = r \cdot x \cdot (1 - x)$$

is a very simple system that, depending on the value of the parameter r, can exhibit very complicated behavior. Chapter 1 of [20] gives a comprehensive introduction to the varieties of dynamics seen in time series

$$x_0, x_1 = L_r(x_0), x_2 = L_r(x_1), \dots \tag{2}$$

generated by iterating the logistic map.

We'll focus mostly on the range $3.5 \le r \le 4$. The chaotic dynamics of the logistic map occurs for some, but not all, of these r values. Within this range are infinitely many windows of periodic behavior. Networks of these logistic maps may exhibit behaviors complicated enough to capture some of the dynamics of cardiac or other biological data. To underscore the range of possible application, we'll make two comments.

By excluding the initial range of period-doublings from our network parameters, we have not ignored the dynamics of biological systems that undergo period-doublings, for example, the Hindmarsh–Rose neuronal model [22, 23], because in the range $3.5 \le r \le 4$ the logistic map exhibits infinitely stable periodic cycles, each of which is followed by its own family of period-doublings.

Second, some biological systems have return maps that are not unimodal. For our networks of logistic maps, we have constructed the return map from the sequence of iterates. Put another way, because we know the generating dynamics, we know the natural sampling time. This is not so clear for some biological data (though it is for cardiac interbeat intervals), so the effect of sampling time choice may be better mapped to the network second-return map (x_i, x_{i+2}) or higher return maps, which need not be unimodal.

4 Model networks of coupled logistic maps

Recall $L_i(x) = r_i x(1 - x)$. Then the general network of M coupled logistic maps is

$$\begin{aligned}
x^1_{n+1} &= c_{1,1}L_1(x^1_n) + c_{1,2}L_2(x^2_n) + \cdots + c_{1,M}L_M(x^M_n) \\
x^2_{n+1} &= c_{2,1}L_1(x^1_n) + c_{2,2}L_2(x^2_n) + \cdots + c_{2,M}L_M(x^M_n) \\
&\dots \\
x^M_{n+1} &= c_{M,1}L_1(x^1_n) + c_{M,2}L_2(x^2_n) + \cdots + c_{M,M}L_M(x^M_n)
\end{aligned} \tag{3}$$

where x^j_n is the current value of the jth variable, and x^j_{n+1} is the next value. To guarantee that all the variables remain in the interval [0, 1], we impose the condition $c_{i,1} + c_{i,2} + \cdots + c_{i,M} = 1$ for each i, $1 \le i \le M$.

Varying the $c_{i,j}$ can produce any network coupling topology. For example, all $c_{i,i} = 1$ is the network of M independent logistic maps, while each $c_{i,j} = 1/M$ is the homogeneously coupled network.

Here we focus on the cross section of parameter space determined by *nearest-neighbor coupling* with all the coupling strengths determined by a single parameter, c. Specifically,

1. all $c_{i,i} = 1 - c$,
2. for $2 \leq i \leq M - 1$, $c_{i,i-1} = c_{i,i+1} = c/2$, and
3. $c_{1,2} = c_{1,M} = c/2$ and $c_{M,M-1} = c_{M,1} = c/2$

This last condition guarantees that logistic maps 1 and M are nearest neighbors, so topologically the network consists of logistic maps arranged around a circle.

We'll think of these logistic maps as agents in a larger system, and we'll study the average of the states of the agents, a macroscopic system property. That is, to this network we associate the time series

$$z_n = (x_n^1 + x_n^2 + \cdots + x_n^M)/M$$

then drive the IFS with this series.

5 A fitness function

The number of points in, say, address 12, divided by the total number of points in the driven IFS, estimates the probability of the pair transition 2 → 1. In principle this estimate can be generalized to addresses of any length, but because the data strings are of finite length, the reliability of this estimate decreases with address length. Our approach to this problem is to limit our calculations to $L = 5$ addresses, sample the coefficient of variation of these address occupancies, and discount addresses with fewer than five points. In Sect. 7 we'll describe why we made these choices.

Suppose A is an IFS driven by a data set, and B is a driven IFS from our dictionary. To assess the fitness of B at capturing the dynamics of A, we define the *length-L fitness* by

$$\varphi(A, B, L) = \frac{N - L + 1 - \frac{1}{2}\sum_{i_1,i_2,\ldots,i_L=1}^{4} |A_{i_1 i_2 \ldots i_L} - B_{i_1 i_2 \ldots i_L}|}{N - L + 1} \tag{4}$$

where N is the number of points in the data list and $A_{i_1 i_2 \ldots i_L}$ is the number of points in the region of attractor A with address $i_1 i_2 \ldots i_L$. The term $N - L + 1$ is the number of consecutive length-L strings in a sequence of length N: we see $N - 1$ consecutive pairs, $N - 2$ consecutive triples, and so on.

One way to visualize what $\varphi(A, B, L)$ measures is to overlay the driven IFS attractors A and B and see how well the points of one overlap the points of the others. This is akin to a difference density map in protein crystallography.

For example, take A to be the attractor of the driven IFS generated by iterating two logistic maps with $r_1 = 3.82$, $r_2 = 3.53$, and $c = 0.7$, and B the attractor with $r_1 = 3.93$, $r_2 = 3.65$, and $c = 0.2$. These driven IFS are shown in Fig. 4.

For this example, suppose A is a dictionary entry with the data of its time series iterates partitioned with equal-size bins. Further, suppose B represents the target data with bins selected so the number of data points in each bin of B matches, as closely as possible, the number of points in the corresponding bin of A. With data generated by 10,000 iterates of each network, we find

$$\varphi(A, B, 2) = 0.901, \ \varphi(A, B, 3) = 0.817, \ \varphi(A, B, 4) = 0.722, \ \varphi(A, B, 5) = 0.604$$

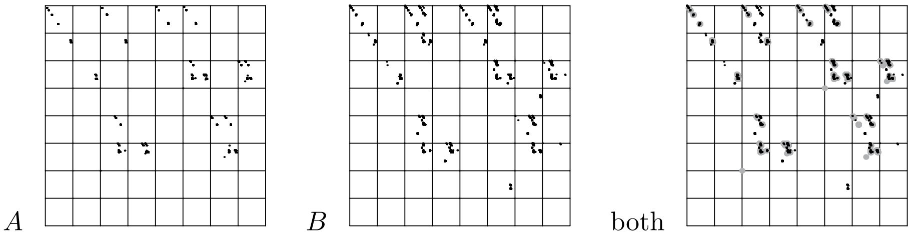


Fig. 4 Driven IFS A (left), B (middle), and their overlay (right). In the right graph we've made the points of A larger and gray to ease visual comparison

(With this choice of bins for B we always have $\varphi(A, B, 1) = 1.0$, so we needn't mention the $L = 1$ fitness again.) We see that increasing the address length reduces φ. This is expected: longer addresses subdivide shorter addresses and so improve the discriminatory power of φ.

6 Complications: transients, intermittency, and multiple attractors

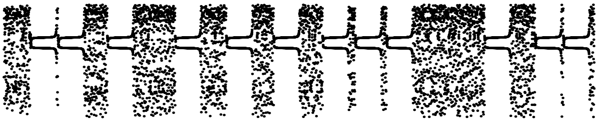
Comparisons based on address occupancy assume that the driven IFS points have converged to the attractor generated by the network. An obvious complication is that the initial IFS points may not lie on this attractor. Dropping some initial points before collecting address occupancy data is the standard approach to these transient values. In the early days of fractal geometry (the early 1980s), programmers dropped the first few points generated by an IFS algorithm in order to remove the annoying nimbus of points that surrounded the attractor. Soon enough people realized that if the goal is to generate the attractor of the IFS, the annoying nimbus can be avoided by taking the initial point to be that fixed point of one of the IFS transformations. For our applications, we do not know if any fixed point belongs to the attractor, so we return to the early idea of dropping some initial points.

A more subtle issue is *intermittency*: the dynamics are approximately periodic, punctuated by short excursions into more complex behavior. In Fig. 5 we see a time series for the average value of two logistic maps, $r_1 = 3.83$ and $r_2 = 3.75$, coupled with $c = 0.14817$, illustrating a 2-cycle with intermittency. Interleaving two significantly different dynamics complicates any interpretation of the driven IFS. In Sect. 7 we describe an approach to handle intermittency; another approach is outlined in Sect. 11.

Another issue is that a network can have several attractors. Single logistic maps exhibit only one attractor for each r-value. The situation can be more complex for networks of even two logistic maps. For example, in Fig. 6 we see the equal-size bin driven IFS for the two-map network $r_1 = 3.93, r_2 = 3.24$, and $c = 0.9$, with these initial values: $x_0^1 = 0.8, x_0^2 = 0.4$ for the left plot and $x_0^1 = 0.8, x_0^2 = 0.8$ for the right.

For both, we drop the first 100 points and plot the next 10,000. This network exhibits multiple attractors. That some parameter values can give rise to different attractors cannot

Fig. 5 Intermittency for two coupled logistic maps



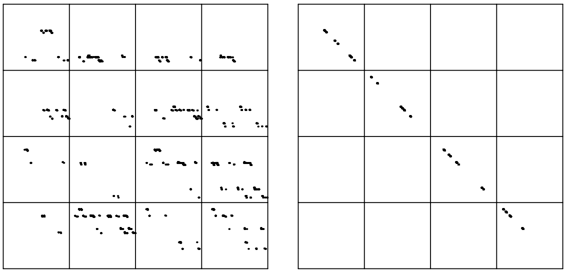


Fig. 6 Same network, different initial points

be ignored. Simply averaging the address occupancies when substantially different attractors occur can only give a nonsensical result. If time series from different initial values give substantially different address occupancies, we assume they generate different attractors, both valid for the parameter values. In this case, a single dictionary entry (logistic parameters, coupling constant) has multiple sets of address occupancies, each of which must be compared against the target data.

A final comment about Fig. 5 in light of the notion of multiple attractors: maybe intermittency can be interpreted as two attractors, in this case a 2-cycle and a chaotic regime, and the 2-cycle points lie very close to points of the chaotic attractor. The finite accuracy of computer-generated iterates can induce jumps from one attractor to the other. Unless we observe similar behavior in biological time series, or unless an open set of initial values lead to a single attractor, these particular networks will not be good cartoons of those systems.

7 Dictionary construction

Because the potential sample space of networks contains infinitely many points, we made choices to capture some interesting aspects of the space while keeping the dictionary a reasonable size for computational use. To course-grain the sample space, we limited the precision of c to 0.1 and the precision of r to 0.01, for r ranging between 3.57 and 4, where the lower bound of r is chosen near the Myrberg–Feigenbaum point ≈ 3.5699, the parameter value marking the onset of chaos for a single logistic map. The range $3.57 \leq r \leq 4$ contains chaotic behavior, as well as infinitely many windows of periodic behavior. The dictionary for single logistic maps is complete from $3.57 \leq r \leq 4.00$ in steps of 0.01. For more than one logistic map, the combinations of parameter values which give chaotic or periodic behavior are not as straightforward. Combinations of logistic coefficients r in the chaotic range can produce limit cycles for certain values of c, and combinations with some logistic coefficients r less than the Myrberg point can produce chaotic time series when coupled to maps with r-values in the chaotic range. The oscillators influence each other in unpredictable ways. In Fig. 7 we see the driven IFS and the return map, a plot the points (z_i, z_{i+1}), for two coupled logistic maps, both with r-values in the chaotic range. For the smaller coupling value, the return map suggests limit cycle behavior; while a slight increase in c produces something much more intricate.

We produced 500 dictionary entries per value of c for each of 2, 3, and 4 coupled logistic maps by sampling from the values of r. At least one value of r must be in the chaotic range for the system to behave chaotically (according to all observations so far), so we sampled from $r = 3.57, 3.58, ...3.99, 4.00$. For the other maps, we sampled from

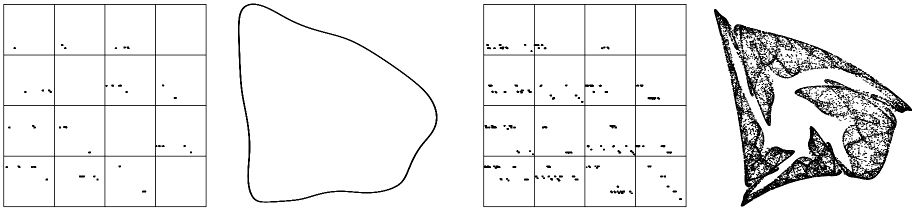


Fig. 7 The driven IFS and return map for the average of two logistic maps, $r_1 = 4.0$, $r_2 = 3.88$, and $c = 0.895$ (left) and $c = 0.905$ (right)

$r = 3.40, 3.41, ..., 3.99, 4.00$, which was chosen to include a significant parameter range of periodic dynamics, enabling the inclusion of some of the state space for multiple coupled maps which behave chaotically but originate from periodic functions. We applied a Fourier transform to the data series to detect periodicity, and discarded combinations of parameter values which produce cycles of size 16 or less.

The reliability of the average address occupancy data in the dictionary increases with the number of points in the generating sequence; however, using longer time series increases computation time. We looked at the maximum and average standard deviations of the address occupancies for 50 runs of different lengths (5,000 to 50,000 in steps of 5,000) for a few sample systems. The average standard deviations seemed to level off at about 25,000 points, so the dictionary entries were produced from time series of this length.

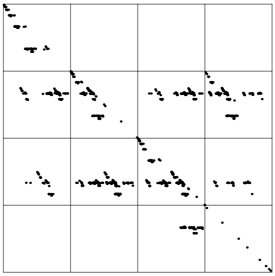
We generated each entry in the dictionary from a 25,000-point time series, divided into 50 equal parts. As we saw in Sect. 6, time series may contain regimes of different behaviors. Again, we are not interested in fixed points or cycles because they are characterizable by other means. Also, driven IFS is not an effective tool to detect a low-period cyclic window in a more complex pattern because a cycle of length n produces a driven IFS that consists of n points, not easily visible in a more full background. To account for this, for each 500-point part of the time series, we applied the Fourier transform to detect periodicity. If the part of the time series produced a clean (i.e. not noisy) cycle of size 16 or less, we removed that part from the address occupancy calculations. For the parts not discarded by the Fourier sieve, we took the average number of points in each address from all of the parts to store as the address occupancies, up through the $L = 5$ addresses. In this way, each data string was passed through the Fourier transform twice: first, to ensure that the overall behavior of the 25,000-point series was not as a whole periodic, and second, to remove the 500-point sections which were periodic.

We dropped the first 50 points of the data sequence before setting the bin boundaries and computing the address occupancies because these points are often transient and outside the range of long-term system behavior. The bins we used for the dictionary entries were equal-sized. We chose to do one long 25,000 point run that was then split into parts rather than fifty 500-point runs in order to give one set of bin boundaries for the system.

After the bin boundaries were set, the address occupancies up through $L = 5$ were calculated for each of the 50 parts of the data list, minus those removed from the Fourier sieve, and the average address occupancies were stored to comprise one entry of the dictionary. Rather than store the absolute address occupancies themselves, we store the relative occupancies, which enables the dictionary entries to be compared to a data list of any length.

Fig. 8 Length-2 address occupancies (left) and driven IFS (right)

0.173	0	0	0
0.360	0.467	0.469	0.531
0.031	0.969	0.796	0.031
0	0	0.171	0.002



8 Fitness landscapes

To illustrate landscape construction, we take as target a network of three logistic maps, Eq. (3) with $r_1 = 3.95$, $r_2 = 3.60$, $r_3 = 3.91$, and with coupling strengths $c_{1,1} = c_{2,2} = c_{3,3} = 0.8$ and all other $c_{i,j} = 0.1$. One way to guarantee that the length-1 address occupancy fractions of the target sample approximately match those of each dictionary entry is to use equal-weight bins. A simulation of this network with 50,000 iterates and equal-weight bins produces the length-2 address occupancies shown in Fig. 8. The map of length-2 addresses is the right image of Fig. 2.

We build a cartoon dictionary, a simplified version of the main dictionary described in Sect. 7, with only three coupled logistic maps.

For the parameter space we take $0 \le c \le 1$ and $3.5 \le r \le 4$. In Fig. 9 we plot the fitness of the dictionary entries with the parameter space sampled on a 20×20, a 30×30, and a 40×40 grid. While the general trends in the coarser grids are repeated in the finer, as they must be, finer grids reveal more variation. The complex interleaving of periodic and chaotic windows for a single logistic map suggests, though does not prove, an ever-increasing complexity of these landscapes when sampled on smaller scales. Fractality of landscapes is investigated in [24], Sects. 8.3 and 8.5 of [25], and [26]. We'll explore this in Sects. 9 and 10.3.

The possible fractality of the fitness landscape can be investigated visually by a sequence of magnifications. In Fig. 10 we see two zooms into the right plot of Fig. 9. For that image, the axes ranges are $0 \le c \le 1$, $3.5 \le r \le 4$, and $0 \le \varphi \le 1$. In the left image of Fig. 10, we have $0.5 \le c \le 1$, $3.75 \le r \le 4$, and $0 \le \varphi \le 1$; for the right image the ranges are $0.75 \le c \le 1$, $3.875 \le r \le 4$, and $0.5 \le \varphi \le 0.8$. In terms of the r-c parameter plane, the left image is the upper right quadrant of the parameter plane of Fig. 9,

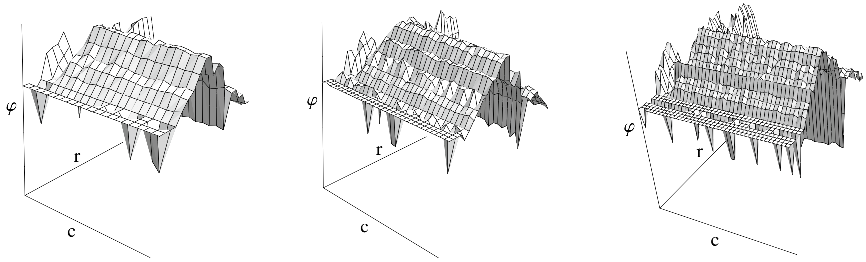
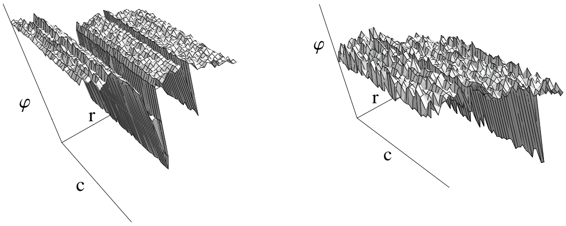


Fig. 9 A fitness landscape with the c and r ranges divided into 20, 30, and 40 intervals. Slightly different viewpoints aid visualization of the surface

Fig. 10 Landscape magnifications



and the right image is the upper right quadrant of the left image. At least over this range, magnification does appear to reveal roughness across levels.

The plots of Fig. 9 are a cross section of the full landscape: in the parameter space $[3.5, 4]^3 \times [0, 1]$, the r-values lie on the diagonal segment between (3.5, 3.5, 3.5) and (4, 4, 4). Other paths are possible.

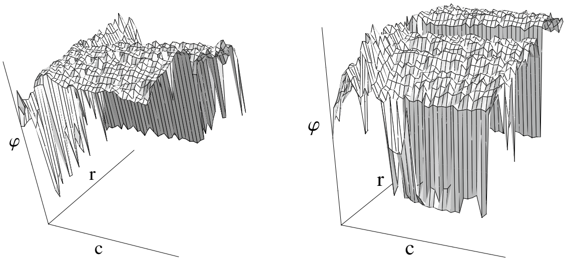
The left graph of Fig. 11 shows the fitness plot above the cross section $r_1 = r_2 = r$ and $r_3 = 7.5 - r$, so as $r_1 = r_2$ ranges from 3.5 to 4, r_3 ranges from 4 to 3.5. The right graph of Fig. 11 shows the fitness plot above the cross section $r_2 = r$ and $r_1 = r_3 = 7.5 - r$. These cross sections suggest that the complete fitness landscape is fractal. More detailed tests are needed in order to establish this. Fractality of landscapes has consequences for maintaining genetic diversity in a population: roughness implies the existence of many nearby local maxima and gradient ascent can trap, at least temporarily, individuals in many of these maxima, thus maintaining genetic diversity. In terms of our dictionary matches, this offers a range of ways to parse important processes in the system that generates the data.

9 Landscape roughness

Gregory Sorkin [24] studied the effectiveness of simulated annealing to find minima of energy landscapes. He argued that if a landscape is fairly regular, if it has peaks of similar heights and valleys of similar depths, then moving from one valley to another can require a considerable energy level, while selecting a global minimum from among all the valleys can require a much lower energy level. Sorkin concluded that simulated annealing is not an efficient search protocol for regular landscapes.

On the other hand, simulated annealing can be efficient on fractal landscapes: at high energy the highest peaks are crossed, and as the energy decreases a hierarchy of peaks is crossed, finding lower local minima. Sorkin defined a landscape to be fractal of type h if

Fig. 11 Landscape over other cross sections



$$E(d) = \langle \left(\varphi(\vec{x}) - \varphi(\vec{y})\right)^2 \rangle \sim \mathrm{dist}(\vec{x},\vec{y})^{2h} \tag{5}$$

To test the fractality of a landscape, plot log(average fitness) against log(distance between parameters) for a range of distances. We measure distance with the *taxicab metric*

$$d = \mathrm{dist}\left((r_1, \dots, r_n, c), (r'_1, \dots, r'_n, c')\right) = |r_1 - r'_1| + \cdots + |r_n - r'_n| + |c - c'|$$

If the points fall close to a straight line, then at least over that range of distances the landscape is fractal of type twice the slope of the line.

Sorkin describes another approach, based on the statistics of random walks on energy landscapes. The simplicity of our parameter space supports our use of the simpler log-log approach.

We sample pairs of points $\vec{x}, \vec{y}$ randomly distributed across the parameter space, with the condition that the distance between the pairs sampled is considerably smaller than the diameter of the parameter space. This avoids "edge effects" which perturb the graph. For all pairs the same distance apart, we average the square of the difference in fitnesses and assess the linearity of the log-log plot. (See Fig. 12)

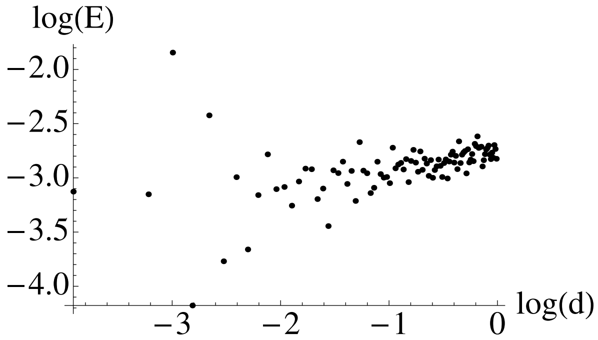
Many examples are explored in [24, 26]. We'll investigate the fractality of fitness landscapes for a biological example in Sect. 10.3. As Sorkin noted, hierarchical roughness of a landscape allows fairly modest moves in parameter space to allow the exploration of a significant portion of that space; in more regular landscapes a considerable amount of energy is needed to move from one local minimum to another.

If we think of evolution, or coevolution, at work on fitness landscapes, then we see reasons that evolution favors fractal landscapes. A thoughtful exploration of evolution acting on many levels is given in [27].

10 Results

Here we present some results of applications of the dictionary. We first test dictionary entries against data generated from logistic map systems, giving both a sense of the uniqueness of systems with different parameter values as well as the synchronization of certain systems to produce the dynamics of systems with other combinations of parameter values. We then present the results of matching the dictionary to ECG data and EEG data and offer an interpretation of the findings.

Fig. 12 A log-log plot of Eq. (5)



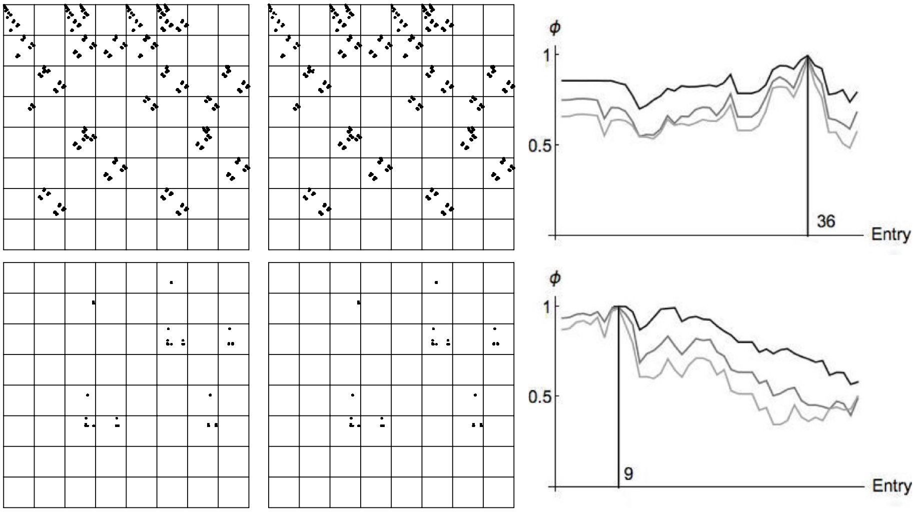


Fig. 13 Driven IFS plots and φ graphs against dictionary entry number. The dark curve corresponds to $L = 2$, the middle to $L = 3$, and the light curve to $L = 4$

10.1 Finding logistic maps with dictionary entries

We found the best matches for a logistic map with the single map entries in the dictionary. For example, 25,000-point time series from logistic maps with $r = 3.93$ (in the dictionary) and $r = 3.6454$ (not in the dictionary) were compared to the 42 entries of the single map portion of the dictionary. In Fig. 13 we see results of single map tests: the driven IFS (equal-size bins) for input (left images) and dictionary best-match (middle images), and plots of the dictionary entry versus $\varphi(IFS_{input}, IFS_{entry}, L)$ for $L = 2, 3, 4, 5$ for two tests of single logistic map data with $r = 3.93$ (top row) and $r = 3.6454$ (bottom row).

In the first example, the dictionary entry for the same value of r as the input logistic data was found (the peak value in the plots of φ). For the second example, the value of $r = 3.6454$, which itself is not an r-value in the dictionary, found the best match in the entry for $r = 3.64$, and the second best match for the entry $r = 3.65$. The values of r immediately surrounding these values had high values of φ as well, shown in Table 1.

For single logistic maps for each value of $r = 3.57 - 4.00$ in steps of 0.01, the dictionary almost always found the best match for $L = 2, 3, 4, 5$ φ at the entry with the corresponding value of r. This was not the case for $r = 3.58$ and $r = 3.84$, both of which found nearby values of r ($r = 3.57$ and $r = 3.85$, respectively). Out of the 42 values of r included in the dictionary, these two which did not find the correct match were near periodic ranges and are a noisy 4-cycle and a 3-cycle, respectively. This further supports our claim that this tool is not sensitive for studying periodic behavior.

Table 1 Highest values of φ for $r = 3.6454$ logistic map data

r	φ_2	φ_3	φ_4	φ_5
3.66	0.999	0.960	0.903	0.872
3.65	0.999	0.989	0.978	0.941
3.64	0.999	0.999	0.999	0.999
3.63	0.999	0.912	0.824	0.780

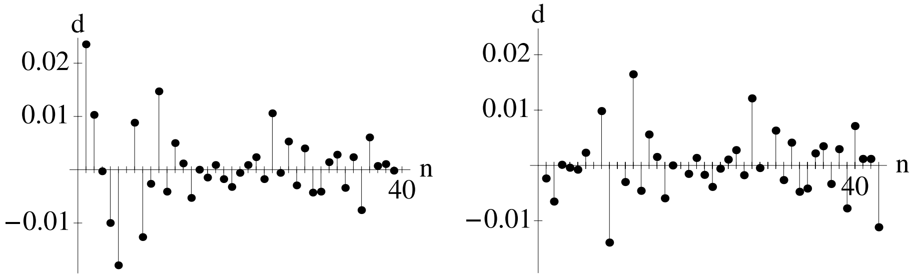


Fig. 14 The difference d between the r values of the target logistic map and of the best dictionary matches n for $L = 4$ (left) and $L = 5$ (right)

For the next test we took each value of $r = 3.57 - 4.00$ in steps of 0.01 contained in the dictionary of single maps and added noise, $\epsilon = \pm 0.005$, to r. The input test data likewise found entries in the dictionary corresponding to the nearest values for r, within ± 0.03 for φ for $L = 4$ and ± 0.02 for φ for $L = 5$. For example, test data produced from $r = 3.876$ had the highest correlations $(\varphi(IFS_{test}, IFS_{entry}, 4) = 0.90)$ with the dictionary entry corresponding to $r = 3.88$. Test data produced from $r = 3.653$ had the highest correlations $(\varphi(IFS_{test}, IFS_{entry}, 5) = 0.988)$ with the $r = 3.64$ entry in the dictionary. As Fig. 14 suggests, the highest φ values of the dictionary entries for single map systems can discriminate for the value of r within a range of ± 0.03 for $L = 4$ and a range of ± 0.02 for $L = 5$.

Next, a system of two coupled logistic maps with $(r_1, r_2, c) = (3.81, 3.62, 0.35)$ was compared to the two-map dictionary entries. Because our dictionary construction limits c values to $0, 0.1, \dots, 0.9, 1.0$, this system of coupled maps is not a dictionary entry. The comparison tool found several good matches; we'll mention one: $(r_1, r_2, c) = (3.73, 3.68, 0.7)$. Comparison of address occupancies of this system with those of the target system gives these fitnesses φ for $L = 2, 3, 4$, and 5:

$$0.995,\ 0.973,\ 0.941,\ \text{and}\ 0.908$$

This is particularly interesting, because while its driven IFS plot and address occupancies, and its return map, are very close to those of the target system, both seen in Fig. 15, its parameters aren't. So in terms of address occupancies and driven IFS and return map plots, or in terms of the fitness function φ, the parameter space is partitioned into subsets of approximately constant value. Whether these regions are smooth submanifolds, fractal subsets, or something else altogether, remains to be determined.

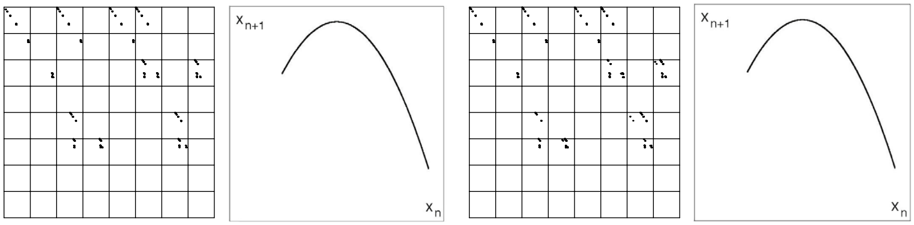


Fig. 15 The driven IFS and return map of the target system (left), the dictionary entry $(r_1, r_2, c) = (3.73, 3.68, 0.5)$ (right)

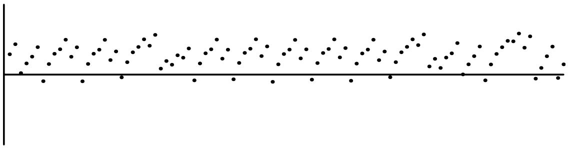


Fig. 16 Differences in individual maps

The return map resembles a portion of a parabola. The maximum height of return map points is 0.92621. The logistic map $L_r(x) = rx(1 - x)$ has maximum height $r/4$, and plotting the return map along with the graph of $L_{3.70484}$, we see that the graphs coincide.

A possible explanation is that the individual logistic maps that make up the network have synchronized. Synchronization of chaotic processes is a familiar occurrence, described clearly in Chapter 7 of [28]. In this case, something more complex happens. In Fig. 16 we see a plot of the differences of the individual logistic maps of the network, for iterates 9,900 through 10,000. The scale of the vertical line is $-0.02 \leq y \leq 0.02$. If the maps synchronize, these differences would converge to 0. Something else has happened here: The differences oscillate, but not quite periodically. Because we've not plotted the first 9,900 points, almost surely any transients have died out long before the plotted points. Yet the parabola appears clean, not fuzzy, so this plot signals a more complex dynamics. Nevertheless, this approximate synchronization provides an explanation for some of the sets in the partition of parameter space.

10.2 An ECG example

The driven IFS algorithm can be applied to any sequence ordered by time. For data we took the 2002 PhysioNet challenge [29] to simulate cardiac data, and to differentiate simulated and physiological data. The data are the *RR* intervals from 50 ECG recordings and simulations. Because *R* is the peak of the *QRS* complex in an ECG, the intervals between *R* peaks are the intervals between heartbeats.

The challenge was to distinguish which of the 50 signals were real cardiac data, and which were simulations. Using the key in [30], we apply the comparison tool to sample *rr*08, real cardiac data. The length of the data set is the number of heartbeats in a day, giving 97,659 *RR* intervals for this sample. From this we generate driven IFS with the first 10,000 points. The left graph of Fig. 17 is the driven IFS of the *rr*08 data with equal-size bins. The plot is mainly a *backward Z*, a consequence of most bin changes being between adjacent bins. For example, only rarely is a very small interbeat interval followed immediately by a very long interval. While this structure is easy to understand,

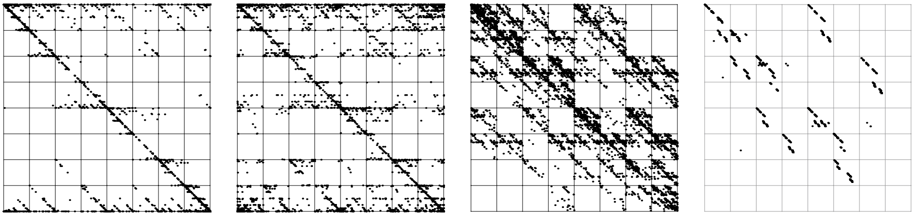


Fig. 17 Driven IFS for the *rr*08 cardiac data with equal-size bins (first) and equal-weight bins (second), the best equal-size dictionary match (third), and the best equal-weight dictionary match (fourth)

it is not particularly useful for detecting a cartoon with a good match to the experimental data. The middle graph of Fig. 17 is the *rr*08 driven IFS with equal-weight bins.

The best equal-size dictionary match for the *rr*08 data is for a 4-map system with $r_1 = 3.69$, $r_2 = 3.81$, $r_3 = 4.0$, and $r_4 = 3.61$, with $c = 0$. For this system the fitness φ is

$$0.735,\ 0.689,\ 0.640,\ 0.608,\ \text{for L} = 2, 3, 4, 5$$

For all 110 best matches, some with 2 maps, some with 3 maps, some with 4 maps, all have $c = 0$.

The best 3-map equal-weight dictionary entry has $r_1 = 3.98$, $r_2 = 3.99$, $r_3 = 3.63$, with $c = 1.0$. For this system the fitness φ is

$$0.669,\ 0.571,\ 0.550,\ 0.534,\ \text{for L} = 2, 3, 4, 5$$

In Fig. 18 we see the log-log plot of the length-2 address fitness of networks of two logistic maps when matched to the ECG data of the *rr*08 sample. This plot is generated using the method presented in Sect. 9. The points suggest a linear fit in log-log coordinates. Linear regression gives a slope of 0.02, but a 90% confidence interval includes a slope of 0, so we cannot deduce a positive slope. We'll get a similar plot in Sect. 10.3 and discuss interpretations there.

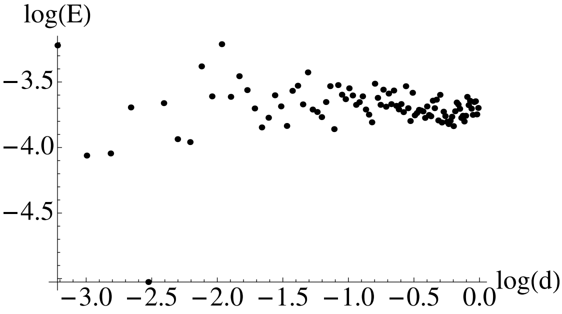
Of course, the network whose driven IFS is pictured in the right image of Fig. 17 is far from a perfect match with the ECG data. It is unrealistic to expect that the dynamics of a system as complex as the heart could be captured completely by a small number of uncoupled logistic maps. On the other hand, at least in terms of bin occupancies, the fitness function calculations suggest that about 70% of the heart's behavior can be described in this way. Subtract out the parts of cardiac dynamics that can be accounted for by the behavior of a few simple oscillators, and then look to the electrophysiology of cardiac ion channels for guidance in the residual behavior. This may be the most obvious use of our dictionary fitness landscape.

Visualization remains tricky. For 4 maps the parameter space is 5-dimensional, so a plot of the fitness landscape lies in 6-dimensional space. A careful analysis of the landscape roughness could direct more effective searches without landscape visualization. This is another way that geometry can aid our understanding of biology.

10.3 An EEG example

The fractal properties of EEG data have been explored by many, notably Accardo [31] and Kulish [32]. As another biological time series, we took open-source EEG data from

Fig. 18 Log-log plot for ECG data



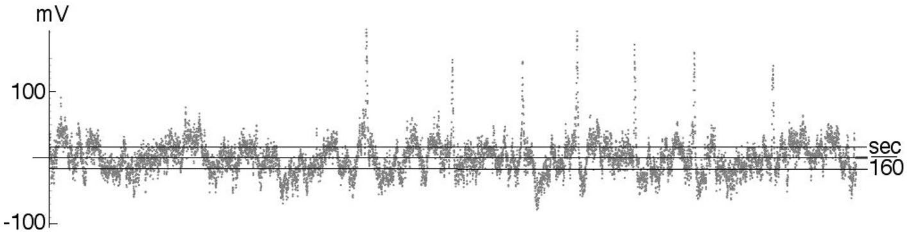


Fig. 19 An EEG time series with equal-weight bin boundaries

Schalk [33] through PhysioNet. They supplied the EEGs for 109 volunteers performing several tasks, recorded using the BCI2000 system with the standard 64 electrodes on an international 10-10 system sampling at 160 samples per second. We took the baseline (eyes open, no motor or visual tasks) time series for the S004 subject's 5th electrode, which had 9760 points, corresponding to about 1 minute of data. The time series with equal-weight bin boundaries is shown in Fig. 19.

The left image of Fig. 20 shows the driven IFS for the EEG data of Fig. 19. The middle image is the best equal-weight bins dictionary match among 2-map systems. The system parameters are $r_1 = 3.94$, $r_2 = 3.43$, and $c = 0$. The fitnesses φ are

$$0.5800,\ 0.5911,\ 0.5525,\ 0.5236 \text{ for } L = 2, 3, 4, 5.$$

The right image of Fig. 20 shows a good dictionary match for a 4-map system with $r_1 = 3.88$, $r_2 = 3.9$, $r_3 = 3.65$, $r_4 = 3.96$, and $c = 0.1$. The fitnesses φ are

$$0.6958,\ 0.5339,\ 0.5142,\ 0.4980 \text{ for } L = 2, 3, 4, 5.$$

Both these dictionary matches do a decent job of reproducing some of the distribution of points on the diagonal, as well as some of the off-diagonal clusters. The 4-map network also reproduces some of the sub- and super-diagonals, lines parallel to the diagonal. As in Sect. 10.2, we see that our simple networks reproduce some, but not all, of the interesting dynamics of this EEG signal. We'll discuss the implications in Sect. 12.

In Fig. 21 we see the log-log plot to assess landscape roughness. The plot is quite similar to that of the ECG data. How are we to understand this slope which appears close to 0? For example, in calculations of mass dimension, the relation $m(r) = kr^d$ between the mass contained within a sphere and the radius of that sphere implies that a log-log plot gives

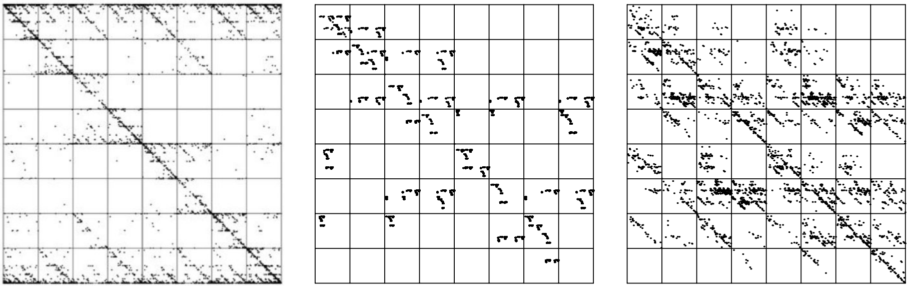


Fig. 20 Left: the IFS driven by the EEG data of Fig. 19. Center: a 2-map match. Right: a 4-map match

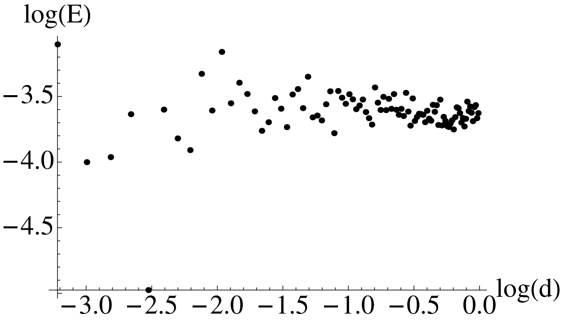


Fig. 21 Log-log plot for EEG data

points on a line of slope *d*. Because a larger sphere contains more mass, the slope *d* is positive. In our landscape, we measure how the average of the square of the difference of fitnesses scales with the distance between points in parameter space. Linear regression gives a slope of 0.02, but a 90% confidence interval includes a slope of 0 so all we can safely say is that eventually the average squared difference seems to stay approximately constant. This effect is not monotone: local increases and decreases are interwoven throughout the range of scales we have investigated. This signals a rough landscape, though perhaps not fractal in Sorkin's sense.

Comparison of Figs. 18 and 21 show they are very similar. This means only that these two fitness landscapes have similar roughnesses. This does not imply that the landscapes are nearly identical, but it does suggest that search strategies such as simulated annealing would be comparably effective at finding best matches for both surfaces.

One final observation about this graph. The plots in Figs. 18 and 21 were sampled on a grid with $\Delta r = \Delta c = 0.005$. The dark points of Fig. 22 also are sampled on this grid. The lighter points are sampled with $\Delta r = \Delta c = 0.0025$. Note that at least for these two examples the width of the point scatter increases as the size of the sample grid decreases. This suggests that the more closely we look, the wider the variation of the average fitnesses. The complexity of this landscape is consistent with the intricate interweaving of periodic and chaotic behavior displayed in the familiar logistic map bifurcation diagram.

11 Future directions

In this section we'll outline some directions in which the dictionary can be refined.

The simplest approach to categorize driven IFS images is to note whether a given address is empty. Because IFS are driven by time series of finite length, we need proper

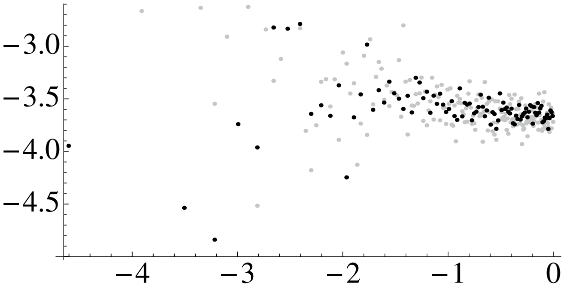


Fig. 22 Dark points are sampled on a coarser grid, light points on a finer grid

statistics to determine if an address is empty due to an exclusion in the data rather than too little data. One approach is presented in Appendix A.92 of [34].

Building the dictionary around average address occupancies for each network is predicated on the unicity of the attractor of each network. As we have seen in Sect. 6, this is not always the case. For systems exhibiting multiple attractors, the average occupancies of these attractors will not give values that accurately represent any attractor. Our sampling of initial value space suggests that multiple attractors are uncommon, but they do occur. One approach is to compare the address occupancies of each new sample with the running averages. If the differences fall outside a set range, the new sample will start a new dictionary entry with the same network parameters.

Intermittency and regime change present related, but more subtle, problems because the time series switch again and again between attractors. The Lyapunov exponent (Sect. 2.2 of [21] or Appendix A.94 of [34]) may allow automated dictionary generation to separate a time series into different regimes. The Lyapunov exponent, which measures the average rate of divergence of nearby iterates, can be estimated by $\ln|df/dz|$ averaged along a time series. On the left side of Fig. 23, we see the time series for the average of two logistic maps, with $r_1 = 3.93$, $r_2 = 3.94$, and $c = 0.1$. The dynamics are uniformly chaotic, and the running average of the Lyapunov exponent calculation appears to converge nicely, even when the vertical scale is magnified in the plot below the top graph. The middle ($r_1 = 3.83$, $r_2 = 3.51$, $c = 0.1001$) shows an intermittent time series, the right ($r_1 = 3.83$, $r_2 = 3.75$, $c = 0.14816$) exhibits switching between multiple attractors. The magnification of these Lyapunov sequences shows more complex behavior with little evidence of convergence. In particular, note that regime changes appear to correlate with changes in the Lyapunov plots, suggesting an approach to testing for multiple regime attractors.

The top right graph of Fig. 23 may cause some confusion. Two regions of the time series appear to be 2-cycles, yet the Lyapunov exponents are positive. Because periodic attractors are signaled by negative Lyapunov exponents, the time series and Lyapunov graphs may seem inconsistent. If we magnify the vertical scale of the apparent 2-cycle, we see the distance between the upper and lower points increases with time. This growing separation is the source of the positive Lyapunov exponent.

Additionally, with estimates of transformation probabilities by address occupancies, $f(\alpha)$ curves can be calculated using the method described in Sect. 17.3 of [35] or Sect. 7.3 of [34]. Comparison of $f(\alpha)$ curves may give a way to sieve the dictionary during the comparison with a given system.

Furthermore, the thickness of the return maps can be used as a measure of synchronization, and the embedding dimension as a measure of the amount of memory in the network.

The complexity of the fitness landscapes generated by comparing experimental data to the whole dictionary suggests an alternate approach: abandon the current static dictionary and replace it with an adaptive dictionary. For example, find the closest match from a

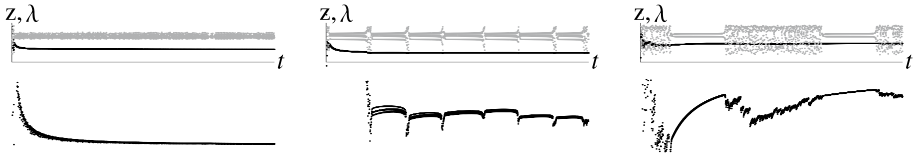


Fig. 23 Top: time series (gray) and Lyapunov exponent (black) computation. Bottom: magnification of the vertical scale for the Lyapunov calculations. Left to right: $0.32 \leq \lambda \leq 0.40$, $0.225 \leq \lambda \leq 0.25$, and $0.43 \leq \lambda \leq 0.49$

handful of sample systems randomly scattered through parameter space. Then build dictionary entries near this first match. Select the best and sample the parameter space on increasingly finer grids, exploiting hierarchical distributions in parameter space.

Alternately, network features could be explored sequentially: ascertain the coupling topology first, then the number of maps, the strengths of the couplings, and finally narrow down possible values of the logistic coefficients, r.

We know that systems with different parameters can produce remarkably similar behaviors. For example, two coupled $r = 4$ logistic maps with $0.3 \leq c \leq 0.7$ synchronize to generate the dynamics of a single $r = 4$ logistic map. While this may seem to be a complication, if these commonalities can be mapped, even roughly, then the search strategy can be sharpened to include only one representative from each region.

12 Conclusion

Robert May [19] and others have shown that complex dynamics may result from simple nonlinear systems. But most biological systems are not simple. Nevertheless, some aspects of the behavior of complex systems may result from the coherent action of groups of agents, perhaps not organized in a biologically obvious way. Comparison of time series data with a robust dictionary of nonlinear systems leads to the construction of a fitness landscape. Understanding the roughness of this landscape can guide efficient searches. Simulated annealing, parameter space random walks of gradually decreasing step size, is a promising approach. The identification of the closest matches may provide some insight into hidden dynamical processes. Our experiments are a first step in this direction.

Acknowledgements This paper is an elaboration of the work in the first author's Applied Mathematics Yale senior thesis [36]. In early stages of this project we benefitted from many thoughtful and energetic discussions with Rachel Lawrence and Hannah Otis. Comments by a very thoughtful anonymous referee led to improvements in the readability of this paper.

Funding This research was supported by grant funding for the first author from the Science and Technology Research Scholars (STARS) II program at Yale University.

Declarations

Conflicts of interest We have no conflict of interest.

References

1. Wright, S.: Evolution in Mendelian populations. Genetics 16, 97–159 (1931)
2. Wright, S.: The role of mutation, inbreeding, crossbreeding, and selection in evolution. Proc. Sixth Int. Cong. Genetics **1**, 356–366 (1932)
3. Nowak, M.: Evolutionary Dynamics: Exploring the Equations of Life. Harvard University Press (2006)
4. Waddington, C.: Principles of Development and Differentiation. Macmillian (1966)
5. Waddington, C.: Strategy of the Genes. A Discussion of Some Aspects of Theoretical Biology. George Allen & Unwin (1957)
6. Thom, R.: Structural Stability and Morphogenesis. An Outline of a General Theory of Models. Benjamin (1975)
7. Frauenfelder, H., Sligar, S., Wolynes, P.: The energy landscape and motions of proteins. Science **254**, 1589–1603 (1991)

8. Frauenfelder, H., et al.: A unified model of protein dynamics. Proc. Nat. Acad. Sci. U.S.A. **106**, 5129–5134 (2009)
9. Frauenfelder, H., Leeson, D.: The energy landscape in non-biological and biological molecules. Nat. Struct. Biol. **5**, 757–759 (1998)
10. Frauenfelder, H.: Energy landscape and dynamics of biomolecules. J. Biological Physics **31**, 413–416 (2005)
11. Jeffrey, H.: Chaos game representation of gene structure. Nucl. Acid Res. **18**, 2163–2170 (1990)
12. Jeffrey, H.: Chaos game visualization of sequences. Comput. Graph. **16**, 25–33 (1992)
13. Lind, D., Marcus, B.: An Introduction to Symbolic Dynamics and Coding. Cambridge University Press (1995)
14. Kitchens, B.: Symbolic Dynamics: One-Sided, Two-Sided, and Countable State Markov Shifts. Springer (1998)
15. Blanchard, F., Maass. A., Nogueira, A.: Topics in Symbolic Dynamics and Applications. Cambridge University Press (2000)
16. Hutchinson, J.: Fractals and self-similarity. Indiana Univ. J. Math. **30**, 713–747 (1981)
17. Barnsley, M.: Fractals Everywhere. 2nd ed., Academic Press (1993)
18. Stewart, I.: Order within the Chaos Game? Dynamics Newsletter **3**, 4–9 (1989)
19. May, R.: Simple mathematical models with very complicated dynamics. Nature **261**, 459–467 (1976)
20. Devaney, R.: An Introduction to Chaotic Dynamical Systems. 2nd ed., Addison-Wesley (1989)
21. Rasband, N.: Chaotic Dynamics of Nonlinear Systems. Wiley-Interscience (1990)
22. Hindmarsh, J., Rose, M.: A model of neuronal bursting using using three coupled first order differential equations. Proc. Roy. Soc. London. Biol. Sci. **221**, 87–102 (1984)
23. Storace, M., Linaro, D., de Lange, E.: The Hindmarsh-Rose neuron model: bifurcation analysis and piecewise-linear approximations. Chaos **18**, 033128 (2008)
24. Sorkin, G.: Efficient simulated annealing on fractal energy landscapes. Algorithmica **6**, 367–418 (1991)
25. Adami, C.: Introduction to Artificial Life. Springer (1997)
26. Weinberger, E., Stadler, P.: Why some fitness landscapes are fractal. J. Theoret. Biol. **163**, 255–275 (1993)
27. Bahar, S.: The Essential Tension: Competition, Cooperation and Multilevel Selection in Evolution. Springer Nature (2018)
28. Strogatz, S.: Sync: The Emerging Science of Spontaneous Order. Hyperion Books (2003)
29. http://physionet.org/challenge/2002/event-2.shtml#downloading-the-challenge-dataset Accessed 10 Feb 2013 (2013)
30. Wessel, N., et al.: Classifying simulated and physiological heart rate variability signals. Computers in Cardiology **29**, 133–135 (2002)
31. Accardo, A., et al.: Use of the fractal dimension for the analysis of electroencephalographic time series. Biol. Cybernetics **77**, 339–350 (1997)
32. Kulish, V., Sourin, A., Sourina, O.: Human encephalograms seen as fractal time series: Mathematical analysis and visualization. Comput. Biol. Med. **36**, 291–302 (2006)
33. Schalk, G., et al.: BCI2000: A general-purpose brain-computer interface (BCI) system. IEEE Trans. Biomed. Eng. **51**, 1034–1043 (2004)
34. Frame, M., Urry, A.: Fractal Worlds: Grown, Built, and Imagined. Yale University Press (2016)
35. Falconer, K.: Fractal Geometry. Mathematical Foundations and Applications. 3rd ed., Wiley (2014)
36. Thew, N.: A Dictionary of Driven Iterated Function Systems to Characterize Chaotic Time Series. Yale University Applied Mathematics Senior Thesis (2014)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.