Examining k-nearest neighbour networks: Superfamily phenomena and inversion

Published in:
Chaos

DOI:
10.1063/1.4945008

Document Version
Publisher's PDF, also known as Version of record

Link to publication in the UWA Research Repository

Rights statement
This article may be downloaded for personal use only. Any other use requires prior permission of the author and AIP Publishing.

General rights
Copyright owners retain the copyright for their material stored in the UWA Research Repository. The University grants no end-user rights beyond those which are provided by the Australian Copyright Act 1968. Users may make use of the material in the Repository providing due attribution is given and the use is in accordance with the Copyright Act 1968.

Take down policy
If you believe this document infringes copyright, raise a complaint by contacting repository-lib@uwa.edu.au. The document will be immediately withdrawn from public access while the complaint is being investigated.
Examining k-nearest neighbour networks: Superfamily phenomena and inversion

Alexander Khor* and Michael Small

Citation: Chaos 26, 043101 (2016); doi: 10.1063/1.4945008
View online: http://dx.doi.org/10.1063/1.4945008
View Table of Contents: http://aip.scitation.org/toc/cha/26/4
Published by the American Institute of Physics

Articles you may be interested in

Experimental demonstration of revival of oscillations from death in coupled nonlinear oscillators
Chaos 26, 043112 (2016); 10.1063/1.4947081

Suppression of phase synchronisation in network based on cat's brain
Chaos 26, 043107 (2016); 10.1063/1.4945796

Chaos control of Hastings–Powell model by combining chaotic motions
Chaos 26, 043106 (2016); 10.1063/1.4946811

Epidemic spreading with activity-driven awareness diffusion on multiplex network
Chaos 26, 043110 (2016); 10.1063/1.4947420

Synchronization criteria for generalized reaction-diffusion neural networks via periodically intermittent control
Chaos 26, 043113 (2016); 10.1063/1.4947288

Successful network inference from time-series data using mutual information rate
Chaos 26, 043102 (2016); 10.1063/1.4945420
Examining $k$-nearest neighbour networks: Superfamily phenomena and inversion

Alexander Khor$^{1,a)}$ and Michael Small$^{1,2}$

1School of Mathematics and Statistics, The University of Western Australia, Crawley, Western Australia 6009, Australia
2Mineral Resources, CSIRO, Kensington, Western Australia, Australia

(Received 4 February 2016; accepted 15 March 2016; published online 4 April 2016)

We examine the use of recurrence networks in studying non-linear deterministic dynamical systems. Specifically, we focus on the case of $k$-nearest neighbour networks, which have already been shown to contain meaningful (and more importantly, easily accessible) information about dynamics. Superfamily phenomena have previously been identified, although a complete explanation for its appearance was not provided. Local dimension of the attractor is presented as one possible determinant, discussing the ability of specific motifs to be embedded in various dimensions. In turn, the Lyapunov spectrum provides the link between attractor dimension and dynamics required. We also prove invertibility of $k$-nearest neighbour networks. A new metric is provided, under which the $k$-nearest neighbour and $\epsilon$-recurrence construction methods produce identical networks. Hence, the already established $\epsilon$-recurrence inversion algorithm applies equally to the $k$-nearest neighbour case, and inversion is proved. The change in metric necessarily distorts the shape of the reconstructed attractor, although topology is conserved. © 2016 AIP Publishing LLC.

[http://dx.doi.org/10.1063/1.4945008]

In recent times, complex network based techniques have proved valuable in the analysis of non-linear systems. In the case of time series measurements obtained from deterministic dynamical systems, recurrence networks may be employed. Recurrence networks utilise time series elements as nodes of the network, with edges constructed using one of a handful of methods. One of these methods produces so called $k$-nearest neighbour networks, which construct edges according to spatial proximity of state vectors. In this paper, two open problems surrounding $k$-nearest neighbours are addressed. First, an explanation is given for previously identified superfamily phenomena, in which systems exhibiting similar dynamics produce related networks. Second, we prove invertibility of $k$-nearest neighbour, allowing the original time series to be recovered from the network.

I. INTRODUCTION

The use of complex networks has repeatedly been shown to be a promising tool in the analysis of non-linear deterministic dynamical systems, specifically time series measurements of such systems. One such class of complex networks, recurrence networks, has been used to successfully reveal important dynamical information through inspection of network properties.\(^1\) Recurrence networks employ state vectors as vertices of the network, whilst a scalar time series offers observations in only one component of typically multidimensional state space. Takens\(^2\) showed that scalar time series observations are indeed sufficient to reconstruct the full dynamics of a higher dimensional, non-linear system, namely, through a suitable delay-coordinate embedding. If \(\{x(t)\}_{t=1}^{N}\) is a scalar time series, then a delay-coordinate embedding is a new vector time series \(\{v(t)\}\) such that \(v(i) = (x(i), x(i - \tau), x(i - 2\tau), \ldots, x(i - (m - 1)\tau))\). Here, the free parameters \(\tau\) and \(m\) are called the embedding lag and the embedding dimension, respectively; much work has been done in finding methods for appropriately choosing these parameters.\(^3\)

The history of recurrence networks begins with the use of recurrence plots (RPs), first introduced by Eckmann, Kamphorst, and Ruelle.\(^4\) These RPs captured information about the proximity of state vectors on the attractor of a dynamical system. For a vector time series given by \(\{v(t)\}_{t=1}^{N}\), its RP is given in \(\mathbb{R}^2\) by

\[
R(i,j) = \begin{cases} 
1 & \|v(i) - v(j)\| < \epsilon \\
0 & \text{otherwise,}
\end{cases}
\]

where \(\epsilon\) is some chosen threshold parameter. RPs were originally analysed by looking at visual properties, such as off diagonals, which represented successively close states.

The progression from recurrence plots to complex networks, whilst seemingly trivial and obvious in hindsight, unlocked an entirely new level of insight into dynamical system analysis. By viewing the RP not as a visualisation in \(\mathbb{R}^2\), but instead as the adjacency matrix of a network, the entire toolkit of network theory is now afforded. Furthermore, a number of methods for producing recurrence networks have since been presented. Two common types, and indeed those which we will focus on in this paper, are \(\epsilon\)-recurrence and $k$-nearest neighbour networks.

When studying these networks, Xu, Zhang, and Small\(^5\) presented one less intuitive result by identifying superfamily

\(^a\)Electronic mail: alexander.khor@uwa.edu.au
phenomena existent in $k$-nearest neighbour recurrence networks. Superfamily phenomena concern the relative prevalence of motifs within $k$-nearest neighbour networks. Whilst Xu et al. showed empirically that superfamily phenomena reflected underlying dynamics, they did not provide any particular theory as to why this was the case. Liu and Zhou\textsuperscript{6} aimed to address this by comparing superfamily phenomena found in networks based on dynamical systems to those built from fractional Brownian motions and multifractal random walks. It was found that motif rank ordering was correlated to the detrended fluctuation analysis (DFA) scaling exponent of the original time series.

Section III provides an alternate explanation of superfamily phenomena by examining the dependence of attractor dimension on motif prevalence. We identify that for recurrence networks constructed using the $k$-nearest neighbour algorithm the relative prevalence of motifs are strongly dependent on the local dimension of the space from which the state vectors are taken. Since recurrence networks inherently capture the proximity of states, motifs represent specific arrangements of states in space, some of which are more or less likely to occur as dimension changes. Heuristic evidence is given by identifying comparable superfamily phenomena in networks constructed from states randomly distributed in spaces of varying dimensions.

We believe this explanation is superior in that it provides a direct link between dynamics and superfamily phenomena, as opposed to introducing an intermediary parameter as is the case with the detrended fluctuation analysis scaling exponent. Explanation of superfamily phenomena provides important validity to the use of these techniques in studying deterministic, non-linear dynamical systems. Similarly, through understanding the underlying mechanisms, we are also able to more confidently assess where the use of superfamily phenomena based analysis is not appropriate.

Section IV addresses inversion of $k$-nearest neighbour networks. Specifically, this is achieved by extending the inversion algorithm provided by Hirata et al.\textsuperscript{7} for the case of $\epsilon$-recurrence networks. We show $k$-nearest neighbour and $\epsilon$ recurrence networks can be viewed as identical structures under a change of (equivalent) metrics, and thus that $k$-nearest neighbour networks can be equivalently inverted. By providing an inversion of such networks, we show all information about the dynamics of the system is contained in the network, or rather that no information is lost in the network transformation process. This discovery further validates the use of complex networks as a valid means of studying dynamical systems, whilst also revealing an equivalence between $\epsilon$-recurrence and $k$-nearest neighbour classes of complex networks.

A final summary is given in Section V, and suggestions are given to possible extensions of the work presented.

II. BACKGROUND

A. Recurrence networks

A complex network is an ordered pair $G(V, E)$ made up of a set of vertices $V$ connected by edges $E \subset V \times V$. Complex networks are mathematically equivalent to the idea of a graph, yet a distinction is made due to their varying properties and uses, often related to their origins. Complex networks are often constructed from large data sets, made up of thousands of vertices connected in ways that produce non-trivial topologies.

Whilst there are many techniques for constructing complex networks from time series, the simplest is arguably $\epsilon$-recurrence. Similarly to recurrence plots, we assign each element of our vector time series $v(i)$ a node $i \in V$ and connect them by $(i, j) \in E$ if $||v(i) - v(j)|| < \epsilon$. In this case, we take $\epsilon$ to be a chosen parameter fixed for all $i$. Clearly, the choice of $\epsilon$ is non-trivial, with too small $\epsilon$ producing an inadequately connected structure and too large over-saturating the network with connections. Usually, $\epsilon$ is chosen such that all nodes are at least indirectly connected and the average vertex degree, proportional to $|E|/|V|$, is greater than some chosen value.

We can address this arbitrary choice by replacing the fixed threshold distance $\epsilon$ with a fixed number of connections from each node. In doing this, we produce so called $k$-nearest neighbour networks. Just as in the $\epsilon$-recurrence case, we assign each state $v(i)$ a unique vertex $i \in V$. For each $v(i)$, we identify its $k$ nearest neighbours $v(i_1'), v(i_2'), \ldots, v(i_k')$, and construct unweighted, undirected edges $(i, i_p')$. An exception is made in the case where two states $v(i), v(j)$ are mutual nearest neighbours, i.e., $v(j) = v(i_q')$ and $v(i) = v(j_p')$ for some $p, q \leq k$. We construct the edge $(i, j)$ and also make the next nearest connection of either $(i, i_p'_{p+1})$ or $(j, j_p'_{p+1})$. In doing this, the average degree of each node remains exactly $2k$ (the two arises from the fact each undirected edge is adjacent to two nodes), although some will certainly have a higher degree than others.

Whilst the use of $k$-nearest neighbour networks appears only to have replaced the free choice of $\epsilon$ with that of $k$, the change is non-trivial. Choosing the parameter $k$ is consistently easier, for instance, $k = 4$ has proven to be satisfactory for most systems, whilst $\epsilon$ must be reevaluated in each specific case depending on scale. Yet, a new problem arises for $k$-nearest neighbour networks in the forming of a bias for neighbours to be recent predecessors or successors. These temporal neighbours will naturally be close in space to the current state, particularly for high sampling rates. To counter this issue, we introduce a new parameter $T$ and exclude edges $(i, j)$ where $|i - j| < T$. A common choice for $T$ is one quarter the recurrence time, that is, the average time for a trajectory to return to the same region in state space.

B. Superfamily phenomena

A number of techniques for the analysis of recurrence networks have emerged in the literature. Most involve implementations of network metrics that precede the application of networks in studying dynamical systems. One alternate tool that has arisen in recent years examines network motif prevalence.

Motifs are small connected graphs of fixed order; in the context of complex networks, we are interested in their appearance as subgraphs of the larger network. In the case of recurrence networks constructed from dynamical systems,
these motifs have been shown to occur with distinct relative likelihood for different classes of dynamics. We call these behaviour superfamily phenomena. Xu, Zhang, and Small first studied superfamily phenomena in the case of order four motifs within k-nearest neighbour networks constructed from well known dynamical systems. The six possible nonisomorphic motifs of size four are shown in Figure 1. The use of order four motifs prove robust in their ability to be identified with relative computational ease (as opposed to the twenty one possible motifs of order five) yet with sufficiently many options to be interesting (there are only two possible motifs of size three).

After counting occurrences of the six possible motifs within networks constructed from a number of well known dynamical systems, a consistent motif rank ordering was found to be associated with different classes of underlying dynamics. Periodic flow closely followed the rank ordering ABCFDE, or ADBCEF with the addition of noise, whilst chaotic flow tended to follow ABCDFE. Non-linear maps were also shown to exhibit these superfamilies, with distinctions found between chaotic, hyper-chaotic, and noisy dynamics.

C. Network inversion

Recurrence networks are an intuitive and easily obtainable way of visualising often complicated dynamics of higher dimensions. As such, an important question arises regarding their limitations. Namely, whether the simplicity of the network has come at the cost of any important information about the underlying system. Clearly, some information is lost regarding precise distances in space, as well as the time ordering of states. Yet, it is claimed the overall topology of the attractor is preserved through network transformation. One way to prove no information is lost is to find an inversion from the network back into state space. Hirata et al. achieved this in the case of c-recurrence networks. Using an inversion algorithm, and by preserving the time indices of each node, they were able to reconstruct the full time series up to topological equivalence.

Hirata et al. present the following algorithm to invert the network transformation:

1. Construct an c-recurrence network as described above for some non-linear vector time series of state vectors.
2. For each node \( i \in V \), define its neighbourhood \( G_i = \{ j \in V : (i,j) \in E \} \). Now assign a weight to each edge by

\[
 w(i,j) = 1 - \frac{|G_i \cap G_j|}{|G_i \cup G_j|} .
\]

3. We now have a new network with edges identical to our original c-recurrence network, but now weighted by \( w(i,j) \).

Hence, we calculate the least distance between any two nodes \( k \) and \( l \) and label it \( d(k,l) \). Many algorithms have been presented to efficiently calculate this quantity, for instance, the Dijkstra method or the Johnson’s algorithm.

4. Finally, we use a method for multidimensional scaling to embed the nodes back in space such that their distances are the same as that given by the distance matrix just calculated. One easily accessible method is that of classical multidimensional scaling. As is necessary to faithfully preserve the distance matrix, the multidimensional scaling program will often embed the state vectors in an excessively large dimension. For the purposes of visualisation, we choose the first three components, that is those with the largest eigenvalues.

Before moving any further, it is important to note two limitations of this algorithm. First, we can never regain exact locations in space for states. This should be clear from the fact that our network only captures information about relative distances, and not their actual locations. In practice, this is rarely an issue, particularly considering the possible use of Takens’ embedding theorem may have already left us with state vectors in a space equivalent, but not identical, to our underlying system. Second, the time ordering is completely lost in network construction, hence we must “hold on to” the time index of each node if we wish to fully reconstruct the original time series, and hence examine dynamics.

Hirata et al. went on to prove that the above algorithm does indeed reconstruct the original time series up to topological equivalence under mild conditions. They did so by showing the Euclidean metric on the original time series is equivalent to the network metric \( d(k,l) \).

III. SUPERFAMILY PHENOMENA

Here, we continue on the work done by Xu et al., seeking to explain why such superfamily phenomena occur. Whilst Liu and Zhou have used DFA to nominate the DFA scaling exponent of the underlying time series as a possible determinant of superfamily phenomena, we will instead focus on the local dimension of the attractor as another, more intuitive explanation.

We can relate the dimension of the attractor to the dynamics of the system through its Lyapunov spectrum. Frederickson et al. presented the Kaplan-Yorke conjecture which directly evaluates fractal dimension from the Lyapunov spectrum via the formula

\[
 D_{KY} = j + \frac{\sum_{i=1}^{j} \lambda_i}{|\lambda_{j+1}|},
\]

where \( \lambda_i \) is the \( i \)-th largest Lyapunov exponent, and \( j \) is the largest integer for which

\[
 \sum_{i=1}^{j} \lambda_i \geq 0 .
\]

We see the Kaplan-Yorke dimension of the attractor is given by the summation of two terms. The \( j \) term is an integer part...
which counts the number of Lyapunov exponents which sum to a value greater than zero. The second terms give the fractional part which will always be less than one by definition of \( j \).

Note that for periodic flows \( \lambda_1 = 0 \). Hence, it is easy to verify that \( D_{KY} = 1 \), consistent with an interpretation of a limit cycle being a one dimensional loop in space. On the other hand, for chaotic flows, we can say that \( j \geq 2 \), since there always be one positive and one zero Lyapunov exponent. Indeed, in the case of the chaotic Lorenz and Rössler systems, we have \( j = 2 \) and Kaplan-Yorke dimensions only marginally larger than that.

Since we are working with time series measurements of our dynamical systems, the Lyapunov spectrum is often difficult to obtain, and hence the Kaplan-Yorke proves a suboptimal means of calculating fractional dimension. For computational ease, we will henceforth be using the correlation dimension as our primary means of measuring fractal dimension. We introduce the Kaplan-Yorke dimension only to reveal the important link between the Lyapunov spectrum (and hence the classification of the dynamics) and the attractor dimension. The correlation and Kaplan-Yorke dimensions are consistently very similar, certainly close enough for our impending analysis, and hence it is reasonable to consider the correlation dimension as being inherently linked to the Lyapunov spectrum as well.

To calculate the correlation dimension, we begin by defining the correlation sum of a sequence of state vectors \( \{v(i)\}_{i=1}^N \)

\[
C(\epsilon, N) = \frac{1}{N(N-1)} \sum_{i \neq j} \Theta(\epsilon - \|v(i) - v(j)\|),
\]

where \( \Theta(x) \) is the Heaviside step function

\[
\Theta(x) = \begin{cases} 
1 & x \geq 0 \\
0 & x < 0.
\end{cases}
\]

The correlation sum counts the number of distinct states within \( \epsilon \) distance of each other. It has previously been shown\(^{14}\) that as the number of states approaches infinity and \( \epsilon \) is small, the correlation sum will scale like

\[
C(\epsilon, N) \sim \epsilon^{d_c},
\]

where \( d_c \) is the correlation dimension we are seeking. Thus, we may evaluate it using

\[
d_c = \lim_{\epsilon \to 0} \lim_{N \to \infty} \frac{\log C(\epsilon, N)}{\log \epsilon}.
\]

The correlation dimension has the benefit of being relatively easy to obtain from the vector time series and, more importantly, can be interpreted in terms of local, not just global properties. By summing over all individual state vectors, the correlation sum essentially averages the local dimension at all points on the attractor. This is an intuitive understanding of fractal dimension, where non integer dimensions may be interpreted as the averaging of the dimension of different regions, all of which are locally integer valued.

As a motivating example, let us consider the chaotic Lorenz system in three dimensions. The correlation dimension of the Lorenz attractor has previously been calculated to be \( 2.05 \pm 0.01 \).\(^{14}\) It is possible to interpret this fractal dimension as an averaging of the two dimensional regions of the wings with their three dimensional intersection, producing an overall correlation dimension slightly greater than two (Fig. 2).

A. Motifs

In order to justify the role of attractor dimension in determining superfamily phenomena, we must consider how the prevalence of the different motifs is related to dimension of space. Recall that \( k \)-nearest neighbour networks are constructed by finding nearest neighbours in space and are first and foremost representations of the spacial proximity of states. More precisely, we can regard edges as describing states that are “near,” whilst the absence of a connecting edge indicates two states are “far.” Of course, as we are considering the nearest neighbours of each state, our concepts of near and far must be thought of flexibly, although these ideas prove robust in the context of adequately large networks.

Since motifs are found as subgraphs of the larger network, we can consider them as describing the local distribution of states in space. It is this fact that we shall exploit in our explanation of superfamily phenomena in \( k \)-nearest neighbour networks. As shown by Xu \textit{et al.}, we consistently see some motifs occurring much more frequently than others, suggesting that some arrangements of states are much more favourable. This is indeed a very reasonable claim. Consider motifs A and E which consistently appear with high and low frequency, respectively, for all systems. In Figure 3, we see that motif A naturally corresponds to (approximately) collinear states, something to be expected for any flow or map. On the other hand, for motif E, in order to preserve distance between diagonal nodes of the motif, we require the states to be embedded in an (approximately) square arrangement, a situation we can expect to be occurring considerably less frequently.

When imagining how these network motifs would appear in state space, it is useful to draw upon the pre-existing literature surrounding graph visualisation. Fruchterman and Reingold\(^{15}\) present one particularly powerful technique. We model each of the nodes of the network as a positive electric charge and the edges as springs, allowing the system to reach equilibrium. Since all nodes are like charged they will be repelled, yet this is balanced by the springs which will attract those that are connected. We suggest this produces exactly

![FIG. 2. Chaotic Lorenz attractor shown from “front on” and in “profile,” demonstrating the intersection of the two wings.](image-url)
the behaviour we seek, as connected states are pulled close in space, whilst those unconnected will be pushed apart.

Yet, as observed by both Xu et al. and Liu and Zhou, it is motifs D and F that truly characterise the separate superfamilies. That is, if we were to disregard subgraphs of type D or F, all the example systems would exhibit the same motif rank ordering, namely, ABCE. Hence, we will focus our studies on these two motifs and their dependence on attractor dimension. The first important point to make is that motifs D and F correspond to locally sparse and dense packing of states, respectively. Consider their embedding in $\mathbb{R}^2$ (i.e., on the page), as seen in Figure 3. Seeking to ensure connected states are close in space and disconnected states are far, we find ourselves forced to embed motif D in a spread out arrangement, whilst motif F may remain compact.

Recognising that motifs D and F correspond to sparse and compact embeddings of states in space, we can relate their prevalence to different dimensions of space. The easiest way to begin to understand this relationship is with the immediate observation that motif D is in fact impossible to embed in $\mathbb{R}$. In Figure 4, we see the states labelled $\alpha$, $\beta$, and $\gamma$ can occur successively, yet the placement of $\delta$ becomes problematic. Since $\delta$ is directly connected to $\beta$ but neither $\alpha$ nor $\gamma$, we have reached an impossibility. If it is placed to the left of $\beta$, we would expect it to be a near neighbour of $\alpha$ which we know cannot occur by definition. The same holds for $\delta$ greater than $\beta$, and hence we reach a contradiction. As such, we have shown motif D will never occur in networks produced from time series of dimension one.

We can extend the same argument to time series of higher dimension, but sampled from one dimensional attractors. Clearly, this is not what we see in the empirical data for periodic orbits, which we earlier claimed correspond to one dimensional attractors. Motif D occurrence is indeed nonzero for the periodic attractors. This can be accounted for by two main features of the larger network. First, the $k$-nearest neighbour construction algorithm required a fixed temporal separation between directly connected states, which may negate our above argument. Second, our networks are constructed from finite time series, with finite separations between state vectors. Thus, whilst the attractors may be locally one dimensional, given a sufficiently large curvature of the limit cycle, we may see evidence of near neighbours occurring across the limit cycle as opposed to along it. This is particularly obvious in the case of the higher period Rössler attractors in which state vectors on the sparsely populated stable trajectories are reaching across to a different trajectory when searching for near neighbours. Nonetheless, we do see a considerably reduced number of motif D occurring for periodic attractors, which is consistent with our above analysis.

On the other hand, higher dimensions allow for a greater average separation between states. We would expect to see a reduced proportion of densely packed states and hence a markedly reduced number of highly connected subgraphs. To make this clear, consider the set of equally spaced states in $\mathbb{R}$. Each state has exactly two equidistant immediate neighbours to either side, after which it must search further away in order to find another neighbour. Now consider a regular lattice of states in $\mathbb{R}^2$. Each state now has four equidistant immediate neighbours; this pattern holds for increasing $\mathbb{R}^n$, with each state having $2n$ immediate neighbours. When constructing our network we connect each state to a fixed $k$ near neighbours, so for increasing dimension (namely, when $2n > k$) a state will only be connected to its immediate neighbours and no others. Hence, as a general rule, we would expect to see fewer occurrences of highly interconnected motifs as the local dimension of state space increases.

Of course, the attractors of meaningful systems rarely possess uniform density and will not resemble regular lattices. However, averaging across the entire manifold, we should expect the above phenomenon to be noticeable. The result of this is a decreased likelihood of the fully connected motif F appearing and a higher proportion of the sparse motif D, just as we see in the experimental data. Of particular note is the case of the Mackey-Glass system, which differs from the other chaotic systems in its motif rank ordering by having a significantly greater number of motifs D and E and less of motif F. This is consistent with our analysis, as the hyperchaotic Mackey-Glass system is in fact infinite dimensional, albeit embedded in a lower dimension. On the other hand, the chaotic Rössler, Lorenz, and Chua circuit systems are all approximately two dimensional, so exhibit similar motif frequencies.

Furthermore, it is important to note that our above arguments, although having been generated by considering flows, should apply equally well for discrete maps. The dimension of the attractor is well defined for maps, and hence we expect the same logic to hold. The only difference will be that periodic maps will no longer correspond to one dimensional attractors nor will chaotic maps correspond to two dimensional attractors. Indeed, we can already see this, as the results obtained by Xu et al. for the chaotic Logistic, Hénon, and Ikeda maps, all of which have less than two dimensional...
attractors, belong to the same ABCFDE superfamily as the one dimensional periodic flows.

B. Superfamily phenomena in randomly distributed states

To justify our claim that it is indeed the dimension of the attractor that determines motif rank ordering, let us consider a set of state vectors normally distributed in space. For the cases of $\mathbb{R}$ through to $\mathbb{R}^4$, Cartesian coordinates are independently sampled from a standard normal distribution a total of 10,000 times. We construct a $k$-nearest neighbour network from these states and perform motif frequency analysis. The results are presented in Figure 5. Comparing to motif rank ordering observed for periodic, chaotic, and hyperchaotic systems, our above claims seem to hold.

The one dimensional noise most closely resembles the motif distribution of period 3 Rössler flow and is consistent with the overall motif rank ordering of the periodic attractors. We also see a strong resemblance to the motif distribution of the Logistic, Hénon, and Ikeda maps, particularly the former two. Although not periodic, these attractors have been shown to have low dimensional attractors. The correlation dimensions of the Logistic and Hénon maps are $0.500 \pm 0.005$ and $1.25 \pm 0.02$, respectively, whilst the Hausdorff dimension of the Ikeda map is approximately $1.7$. Hence, we are unsurprised by the alignment of their motif rank ordering with that of one dimensional noise.

Our two dimensional motif distribution almost exactly matches that measured by Xu et al. for the chaotic Rössler and Lorenz systems, and Chua circuit. We have previously discussed the relationship between chaos and approximately two dimensional attractors, and our results seem to verify this. Furthermore, for our three and four dimensional noise, we see the rank ordering ABDCFE, with a consistent decreasing in the prevalence of motif F as the dimension increases. Hence, we are seeing a movement towards the ABDCFE rank ordering exhibited in Ref. 5 by the hyperchaotic, infinite dimensional Mackey-Glass system, and our results are consistent.

This analysis of random noise for increasing dimension provides important empirical evidence to support our claims that superfamily phenomena are determined by local dimension of the attractor. The overall motif rank ordering is always as expected for the dimension of the attractors, and in many cases the specific distributions are almost identical. We expect some variation in these distributions for different systems due to the unique structure of the different attractors such as edges, curvature, and invariant density, although the data show these effects are dominated by the attractor dimension.

FIG. 5. Motif frequency for normally distributed states in increasing dimension.
IV. INVERSION

In this section, we examine k-nearest neighbour network inversion. As mentioned earlier, Hirata et al.\(^7\) have previously proved inversion for \(\epsilon\)-recurrence networks. Here, we build upon their work and extend it to the \(k\)-nearest neighbour case.

In the simplest terms possible, we claim that the inversion algorithm for \(\epsilon\)-recurrence networks presented by Hirata et al. similarly applies to the \(k\)-nearest neighbour networks. Naturally, a perfect inversion is not achieved, and indeed the reconstructed time series exist on attractors that are topologically equivalent, and consequently that the inversion will preserve dynamics.

The proof presented here relies on an examination of the metrics that underpin the \(\epsilon\)-recurrence and \(k\)-nearest neighbour construction methods.

In the following, we will provide a new metric on phase space (here we will use \(\mathbb{R}^m\), equivalent to the Euclidean metric, under which the \(\epsilon\)-recurrence and \(k\)-nearest neighbour networks of a given time series yield identical results. That is, for each element of the time series, only its \(k\)-nearest neighbours (plus those extras as discussed in the definition of \(k\)-nearest neighbour construction) will be within an \(\epsilon\) distance of itself. Since the \(\epsilon\)-recurrence and \(k\)-nearest neighbour networks are now indistinguishable, the inversion algorithm of Hirata et al. will equally apply to the \(k\)-nearest neighbour case. Note that the inverted time series will be under the new metric, but since it is equivalent to the Euclidean metric, dynamics will be preserved.

Let \(G = (V, E)\) be a \(k\)-nearest neighbour network for some vector series \(\{v(i)\}\). We begin by defining the following quantities:

1. \(L = \max_{(i,j) \in E} \|v(i) - v(j)\|\), the maximum Euclidean distance between any two directly connected state vectors;
2. \(S_i \subset \mathbb{R}^m\), the set of points on the straight lines in state space between \(v(i)\) and its directly connected neighbours. Referred to as the edge points of \(v(i)\);
3. \(S = \bigcup S_i\), the set of all edge points;
4. \(l_i = \min_{(i,j) \in E} \|v(i) - v(j)\|\), the Euclidean distance between state vector \(v(i)\) and its nearest neighbour;
5. \(z_i = \min\{l_i/2, \inf_{y \in S_i} \|v(i) - y\|/2\}\), the minimum of either half \(l_i\) or half the distance from \(v(i)\) to the nearest edge not incident to itself.

Also note that the parameters \(\epsilon\) and \(k\) are the same as those chosen in the relevant \(\epsilon\)-recurrence or \(k\)-nearest neighbour construction methods.

We will proceed by defining two functions that weight points in phase space according to the \(k\)-nearest neighbour network. First, around each state vector, we construct the radial function

\[
\theta_i(x) = 1 + \left(\frac{L}{2z_i} - 1\right) \exp\left\{-\left(\frac{\|x - v(i)\|}{z_i}\right)^p\right\},
\]

where \(P > 0\) is a parameter that determines the severity of the weighting. Note that as \(P \to \infty\) we obtain

\[
\theta_i(x) = \begin{cases} 
\frac{L}{2z_i} & \|x - v(i)\| < z_i \\
1 & \|x - v(i)\| > z_i.
\end{cases}
\]

Consequently, by the definition of \(z_i\)

\[
z_i \leq \frac{l_i}{2} \implies \frac{L}{2z_i} \geq \frac{2L}{l_i} \implies L \geq 1
\]

\[
\theta_i(x) \geq 1.
\]

So for large \(P\), space is made “heavier” by \(\theta_i\) in a ball of radius \(z_i\) around the state vector \(v(i)\). Notice that by the definition of \(z_i\) no two balls will intersect each other (since \(z_i < l_i/2\)) nor will they cover any edges other than those incident to \(v(i)\).

Construct a second weight function

\[
\phi(x) = 1 - (1 - \delta) \exp \left\{-Q \inf_{y \in S} \|x - y\|\right\},
\]

where \(Q > 0\) is a second parameter similar to \(P\) which determines the severity of the weighting. As \(Q \to \infty\), we obtain

\[
\phi(x) \to \begin{cases} 
\delta & x \in S \\
1 & x \notin S.
\end{cases}
\]

For large \(Q\), space is made “lighter” (choose \(\delta \ll 1\)) by \(\phi(x)\) along the edges, but does not have any effect elsewhere.

With the above, we define a new distance function between any two points in space as a path integral

\[
d(x, y) = \frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i ds,
\]

along a continuous curve \(C\). The curve \(C\) is chosen from the set of all continuous curves starting at \(x\) and ending at \(y\) as that which minimises the value of the path integral. The parameter \(\kappa\) is a normalisation factor which scales space uniformly to ensure all directly connected state vectors are within \(\epsilon\) of each other.

Also note that the parameters \(\epsilon\) and \(k\) are the same as those chosen in the relevant \(\epsilon\)-recurrence or \(k\)-nearest neighbour construction methods.

Recall a metric is a function \(f(x, y)\) that satisfies the following properties:

(i) \(f(x, y) \geq 0\) and \(f(x, y) = 0\) if and only if \(x = y\);
(ii) \(f(x, y) = f(y, x)\);
(iii) \(f(x, z) \leq f(x, y) + f(y, z)\).

**Lemma.** The function \(d(x, y)\) forms a metric.

**Proof.** The function \(d(x, y)\) satisfies the metric conditions:

(i) Since \(\phi_i(x), \theta_i(x) > 0\), then the path integral is zero when \(x = y\) and strictly positive otherwise.
(ii) Holds by definition of path integrals.
So $d(x, y)$ is a metric on phase space. We will now show that it is also equivalent to the Euclidean metric. Recall two metrics $f(x, y)$ and $g(x, y)$ are equivalent if there exists positive constants $\mu, \nu$ such that for all $x, y$

$$\mu g(x, y) \leq f(x, y) \leq \nu g(x, y).$$

**Lemma.** The metric $d(x, y)$ is equivalent to the Euclidean metric $\|x - y\|$.

**Proof.** From above, $\theta_i$ and $\phi$ are continuous, bounded functions

$$1 \leq \theta_i(x) \leq \frac{L}{2\varepsilon_i};$$

$$\delta \leq \phi(x) \leq 1.$$ 

Hence

$$d(x, y) = \frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds$$

$$\geq \frac{\epsilon}{\kappa} \int_C \delta \prod_i 1 \, ds$$

$$= \delta \frac{\epsilon}{\kappa} \int_C ds$$

$$\geq \delta \frac{\epsilon}{\kappa} \|x - y\|.$$ 

The last line is an inequality, since the path $C$ is fixed to begin with. Since the integrand is now simply one, the straight line distance (that is the Euclidean distance) is always shorter than or equal in length to any other path connecting two points.

Similarly

$$d(x, y) = \frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds$$

$$\leq \frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds$$

$$\leq \frac{\epsilon}{\kappa} \int_C 1 \prod_i \frac{L}{2\varepsilon_i} \, ds$$

$$= \left( \prod_i \frac{L}{2\varepsilon_i} \right) \frac{\epsilon}{\kappa} \int_C ds$$

$$= \left( \prod_i \frac{L}{2\varepsilon_i} \right) \frac{\epsilon}{\kappa} \|x - y\|.$$

Here, $C'$ is the straight line between $x$ and $y$, and so the second line holds by the definition of $C$ as the path which minimises the path integral.

Setting

$$\mu = \delta \frac{\epsilon}{\kappa}, \quad \nu = \left( \prod_i \frac{L}{2\varepsilon_i} \right) \frac{\epsilon}{\kappa},$$

we obtain the result we require and have thus proven equivalence.

At this point, it becomes useful to pause and reflect on the properties of the metric we have constructed above, and how we intend to exploit them. In our above metric, these properties are a direct result of the three important terms we have included, namely, $\epsilon/\kappa$, $\phi$, and $\theta_i$. We will briefly discuss each of them, and how they act to deform the metric. Here, we use the discourse of metric deformation to refer to the terms in the integrand, which affect the value of the path integral. We use the term efficiency to refer to how well a given path minimises the path integral between two state vectors.

The $\epsilon/\kappa$ term is the easiest to consider as it is a simple normalisation factor. It uniformly scales phase space at all locations to ensure the maximum distance between any two directly connected state vectors is $\epsilon$.

The function $\phi$ acts to ensure that the most efficient path between any two state vectors is via the edge points. Since $\delta \ll 1$, we can say that those paths utilising the edges will always be much more efficient than those which do not. One possible issue that may arise with this is the intersection of edges with each other or with other nodes. In Figure 6(a), we see the unwanted intersection of two edges, unintentionally creating a connection between nodes $A$ and $C$, as well as $B$ and $D$. Figure 6(b) shows the similar issue of an edge falling on top of a vector other than its intended destinations. We can resolve both of these situations by having our edge perform a small detour around the intersection whenever this occurs. Sufficiently small detours will not meaningfully change the value of the path integral and hence will not affect our construction. Note this detour implicitly requires at least three dimension of space to be able to be performed. Thus, in lower dimensions, we can simply embed our space in at least $\mathbb{R}^3$ and continue normally.

The meaning of $\theta_i$ is not immediately obvious, and its importance will become evident in Equations (5) and (6). At this point, it is sufficient to note two important features of $\theta_i$. First, recall that in the limit $P \rightarrow \infty$ as seen in Equation (1),

![Figure 6](image-url)
\( \theta_i \) is only non-singular in an \( r_i \) radius around the state vector \( v(i) \). Thus, by our definition of \( r_i \), each \( \theta_i \) will only ever deform those edges incident to it nor will any two \( \theta_i \) ever deform the same region of space. Specifically, since \( r_i \leq \inf_{x \in S} ||v(i) - y||/2 \), no nonincident edge will fall within \( r_i \) of a state vector. Similarly, since \( r_i \leq l/2 \), the \( r_i \) balls around each state vector will never overlap.

We now examine the intended property of each \( \theta_i \) term, namely, that it contributes a distance of \( L/2 \) to every edge incident to \( v(i) \). Consider the path from a vector \( v(k) \) to a point \( x_k \) away (in the Euclidean metric) along one of its edges

\[
\int_C \phi \prod_i \theta_i \, ds = \int_C \theta_k \, ds
= \int_0^{\frac{L}{2x_k}} \frac{L}{2x_k} \, dr
= \frac{L}{2x_k} \frac{L}{2}
= \frac{L}{2}.
\]

So travelling to or from a state vector along an edge will always contribute at least \( L/2 \) to the path integral.

As a final note, we have taken the limits \( P, Q \to \infty \) in Equations (1) and (2) and taken \( \delta \) arbitrarily small. This is done to ensure that the deformation of space dies off sufficiently quickly from our desired region, and that the edges will always provide the shortest paths between nodes. In fact, taking limits is not necessary and was only done to simplify our interpretation of \( \phi \) and \( \theta_i \). Since we are only concerned with the finite data set \( \{v(i)\} \), there will always exist sufficiently large, yet finite, \( P \) and \( Q \) and sufficiently small \( \delta \) to achieve our desired outcomes. This is important as it prevents discontinuity in the metric that would break equivalence with the Euclidean metric.

We claim that under our new metric \( d(x, y) \) the \( \epsilon \)-recurrence network constructed from the vector time series \( \{v(i)\} \) is identical to the \( k \)-nearest neighbour network. That is, under the new metric, the distance between two state vectors will be less than \( \epsilon \) if and only if they are directly connected on the \( k \)-nearest neighbour network. To do this, we recall that the most efficient path between state vectors will always utilise the edge points. Further, we show that the distance between any two directly connected state vectors is greater than \( \epsilon/2 \). Hence, the most efficient path between two state vectors which are not directly connected will need to utilise at least two edges and will have a distance strictly greater than \( \epsilon \).

**Lemma.** The distance \( d(v(i), v(j)) > \epsilon/2 \) for all directly connected state vectors \( v(i), v(j) \).

**Proof.** Consider the set of all directly connected state vectors. From this set, take the minimally separated (under the new metric) pair \( (v(m), v(n)) \). Their separation is given by

\[
\frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds,
\]

where \( C \) is the curve which utilises the edge between them.

Similarly, take the pair \( (v(p), v(q)) \) of maximally separated, yet directly connected, state vectors. Their separation is given by

\[
\frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds = \int_C \prod_i \theta_i \, ds
\]

where \( C'' \) is the curve which utilises the edge between them.

Consider the ratio of these minimum and maximum distances

\[
\frac{\frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds}{\int_C \prod_i \theta_i \, ds} = \frac{\int_C \prod_i \theta_i \, ds}{\int_C \prod_i \theta_i \, ds}.
\]

Cancellation is appropriate, since \( \epsilon/\kappa \) is a constant and both paths \( C \) and \( C'' \) utilise their respective connecting edges where \( \phi(x) = \delta \) is also a constant. Hence, we need only to consider the affect of the \( \theta_i \) terms.

Let us consider the extreme cases for these minimal and maximal distances between connected state vectors.

The theoretical minimum distance will occur between two state vectors that are each other’s nearest neighbour and are not near any other edges, i.e., we have \( x_m = l_m/2 = l_n/2 = x_n \). Recall from the definition of \( \theta_i \) and our discussion in Equation (3) that the \( x_i \) (measured under the Euclidean metric) closest section of the edge incident to \( v(i) \) will always contribute a distance of \( L/2 \). Hence, the theoretical minimum distance will always be the sum of the \( x_m \) and \( x_n \) parts of each node (Fig. 7). It follows that the true minimum distance can be bounded by:

\[
\int_C \prod_i \theta_i \, ds \geq \frac{L}{2} + \frac{L}{2} = L.
\]

Recall that by definition, \( L \) is the maximal separation of directly connected state vectors before deformation by the set of \( \theta_i \). But, as discussed, each \( \theta_i \) contributes \( L/2 \) extra distance to each edge in the region around \( v(i) \), whilst the rest of the edge is not affected. So, the maximum distance will be made of up the two \( L/2 \) pieces from each node, and the remaining unchanged middle section which will be shorter than the original \( L \) (Fig. 8). It follows that we can bound the maximum separation of directly connected state vectors by:

\[
\int_C \prod_i \theta_i \, ds < \frac{L}{2} + \frac{L}{2} = 2L.
\]

Substituting Equations (5) and (6) into (4), we achieve

\[
\frac{\frac{\epsilon}{\kappa} \int_C \phi \prod_i \theta_i \, ds}{\int_C \prod_i \theta_i \, ds} > \frac{L}{2L} = \frac{1}{2}.
\]

Finally, recall that by definition of our normalisation factor \( \epsilon/\kappa \), the maximum distance between any two directly connected state vectors is in fact \( \epsilon \) itself. So we can say
and hence all directly connected state vectors are at least $\epsilon/2$ distance apart.

Using this, we prove that the $\epsilon$-recurrence network constructed under this new metric is identical to the original $k$-nearest neighbour network.

**Theorem.** The distance $d(v(i), v(j)) \leq \epsilon$ if and only if $(i, j) \in E$.

**Proof.** Recall that by construction (more specifically by choice of the constant $\kappa$), $d(v(i), v(j)) < \epsilon$ if $(i, j) \in E$.

Since $\delta \ll 1$, the most efficient path between any two state vectors, directly connected or not, will always utilise the edges. By the final lemma, each edge contributes at least $\epsilon/2$ to the path integral. The path between two state vectors which are not directly connected will need to travel along two or more edges, and thus the total distance will always be greater than $\epsilon$.

So we have shown that under our new metric the $\epsilon$-recurrence network exactly resembles the original $k$-nearest neighbour network. It now remains to understand how this is relevant to finding an inversion of the $k$-nearest neighbour network. For this, we recall that Hirata et al.\(^1\) have already presented an inversion algorithm for $\epsilon$-recurrence networks. This algorithm first applies a network metric to measure distances between nodes, and then uses a multidimensional scaling method to embed the nodes as points in some vector space, i.e., $\mathbb{R}^m$, whilst preserving the network metric. Hirata et al.\(^2\) showed this embedding successfully reconstructs the original time series, since the network metric is equivalent to the Euclidean metric on the original state vectors.

Since our $k$-nearest neighbour network can instead be interpreted as an $\epsilon$-recurrence network measured under an alternate metric, we expect the inversion algorithm to successfully invert, albeit with distances between state vectors being given under the new metric. We have shown these two metrics are in fact equivalent, implying they preserve topology of the attractor, and therefore dynamics are equivalent. In this way, our reconstructed vector time series will always be equivalent to its original, and inversion is successful.

The above is a theoretical proof, as in practice we do not have access to an analytical expression for the new metric $d(x, y)$, since it is constructed from the original time series. Thus, the multidimensional scaling algorithm will not be able to embed the states in space in a way that visually resembles the original time series. Instead, it will embed the modified distances in flat space (i.e., using the Euclidean metric). We expect this to result in an overall attractor which visually resembles its original (indeed it is topologically equivalent), but will be non-trivially stretched.

In fact, closer examination can help us predict the nature of this stretching. Our new metric is designed for the sole purpose of producing identical $\epsilon$-recurrence and $k$-nearest neighbour networks for our given vector time series. Effectively, it will ensure that exactly $k$ many neighbours will be within a distance of $\epsilon$ of any state vector. This is simply a flattening of invariant density, albeit on a length scale of $\epsilon$, with approximately $k$ state vectors per ball of radius $\epsilon$. Hence, we expect regions of relatively high density to be spread apart and relatively sparse regions to be pulled together.

This is evident in the examples given in Figure 9. Note that the reconstructions of all bar the period 6 Rössler attractor quite closely resemble their original time series. Moreover, in the cases of the Lorenz and chaotic Rössler attractors, we see exactly the stretching discussed above. The two lower lobes of the Lorenz attractor are pushed out, as time series elements are particularly dense in this area where the two wings intersect. Similarly, the peak of the chaotic Rössler attractor is still recognisable but is significantly smaller. This region is relatively sparse in the original attractor and is thus condensed in the reconstruction.

Further insight can be made by examining the period 1 and chaotic Rössler attractors. At first glance, the reconstructions appear to be quite dissimilar from their originals, yet they preserve some very important properties. First, in the period 1 case, we note that inversion has been successful, at least in a topological sense. Both the original and reconstructed time series quite clearly produce one dimensional manifolds. The only difference is the loss of global structure, where the period 2 nature has been unwrapped into a single period. Nonetheless, the local structure is preserved, and we can claim that the two manifolds are topologically equivalent. This behaviour is related to the nature of the $k$-nearest neighbour networks, which only capture local information by definition of their construction algorithms.

In the chaotic Rössler attractor, we see a similar loss of global information, as some kind of wave is added to the usually flat banded attractor. One important point to note is the preservation of the Möbius strip nature of both the original and reconstructed attractors, with a half twist occurring close to the peak.

Finally, it is undeniable that we have failed to reconstruct the period 6 Rössler attractor. We claim this is
pathological and is related to an under sampling of state vectors as opposed to a problem with the inversion algorithm. The period 6 attractor exhibits a very important dual stripped nature, with two very close strips remaining side by side for three periods. The distinction between the lines is particularly hard to see on the peak, as the plotting function is drawing straight lines between successive states on either side of the bump, obscuring the true dual stripped nature. On the other hard, the distinction is most evident at the furthermost region of the original attractor, where the two strips momentarily separate to reveal the fact that they are indeed two side by side and not one.

In fact, it is this region of separation which is causing the three looped nature of the reconstructed attractor. The time series does not have enough data points to sufficiently fill out the trajectory, causing state vectors to reach across to the neighbouring strip when searching for \( k \)-nearest neighbours to connect to in the network. As a result of this, the network resolves these close yet distinct strips as one, which in turn merged in the reconstruction. The region of separation is an exception to this, where the neighbouring strips are sufficiently far away to avoid reaching across of state vectors when finding neighbours. Two of the loops in the reconstructed attractor correspond to this region, one for each strip, whilst the third loop corresponds to the remainder of the attractor, which is resolved as a single trajectory.

We must remember this is a fault caused by the low data sample, not the inversion algorithm. This could easily be rectified by sampling more state vectors for the time series. We make note of it here to highlight the sensitivity of the inversion algorithm to these factors, namely, whether there are sufficient data points to adequately fill out the attractor.

A. Remarks on inversion

The most stark observation to be made from the above proof is the fact that at no point have we relied on the structure of the \( k \)-nearest neighbour network. This is counter-intuitive in that it would suggest that any arbitrary connected network can be re-imagined as an \( \epsilon \)-recurrence network using a similar process. It seems that this would imply that \( \epsilon \)-recurrence contain no information about the system, although this is clearly not the case based upon the extensive empirical evidence. The resolution to this apparent paradox lies in the examination of what information \( \epsilon \)-recurrence networks, as well as \( k \)-nearest neighbour networks, are truly capturing. Networks examine local structure of the systems. Each state vector shares an edge with only those other state vectors in its immediate neighbourhood, either via a fixed \( \epsilon \)-ball or by seeking \( k \) nearest neighbours. This fact can be exploited by forcing states that would normally be far away in a Euclidean sense to be close to one another by curving space.

To understand this curvature of space, consider three states A, B, and C lying on a straight line, as seen in Figure 10. Typically, we would expect the distance between A and C to always be greater than both the distance from A to B and from B to C. Hence, in an \( \epsilon \)-recurrence network, we never expect to see A and C connected unless both A and B, and B and C are too. Yet, if we fold the line on top of itself, we now see A and C are much closer to each other than to B. This closeness is an abuse of the embedding of the manifold (one dimensional) in a higher dimension (two dimensions), and indeed both A and C are still closer to B when travelling along the line. This folding is analogous to the way in which we deform our metric in order to achieve closeness between some state vectors and not between others. Just in the case of

![FIG. 9. Original time series above and their reconstructions below after being converted to \( k \)-nearest neighbour networks and then applying the inversion algorithm provided by Hirata et al. From left to right: chaotic Lorenz attractor \((a = 10, \beta = 8/3, \gamma = 28)\); period 2 Rössler attractor \((a = 0.1, b = 0.1, c = 6)\); period 6 Rössler attractor \((a = 0.1, b = 0.1, c = 12.6)\); and chaotic Rössler attractor \((a = 0.1, b = 0.1, c = 18)\).](image-url)

![FIG. 10. Curving space in order to bring previously distant states close together. C is brought close to A without also doing the same for their intermediary state B.](image-url)
the simple line, the metric can be deformed independently of any underlying \(k\)-nearest neighbour network and is the cause of our above apparent paradox, yet we claim it is not a problem in practice.

In any meaningful application of \(\epsilon\)-recurrence or \(k\)-nearest neighbour networks, we will always be dealing with manifolds that are sufficiently locally flat. In our above example, we saw that extreme curvature of the line was required to change the relative distances between the three states. In the case of a very large number of state vectors, this becomes increasingly difficult, with many extreme bends having to occur. This is obviously not the case in any real, meaningful system, and hence is not something we need to consider. Since both \(\epsilon\)-recurrence and \(k\)-nearest neighbour networks are constructed by drawing edges between close neighbouring states, we expect their structure to be relatively similar. Hence, the metric will never deform too severely. Specifically, both networks obey the important property that if \((i, k) \in E\) and \(|v(i) - v(j)| < |v(i) - v(k)|\) then \((i, j) \in E\). This is easy to verify by examining the definition of \(\epsilon\)-recurrence and \(k\)-nearest neighbour network construction. The fact that both classes of networks obey this property implies that the metric will never have to deform in a way, which changes the relative distances between a fixed state vector and any two others (as was the case in our line example).

Additionally, we have already noted that the deformation of the metric implies the reconstructed time series will never be exactly identical to the original. Again, this is not a problem in practice, since our proof is entirely theoretical and does not require perfect inversion for a number of reasons. First, the original inversion algorithm makes no claim to perfectly reconstruct time series from \(\epsilon\)-recurrence networks to begin with. The exact locations and distances are lost in the case of \(\epsilon\)-recurrence networks, we are only able to obtain relative distances between state vectors that are then arbitrarily embedded in space using a multidimensional scaling method. Similarly, our state vectors will often be obtained using Takens’ embedding theorem, which itself produces a topologically equivalent system, and not necessarily the exact original. So our inversion will distort the system no more than the \(\epsilon\)-recurrence inversion algorithm or Takens’ embedding theorem.

Most importantly, we must remember why we are interested in inversion at all. Whilst rarely useful in practice (it seems unlikely we would ever encounter such networks without also having their corresponding time series), we are most interested in exploring any potential information loss through network transformation. By proving inversion up to topological equivalence, we have successfully shown that all information about the dynamics of the system must be contained in the network. This is an important proof in that it provides a strong foundation for the use of \(k\)-nearest neighbour networks in the studying of dynamical systems.

V. CONCLUSION

In this paper, we have presented two major ideas concerning the use of complex networks as a means of studying non-linear dynamical systems. Focusing on the case of \(k\)-nearest neighbour networks constructed from vector time series measurements, we have investigated the cause of superfamily phenomena, as well as proved inversion of the network transformation.

Superfamily phenomena exhibited by \(k\)-nearest neighbour networks were analysed in the context of local dimension of the attractor. The Lyapunov spectrum was used to draw a link between dynamics and attractor dimension, whilst the embedding of specific motifs in different dimensions was examined. We saw that the likelihood of particular motifs occurring had strong dependence on the dimension of space, and hence that attractor dimension would affect the motif distribution of the networks. This was supported by comparing the results of Xu, Zhang, and Small for known systems with the motif distribution of \(k\)-nearest neighbour networks constructed from randomly distributed states in increasing dimension. A strong correlation was discovered between the motif rank distribution of the random distributions and those systems with similar attractor dimension, suggesting our analysis was well founded.

Inversion was proved by building upon the work done by Hirata et al.,7 for related \(\epsilon\)-recurrence networks. By constructing a new, equivalent metric under which \(\epsilon\)-recurrence and \(k\)-nearest neighbour construction methods produce identical networks, we were able to show both network types contained identical information about the system. Hence, we would expect the previously established \(\epsilon\)-recurrence method to apply with equal validity to \(k\)-nearest neighbour networks, supported through the study of a few common systems. The results of these case studies further revealed important information about the nature of \(k\)-nearest neighbour networks, particularly when inversion was not successful. These networks were shown to only capture local information about the attractor, as reconstruction often failed to retain global properties such as the period of cycles. Similarly, we saw cases in which insufficient local information was available, and hence the network failed to capture the correct structure of the underlying system altogether.

The work presented here provides an important foundation for the use of complex networks in studying non-linear dynamical systems. Beyond being able to validate past and future applications of the examined techniques, we have revealed important details about the nature of recurrence networks. Perspective is given to both future possibilities, as well as the limitations of their use as an analytic tool. This perspective was most prevalent in our proof of inversion from network space back to the time series. We noted that the \(k\)-nearest neighbour structure of the network was not exploited in the inversion, and hence our technique should theoretically work for any arbitrary network, albeit whilst unintentionally producing some severe results. Yet, we argued this was irrelevant to our overall aim, and that \(k\)-nearest neighbour networks are sufficiently similar to \(\epsilon\)-recurrence networks for the side effects of the inversion to be negligible.

As a continuation of the work presented here, we would first recommend investigation of this claim. Being able to quantify the severity of metric deformation in the specific
case of $k$-nearest neighbour networks is the next step in understanding the relationship between $k$-nearest neighbour and $\epsilon$-recurrence networks. Similarly, the pathological case we have presented with unsuccessful inversion suggests the existence of situations in which the $k$-nearest neighbour network fails to capture the correct information about the system. Future work could be done into providing conditions under which the use of $k$-nearest neighbour networks is appropriate. Finally, we acknowledge that we have only studied the specific case of $k$-nearest neighbour networks, just one in a whole suite of possible network construction techniques. Just as our work on inversion was heavily inspired by the case of $\epsilon$-recurrence networks, others could draw upon the ideas presented here in order to prove inversion for other network constructions.

ACKNOWLEDGMENTS

M.S. was supported by an Australian Research Council Future Fellowship (FT110100896) and Discovery Project (DP140100203).