Comparison study of typical algorithms for reconstructing time series from the recurrence plot of dynamical systems*

Liu Jie(刘 杰)[a)†], Shi Shu-Ting(石书婷)[b)], and Zhao Jun-Chan(赵军产)[b)]

[a)] Research Center of Nonlinear Science, Faculty of Mathematics and Computer Science, Wuhan Textile University, Wuhan 430073, China
[b)] Department of Mathematics, Faculty of Mathematics and Computer Science, Wuhan Textile University, Wuhan 430073, China

(Received 23 August 2012; revised manuscript received 24 September 2012)

The three most widely used methods for reconstructing the underlying time series via the recurrence plots (RPs) of a dynamical system are compared with each other in this paper. We aim to reconstruct a toy series, a periodical series, a random series, and a chaotic series to compare the effectiveness of the most widely used typical methods in terms of signal correlation analysis. The application of the most effective algorithm to the typical chaotic Lorenz system verifies the correctness of such an effective algorithm. It is verified that, based on the unthresholded RPs, one can reconstruct the original attractor by choosing different RP thresholds based on the Hirata algorithm. It is shown that, in real applications, it is possible to reconstruct the underlying dynamics by using quite little information from observations of real dynamical systems. Moreover, rules of the threshold chosen in the algorithm are also suggested.

Keywords: recurrence plot, chaotic system, time series analysis, correlation analysis

PACS: 05.45.Tp, 05.45.Ac, 05.45.Pq **DOI:** 10.1088/1674-1056/22/1/010505

1. Introduction

In the pioneering work by Ekmann *et al.* on recurrence plots (RPs),[1] a tool for visualizing a time series via a two-dimensional plot with vertical and horizontal time axes was first introduced. For each pair of time indices, a point is drawn in the two-dimensional plane if the respective states in the considered time series are close in some definition of distances. Otherwise, nothing is plotted there. From an RP, one can easily learn some characteristics of underlying time series.[2–5] During the past several decades, recurrence plots have been extensively investigated in relevant fields, such as nonlinear time series analysis, electroencephalogram (EEG) signal analysis, chemical two-phase flow analysis, etc.[6–20]

In Ref. [6], the authors answered a critical question: how much information is contained in a recurrence plot? They argued that under some conditions, it is possible to reconstruct an attractor using the recurrence plot, at least topologically. This means that all relevant dynamical information is contained in the recurrence plot. In Ref. [7], the same group discussed the issue of generating surrogates from RPs. These elementary studies have shown that the shapes of the original time series can be reproduced from RPs generated in low- or high-dimensional state space. In other words, RPs can basically preserve topological information, and one may calculate any dynamical invariants based on properly constructed RPs. In Refs. [8]–[10], several related methods have been proposed to fulfill similar functions, and found to be effective for obtaining the shape of original time series via RP analysis. The most widely used methods are the ones proposed in Refs. [6]– [9].

In this paper, we compare three related methods and offer some instructive rules for applications of reconstructing attractors of the underlying real time series. In Section 2, the recurrence plot and the three different methods are briefly introduced. In Section 3, the effectiveness of these methods is studied by means of toy, periodical, random, and chaotic signals. We also compare these methods in terms of signal correlation analysis. Finally, an application of reconstruction of a classical Lorenz chaotic system is illustrated by the recommended reconstruction method. A brief conclusion is given in the last section.

2. Three different reconstruction methods

In this section, we first recall the three most widely used methods for reconstructing a series from related RPs.

2.1. Definition of recurrence plot

In the past two decades, some new characteristics of a dynamical system arising from the widely observed phenomena of recurrences, named by the index of recurrence (such as recurrence rate, determinism, laminarity, entropy, length of diagonal, and trapping time), have received increasing attention,[2–5] since many natural dynamical processes exhibit recurrence behavior. In the context of time series analysis, we refer to a recurrence of a state $\boldsymbol{x}_i$ at time $t = i\Delta t$ (Δt is the sampling time, and $\boldsymbol{x} \in R^m$ is a state in the m-dimensional phase space) whenever the state of the system $\boldsymbol{x}_j$ at another time $j\Delta t$

*Project supported by the Key Project of Ministry of Education of China (Grant No. 2010141) and the National Natural Science Foundation of China (Grant No. 61203159).

†Corresponding author. E-mail: liujie_hch@163.com

© 2013 Chinese Physical Society and IOP Publishing Ltd

is similar to the initial state (i.e., $\boldsymbol{x}_i \approx \boldsymbol{x}_j$) or as close as we wish (but usually not identical).

The RPs visualize the behavior of trajectories in phase space. They are a graphical representation of the matrix

$$\boldsymbol{R}_{ij}(\varepsilon) = \Theta\left(\varepsilon - \|\boldsymbol{x}_i - \boldsymbol{x}_j\|\right), \tag{1}$$

where $\boldsymbol{x}_i \in R^{\mathrm{d}}$ represents the point in phase space, in which the system is situated at time i, $\Theta(\cdot)$ is the Heaviside function, ε is a threshold used to define the neighborhood of a state vector $\boldsymbol{x}_i$, and $\|\cdot\|$ represents the norm. One assigns a "black" dot to 1, and a "white" dot to 0. Such two-dimensional graphical representation of $R_{i,j}$ is called an RP.[1,2]

Remark 1 Although the recurrence concept can be directly generalized to unequally sampled data, we will only discuss the equally sampled cases below for the sake of simplicity. Some recent studies also generalized RPs to cross recurrence plots (CRPs). One can consult Refs. [2] and [20], and some recent references therein.

2.2. Method A: Thiel's algorithm

In Ref. [6], a condition for the reconstructability is given by Thiel's group, which is described below.

If it is impossible to divide the entries of the time series into two subsets to ensure that no point of one of the sets has a neighbor (with respect to its ε neighborhood) in the second set, then the time series is reconstructible.

In fact, most natural times series satisfy the above mentioned conditions for reconstructability. In their paper, the following algorithm was used to reconstruct the (univariate) time series from the RPs. The reconstructed time series in this way has the same rank order and the same number of entries as the one underlying the RP. The three main phases and a total of 10 steps to reconstruct the time series from the RPs are described as follows:[6]

Sort 1) If n columns of the matrix $R_{i,j}$ are identical, store the indices of the columns and cancel $n-1$ of them, so that every column is unique. 2) For all pairs of neighboring points with coordinates i and j (i.e., $R_{i,j}=1$), compute the number of neighbors of x_j which are not neighbors of x_i. We will call this number $n_{i,j}$. 3) There exist two points, say x_{j_1} and x_{j_2}. They satisfy $n_{i,j_{1/2}} = 0\ (\forall i)$. These two points are the maximum and minimum of the time series, respectively. 4) Choose one of these two indices as the starting point. (Call this index k.)

Iteration 1) Denote the last index which has been ranked in the reconstruction of the time series by k. If there is a unique minimum in the set of $\{n_{i,k}\}_i$, i.e., there is a $i_{\min}$ such that $n_{i_{\min},k} < n_{i,k}\ (\forall i \neq i_{\min})$, take the point with the index $i_{\min}$ as the second point. 2) If there is no unique minimum in the $\{n_{l,k}\}_l$, i.e., there is a set of m indices $\mu_1, \mu_2, \ldots, \mu_m$, so that $n_{\mu_1,k} = \cdots = n_{\mu_m,k} \leq n_{i,k}\ (\forall i)$, choose the minimum $w = \min\{n_{k,\mu_i}\}_i$. The next point for the reconstruction is x_w. 3) Iterate the first two steps of this phase, until all indices are ranked. Then, you have a rank order of the points of the time series n_i^{ord} which underlies the RP.

Final reconstruction 1) Generate random numbers so that each entry in the ordered series has one number. Then rank these random numbers y_i. 2) Generate a time series by putting the value y_i at the position n_i^{ord}. Then we obtain $y_{n_i}^{\mathrm{ord}}$. 3) Reintroduce (at the position of the identical column obtained in step 1) the values of the points at corresponding indices which remained in the RP.

2.3. Method B: Hirata algorithm

In Ref. [8], the authors proposed a method of reproducing distance matrices and original time series from recurrence plots. Their procedure can convert a recurrence plot to a weighted graph, and then estimate a distance between each pair of nodes on this weighted graph. They demonstrated their method by reproducing the topological shape of the original time series. The algorithm is described as follows.

i) Let $x(i)$ be a state at time index i $(1 \leq i \leq n)$. A function of two states denoted by R tells whether the two states are close or not. If they are close to each other, then R is reduced to 1. Otherwise, it equals 0. Then, an RP can be defined as $\{(i,j) : R(x_i,x_j) = 1,\ i,j = 1,2,\ldots,n\}$.

ii) Construct a graph from the RP. In such a graph, each node corresponds to a time index, and an edge exists between nodes i and j if the states x_i and x_j corresponding to the time indices i and j are close, namely, $R(x_i,x_j) = 1$.

iii) For each edge, assign a weight. Define G_i by $\{j : R(x_i,x_j) = 1\}$. The G_i is the set of time indices to which i is close. For each existing edge between i and j, define the weight w by

$$w(i,j) = 1 - \frac{|G_i \bigcap G_j|}{|G_i \bigcup G_j|}, \tag{2}$$

where $|A|$ denotes the number of elements in set A, $\bigcap$ represents the intersection of two sets, and $\bigcup$ shows the union of two sets.

iv) Calculate the shortest distance between each pair of nodes on the graph weighted by $w(i,j)$. One can use the fast Dijkstra method for calculating the shortest distance.

v) Use a method of multi-dimensional scaling, which arranges a set of points, so that these points preserve the given distance matrix denoted by $\boldsymbol{X}_{\mathrm{h}}$.

The authors in Ref. [8] used the classical multi-dimensional scaling method. For drawing graphs, they used the component with the largest eigenvalue (one may use another method for multi-dimensional scaling). Their algorithm can be treated as an extended application of Isomap,[12] which is widely used for dimension reduction. Edge weights defined by Eq. (2) are constructed in a new way. In order that the

method works well, all the nodes on the graph should be connected, which means there should be sufficient links in the RPs under the chosen threshold, i.e., there should be a path between every pair of nodes. Since the proposed method reproduces the distance matrices in a sense that the ratio of two distances is preserved, the scale of distances will be changed.[8] Thus, one should choose proper threshold values to reconstruct the underlying time series.

2.4. Method C: G—Z algorithm

In Ref. [9], Gutierrez and Zaldivar proposed another reconstruction method in which they reconstructed directly the time series starting from the adjacency matrix without using the transformation function proposed by Hirata. This method consists in the following five steps:[11]

1) Let $\boldsymbol{A}$ be the adjacency matrix of given undirected networks.

2) Calculate the covariance matrix of $\boldsymbol{A}$.

3) Calculate the eigenvalues and the eigenvectors of the covariance matrix.

4) Project the covariance matrix along the eigenvector basis.

5) Sum the modes to obtain the reproduced time series denoted by $\boldsymbol{X}_{\rm s}$.

3. Comparison research and correlation analysis

In this section, we compare the three most widely used methods for reconstruction algorithms. We also use the same toy model in Ref. [6], where the time series reads $\boldsymbol{X} = [3,2,4,8,2,5,7,6,0,2]'$. In Fig. 1, the related RP is given for illustration.

Then, we try to construct the underlying toy series via different algorithms, and then compare the obtained results. Figure 2 reveals the effectiveness of different algorithms. It can be seen that, the Hirata algorithm behaves most effectively. In fact, we also take periodical, random, and chaotic series as the underlying series. In Figs. 3, 5 and 7, the related RPs are given for illustrations of difference among different types of signals. The Hirata algorithm always works fairly well, which is much better than the other two algorithms.

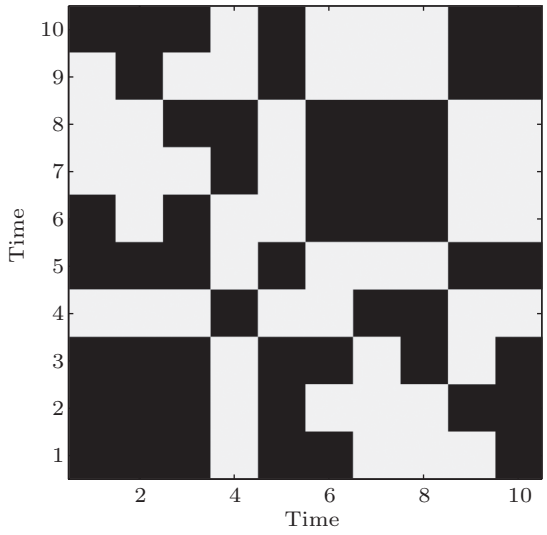


Fig. 1. Recurrence plot of the toy series $\boldsymbol{X}$. The threshold is set at $\varepsilon = 2$.

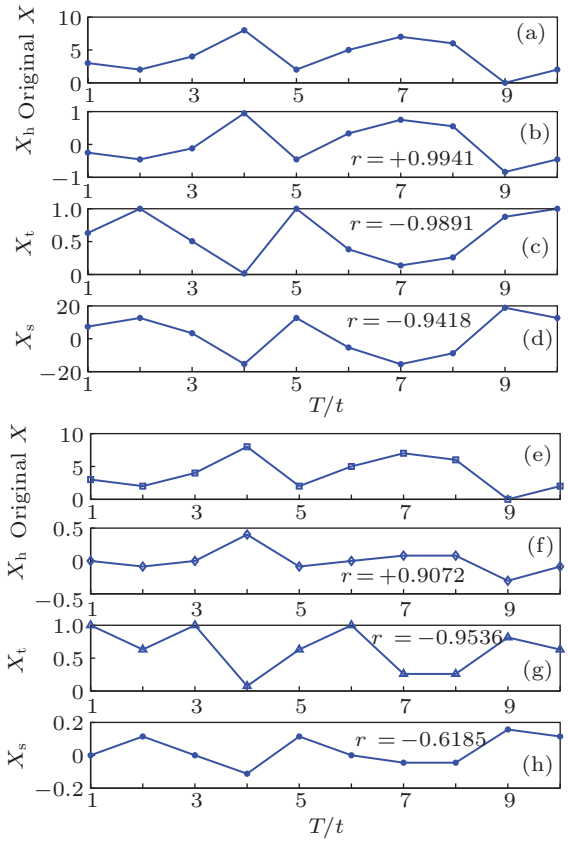


Fig. 2. (color online) Original toy time series and reconstructed time series via three different algorithms with correlation analysis results, where the threshold is set at 2 when constructing the RPs. The correlation coefficients are calculated via $r = \mathrm{corr}(X, X\mathrm{reconstructed})$ in Matlab software. (a) Results for the original time series are unnormalized in (c), (e), and (g); (b) results for the original time series are normalized in (d),(f), and (h).

Furthermore, the other two algorithms may construct anti-phase signals according to calculations of the correlation coefficients, which are negative in value (see Figs. 2, 4, 6, and 8). However, all three methods can successfully recover the topology of the underlying dynamical system according to our numerical investigations.

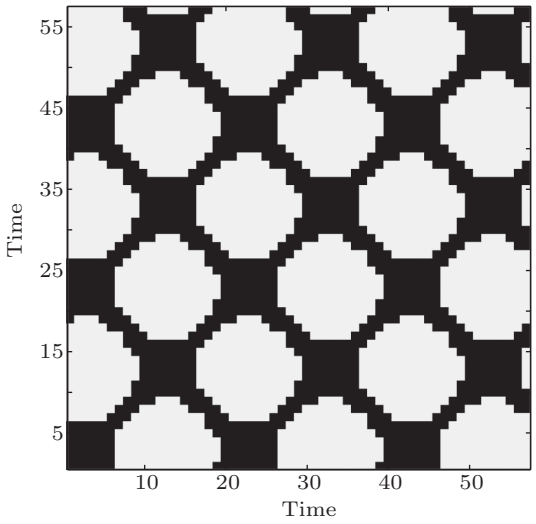


Fig. 3. Recurrence plot of $X = \sin t$ with $\varepsilon = 0.5$.

Remark 2 It should be addressed that, in Appendix A of Ref. [6], the author has illustrated the use of the algorithm with a toy example. However, there are some calculation or type mistakes in the illustration (e.g., Table 1 of Ref. [4] and in the toy model analysis). Here, we list the related results in Table 1 and delete the second and fifth columns. The bold numbers are different from those in Ref. [4].

Table 1. Related calculation results in phase 1 (step 2) of algorithm A.

i	j	n_{ij}	i	j	n_{ij}
1	3	1	6	8	1
1	6	2	7	4	0
1	10	1	7	6	2
3	1	0	7	8	1
3	6	1	8	3	**4**
3	8	2	8	4	0
3	10	1	8	6	1
4	7	1	8	7	0
4	8	2	9	10	2
6	1	**3**	10	1	1
6	3	**3**	10	3	2
6	7	1	10	9	0

Remark 3 According to a large amount of numerical experiments on different dynamical systems, we find the Hirata algorithm behaves most effectively with respect to the correlation coefficient analysis.

In Fig. 4, for the sinusoidal time series, the correlation coefficients are calculated via corr.m in Matlab, in which $(r_{X,X_{\text{Hirata}}}, r_{X,X_{\text{Gutirrez}}}, r_{X,X_{\text{Thiel}}})$ reads

$$(0.9944, -0.9902, -0.9934)_{\varepsilon=0.3}, \quad (3)$$

$$(0.9950, -0.9939, -0.9894)_{\varepsilon=0.5}, \quad (4)$$

$$(0.9932, -0.9913, -0.9653)_{\varepsilon=0.7}, \quad (5)$$

$$(0.9941, -0.9937, -0.9651)_{\varepsilon=0.9}. \quad (6)$$

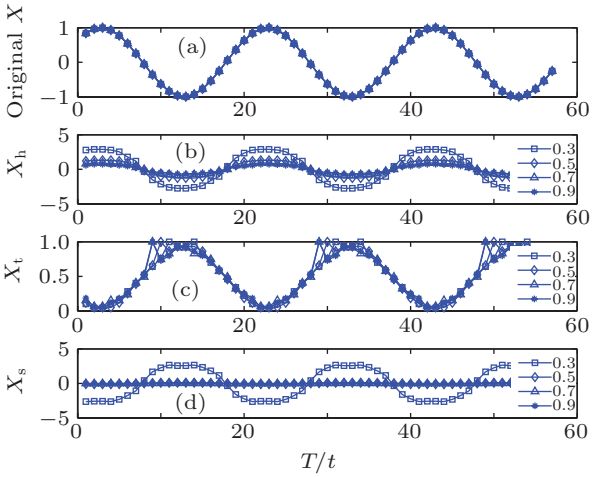


Fig. 4. (color online) (a) The original Sinusoidal time series. (b)–(d) The related reconstructed time series via three different algorithms with correlation analysis results, where the threshold is set at 0.3, 0.5, 0.7, and 0.9, respectively, when constructing the RPs. The time series is unnormalized.

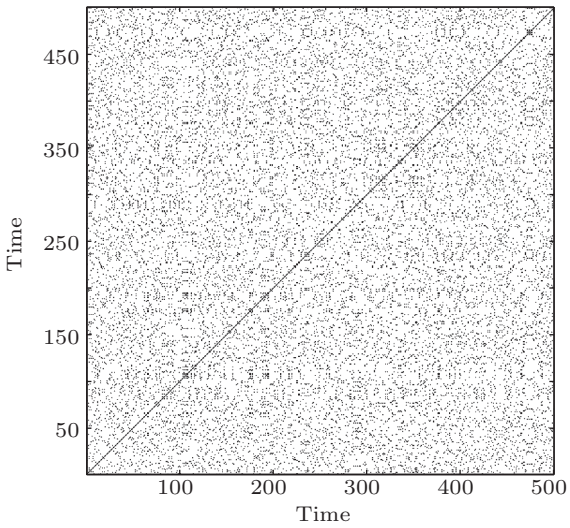


Fig. 5. Recurrence plot of $X = \text{rand}(500, 1)'$, $\varepsilon = 0.5$.

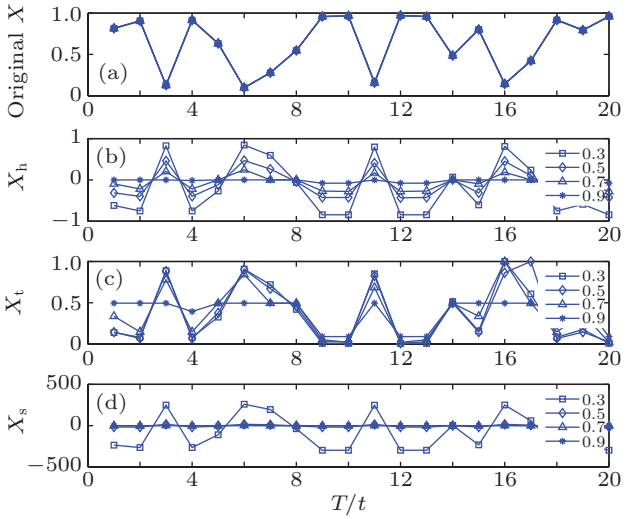


Fig. 6. (color online) (a) The original random time series. (b)–(d) The relevant reconstructed time series via three different algorithms with correlation analysis results, where the threshold is set at 0.3, 0.5, 0.7, and 0.9, respectively, when constructing the RPs. The time series is unnormalized.

In Fig. 6, for the random time series, the correlation coefficients are calculated via corr.m in Matlab, in which $(r_{X,X_{\text{Hirata}}}, r_{X,X_{\text{Gutirrez}}}, r_{X,X_{\text{Thiel}}})$ reads

$$(-0.9961, -0.9931, -0.9974)_{\varepsilon=0.3}, \tag{7}$$

$$(-0.9975, -0.9906, -0.9874)_{\varepsilon=0.5}, \tag{8}$$

$$(-0.9251, -0.9305, -0.9120)_{\varepsilon=0.7}, \tag{9}$$

$$(-0.6097, -0.5429, -0.6160)_{\varepsilon=0.9}. \tag{10}$$

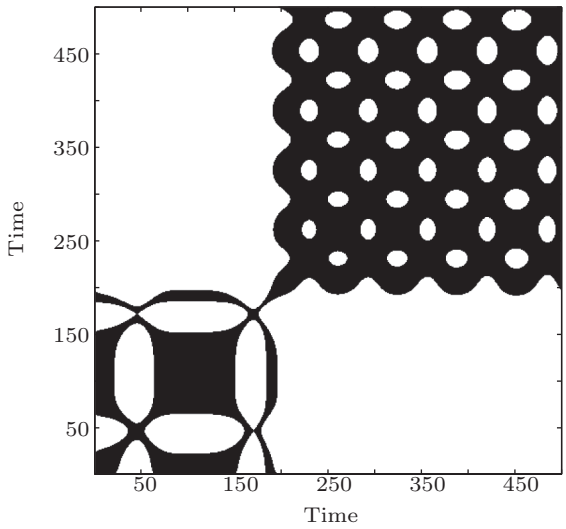


Fig. 7. Recurrence plot on $x(t)$ of a chaotic Lorenz system with $\varepsilon = 2$.

In Fig. 8, for the chaotic time series, the correlation coefficients are calculated via corr.m, in which $(r_{X,X_{\text{Hirata}}}, r_{X,X_{\text{Gutirrez}}}, r_{X,X_{\text{Thiel}}})$ reads

$$(0.9972, -0.9888, -0.9375)_{P=2}, \tag{11}$$

$$(0.9988, -0.9868, -0.9242)_{P=4}, \tag{12}$$

$$(0.9989, -0.9848, -0.9216)_{P=6}, \tag{13}$$

$$(0.9985, -0.9851, -0.9292)_{P=8}. \tag{14}$$

Remark 4 According to numerical experiments on different signals shown in Figs. 4, 6, and 8, it can be seen that the smaller thresholds chosen in the RP construction can help recover the underlying dynamics. In the limit case, ε is larger than the maximum distance, one will obtain a full connected RP matrix. It is probably a trivial case, so we disregard this situation. According to the numerical results, we also find that the method A (the Hirata algorithm) behaves most effectively with respect to the correlation coefficient analysis, no matter which thresholds are chosen.

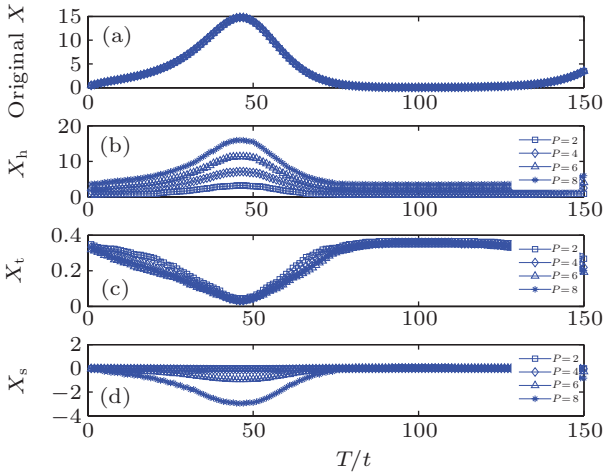


Fig. 8. (color online) Chaotic time series and reconstructed time series via three different algorithms with correlation analysis results, where the threshold is set to be $\varepsilon = P^{-1}\max((w-\min(w))\max(w)-\min(w))$, $w = x(t), y(t), z(t)$, where $P = 2$, 4, 6, and 8 when constructing the RPs. The time series is normalized.

4. Application: reconstruction of a chaotic dynamical system

In this section, we show an application illustration of the approach investigated in this paper. We will reconstruct the attractor from the RPs of an underlying dynamical system. Here, we choose the underlying time series to be states of the classical Lorenz chaotic system, whose dynamics are described as[13]

$$\dot{x} = 10(y-x), \tag{15}$$

$$\dot{y} = -y + (28-z)x, \tag{16}$$

$$\dot{z} = xy - \frac{8}{3}z. \tag{17}$$

The RPs of states $x(t)$, $y(t)$, and $z(t)$ with threshold $P = 2$ and that without any threshold are both given for illustrations in Fig. 9. We have integrated the Lorenz system using the ode45.m of ODE suite in Matlab with initial conditions: $(x(t), y(t), z(t))^{\mathrm{T}} = (1,1,1)^{\mathrm{T}}$, with fixed step of $h = 0.01$ in time interval $(0, 2000h)$ to obtain 2000 state points for illustrations.

Figure 10 presents the original attractor (up) and the reconstructed attractor built from RPs via the Hirata algorithm (down) resulting from three states' RPs shown in Fig. 9, where $P = 2$ (which means the threshold was set to be $\varepsilon = (1/2)\max((w-\min(w))/\max(w)-\min(w))$ and $w = x(t), y(t), z(t)$). One can easily find that the attractors have almost the same topology except the scale in this case, which means the attractor can even be reconstructed with little rough distance information in the state space. If we increase the value of P the threshold decreases, and more accurate distance information between states will be revealed under the connected condition as mentioned before. Then, the reconstruction result will certainly become much better according to the results in Section 3, since much more useful information can be obtained from the related RPs.

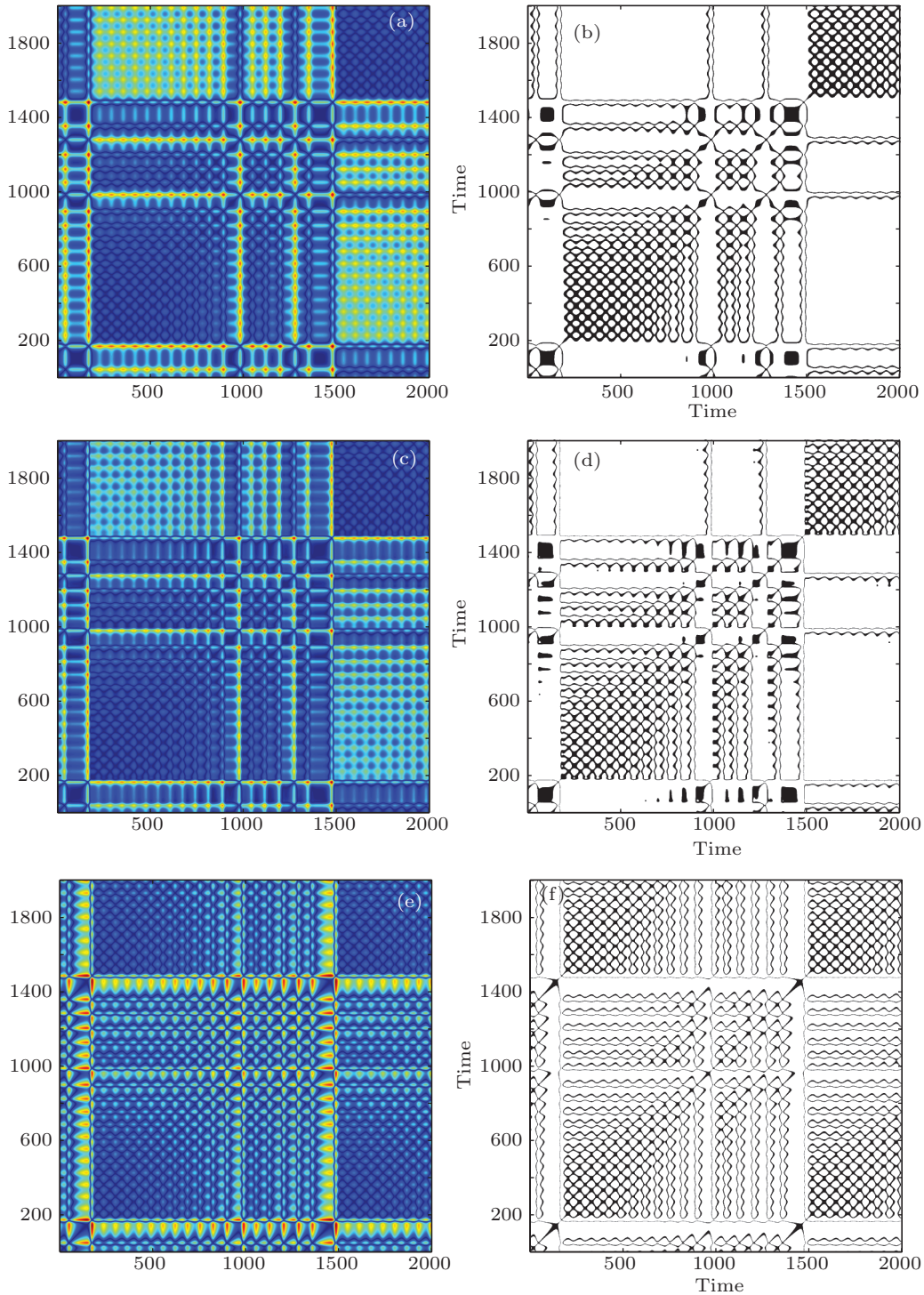


Fig. 9. (color online) Un-threshold recurrence plots of state (a) $x(t)$, (c) $y(t)$, and (e) $z(t)$ of classical Lorenz chaotic system. Recurrence plot of state (b) $x(t)$, (d) $y(t)$, and (f) $z(t)$ of classical Lorenz chaotic system with threshold $\varepsilon = p^{-1}\max((w-\min(w))/\max(w)-\min(w))$, $P=2$, $w=x(t),y(t),z(t)$, from which we reconstructed the attractor.

In fact, based on the un-thresholded RPs, one can fulfill reconstructing the original attractor via choosing different RPs thresholds based on Hirata algorithm. It is suggested that one can choose the threshold ε which satisfies the condition that the absolute value of the correlation coefficient of the reconstructed series and the experimental series is sufficiently close to 1 to guarantee the success of attractor reconstruction. It also means that, in real applications, it is possible that one can use quite little information from observations of real dynamical systems to reconstruct the underlying dynamics. If only one state can be obtained, one should try the Takens theorem to reconstruct phase space with proper time delay τ and embedded dimension m according to some effective methods mentioned in Refs. [14] and [19]. We will discuss such situations in the future.

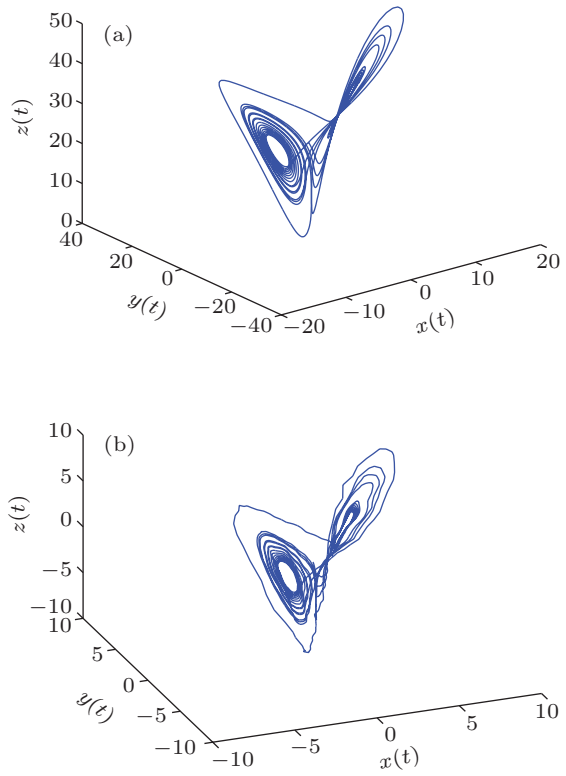


Fig. 10. (color online) The original attractor (up) and the reconstructed attractor built from RPs via Hirata algorithm (down), where the RP thresholds ε are both set at $\varepsilon = P^{-1}\max((w-\min(w))/\max(w)-\min(w))$, $P=2$, for three variables. In spite of the scale being different, both attractors have almost the same topology with similar largest Lyapunov exponents and fractional dimensions.

5. Conclusions

In this paper, we have compared the three most widely used methods of reconstructing the underlying time series based on its RPs. We give some suggestions for choosing a proper algorithm in related research via correlation coefficient analysis. In the future, we will further investigate the reason why in-phase or anti-phase reconstruction occurs in numerical investigations. Moreover, the applications of the effective Hirata algorithms will be extended to reconstruct attractors of fractional order dynamical systems.

References

[1] Ekmann J P, Kamphorst S O and Ruelle D 1987 *Europhys. Lett.* **4** 973
[2] Marwan N, Romano M C, Thiel M and Kurths J 2007 *Phys. Rep.* **438** 237
[3] Trulla L L, Giuliani A, Zbilut J P and Webber C L 1996 *Phys. Lett.* A **223** 255
[4] Zbilut J P and Webber C L 1998 *Phys. Lett.* A **246** 122
[5] Zbilut J P, Giuliani A and Webber C L 2000 *Phys. Lett.* A **267** 174
[6] Thiel M, Romano M and Kurths J 2004 *Phys. Lett.* A **330** 343
[7] Thiel M, Romano M, Kurths J, Rolfs M and Kliegl R 2008 *Phil. Trans. R. Soc.* A **366** 545
[8] Hirata Y, Horai S and Aihara K 2008 *Eur. Phys. J.* **164** 13
[9] Gutierrez E and Zaldivar J M 2000 *Earthquake Eng. Struct. Dyn.* **29** 1261
[10] Atay F M and Altintas Y 1999 *Phys. Rev.* E **59** 6593
[11] Strozzi F, Zaldivar J M, Pioljansek K, Bono F and Gutierrez E 2009 *EUR-Scientific and Technical Research Series* (Luxembourg) pp. 1–56
[12] Tenenbaum J B, de Silva V and Langford J C 2000 *Science* **290** 2319
[13] Lorenz E N 1963 *J. Atmos. Sci.* **20**
[14] Kantz H and Schreiber T 2003 *Nonlinear Time Series Analysis* (Cambridge: Cambridge University Press)
[15] Donner R V, Small M, Donges J F, Marwan N, Zou Y, Xiang R and Kurths J 2011 *Int. J. Bifurcat. Chaos.* **21** 1019
[16] Bian H R, Wang J, Han C X, Deng B, Wei X L and Che Yan Q 2011 *Acta Phys. Sin.* **60** 118701 (in Chinese)
[17] Xie Y, Xu J X, Kang Y M, Yang H J and Hu S J 2003 *Acta Phys. Sin.* **52** 1121 (in Chinese)
[18] Xia H C and Zhan Y Q 2004 *Acta Phys. Sin.* **53** 1299 (in Chinese)
[19] Chen S G and Yang Z A 1997 *Chin. Phys.* **6** 172
[20] Xia H C and Zhan Y Q 2004 *Chin. Phys.* **13** 1299