Available online at www.sciencedirect.com
SCIENCE DIRECT®
Physics Letters A 330 (2004) 343–349
PHYSICS LETTERS A
www.elsevier.com/locate/pla

How much information is contained in a recurrence plot?

Marco Thiel *, M. Carmen Romano, Jürgen Kurths

University of Potsdam, Nonlinear Dynamics, Am Neuen Palais 10, 14469 Potsdam, Germany

Received 18 February 2004; received in revised form 15 April 2004; accepted 7 July 2004
Available online 12 August 2004
Communicated by C.R. Doering

Abstract

Recurrence plots have recently been recognized as a powerful tool for the analysis of data. Not only the visualization of structures of the time series but also the possibility to estimate invariants from them and the possibility to analyze non-stationary data sets are remarkable. However, the question of how much information is encoded in such a two-dimensional and binary representation has not been discussed so far. In this Letter we show that—under some conditions—it is possible to reconstruct an attractor from the recurrence plot, at least topologically. This means that all relevant dynamical information is contained in the plot.
© 2004 Elsevier B.V. All rights reserved.

PACS: 07.05.Kf

Keywords: Recurrence plots; Reconstruction

1. Introduction

Using recurrence plots (RPs) for the analysis of time series, allows not only to visualize but also to quantify structures hidden in the data. However, the question arises, how much information can be extracted from such a plot. But the results presented here on this problem will not only give an answer to this rather theoretical question. They could also be the bases for the issue of generating appropriate surrogates from RPs.

RPs visualize the behavior of trajectories in phase space [1,2]. They are a graphical representation of the matrix

$$\mathbf{R}_{i,j} = \Theta\big(\varepsilon - \|\vec{x}_i - \vec{x}_j\|\big), \quad i, j = 1, \ldots, N, \tag{1}$$

where $\vec{x}_i \in \mathcal{R}^d$ stands for the point in phase space at which the system is situated at time i, ε is a predefined threshold and $\Theta(\cdot)$ is the Heaviside function. One assigns a "black" dot to the value one and a "white" dot to the value zero. The two-dimensional graphical representation of $R_{i,j}$ then is called a RP.

* Corresponding author.
E-mail address: thiel@agnld.uni-potsdam.de (M. Thiel).

0375-9601/$ – see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.physleta.2004.07.050

An unthresholded RP is not binary but its matrix $R^u_{i,j}$ is given by the (real valued) distances of the vectors $\vec{x}_i$ and $\vec{x}_j$. The matrix then is usually represented in a two-dimensional colored plot. It has been shown [3] that from an unthresholded RP it is possible to reconstruct the time series. But unthresholded RPs are more difficult to quantify than binary RPs. For this reason, in data analysis usually binary RPs are used.

To quantify complex structures that occur in RPs, Webber and Zbilut have proposed several measures in their seminal paper [4], that constitute the recurrence quantification analysis (RQA). Basing on these measures, RPs have become very popular and have been applied to various experimental data, especially in physiology and earth science [4–9]. A further potential advantage of RPs is that they enable the computation of dynamical invariants, such as the correlation entropy K_2 and correlation dimension D_2 [10–13]. One crucial point in the analysis based even on binary RPs is whether the estimates of the invariants depend on the parameters used in the computations. To calculate an RP, one has to fix three parameters in advance. One of them is the threshold ε. Recently a lower bound for ε in the presence of noise has been determined [14]. It has been shown that, in order to resolve fine structures, ε should not be chosen too large either [6,10]. So upper and lower bounds for ε are known, at least theoretically. However, up to now it was not known for what choice of ε the information content of the RP is somewhat optimal. In this Letter we will show for which range of thresholds the RP contains all topological information about the underlying attractor.

A further important point is that in the case of experimental data there is often only one component (i.e., a univariate time series) available. Hence, embedding is used to reconstruct the phase space. Using delay embedding, the embedding dimension d and the delay τ needed for the embedding of the time series

$$\vec{x}_i = (x_i, x_{i+\tau}, \ldots, x_{i+(d-1)\tau})^{\mathrm{T}} \qquad (2)$$

have additionally to be fixed [15]. To estimate the "optimal" embedding dimension d and the delay τ methods are presented in [16–18]. Fig. 1 presents RPs for three different systems (uniformly distributed and independent noise, a sine function and the chaotic Rössler system with standard parameters) and for different embedding parameters. The plots show that the embedding via Eq. (2) influences the structure in the

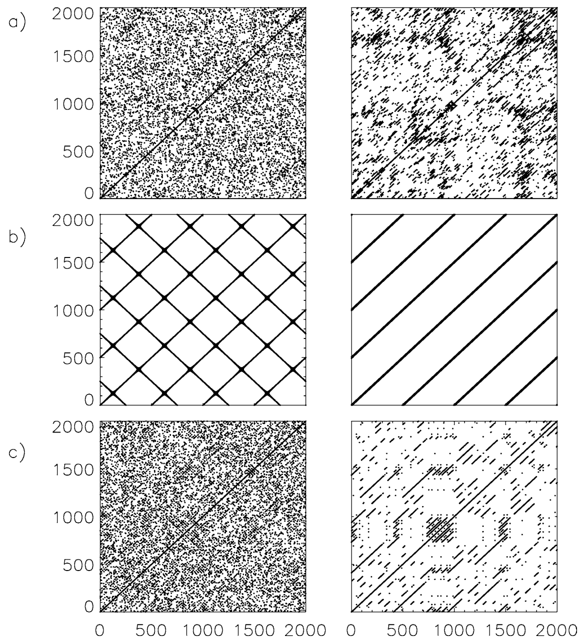

Fig. 1. RPs for uniformly distributed noise (a), the sine function (b) and the Rössler system (c). The left panel shows the plots for $d = 1$. The right panel represents the plots for $d = 14$, $d = 2$ and $d = 3$, respectively, from top to bottom, ε is chosen so that the recurrence rate (percentage of black point in the plot) is the same for the embedded and non-embedded time series.

RPs drastically. The most important feature to quantify the predictability of the system is the distribution of diagonals [19], which seems to depend on the embedding, Fig. 1(c). Hence, it seems to be obvious, that for a quantification of these structures a suitable embedding has to be used. However, we have recently shown, that dynamical invariants can be estimate from the RP even if no embedding is used [10]. In this Letter we will demonstrate that a binary RP obtained from only one component of a system, all dynamical information about the underlying system is contained. Therefore, we reconstruct the time series from the RP and then use Takens theorem to reconstruct the attractor.

The outline of the Letter is as follows. First, we will introduce an algorithm to reconstruct the (univariate) time series from its RP. Then, we will show the results of this algorithm when applied to time series from different paradigmatic systems (logistic map, the Rössler system, and independent noise). In a discussion of the algorithm we give limits for ε and discuss limitations of the reconstructability.

2. Reconstruction of time series from an RP

In this section we want to study how to reconstruct a time series $\{x_1, x_2, x_3, \ldots, x_n\}$ from a binary (i.e., black and white) RP $R_{i,j}$ Eq. (1). Let us further assume that the values of x_i, are distributed rather continuously between the minimum $x_{\min}$ and the maximum $x_{\max}$ of the time series with respect to the threshold ε. By this, we mean that there is no subinterval in $[x_{\min}; x_{\max}]$ of length ε which does not contain at least one point of the time series. In other words we assume that the following condition is fulfilled:

Condition for the reconstructability. *If it is impossible to divide the entries of the time series in two subsets, such that no point of one of the sets has a neighbour* (*with respect to its ε-neighbourhood*) *in the second set, then the time series is reconstructible.*

We will show that then the following algorithm can be used to reconstruct the (univariate) time series from the RP. It is important to note that the so reconstructed time series has the same rank order and the same number of entries as the one underlying the RP. Additionally reconstructing by Takens theorem [15] an attractor from this time series yields a topologically identical object.

The algorithm to reconstruct the time series from the RP consists of three main phases and a total of 10 steps.

(1) Sort
   (a) 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;
   (b) Compute for all pairs of neighbouring points with coordinates i, j (i.e., $R_{i,j}=1$) the number of neighbours of x_j which are not neighbours of x_i. We will call this number $n_{i,j}$;
   (c) There exist exactly two points, say x_{j_1} and x_{j_2}, so that $n_{i,j_{1/2}} = 0\ \forall i$. These two points are the maximum and the minimum of the time series;
   (d) Choose one of these two indices as starting point. (We will call this index k.)
(2) Iteration
   (a) Denote the last index which has been ranked in the reconstruction of the time series so far by k. If there is a unique minimum in the set of $\{n_{i,k}\}_i$, i.e., there is a $i_{\min}$ so 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;
   (b) If there is no unique minimum in the $\{n_{l,k}\}_l$, i.e., there is a set of m indices $\{\mu_1, \ldots, \mu_m\}$, so that $n_{\mu_1,k} = \cdots = n_{\mu_m,k} \leqslant n_{i,k}\ \forall i$, choose the minimum $w = \min\{n_{k,\mu_i}\}_i$. The next point for the reconstruction then is x_w;
   (c) Iterate the two steps (a) and (b) 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.
(3) Final reconstruction
   (a) Generate random numbers so that for each entry in the ordered series there is one number. Then rank order these random numbers y_i;
   (b) Generate a time series by putting the value y_i at the position n_i^{ord}. Then you obtain $y_{n_i^{\mathrm{ord}}}$;
   (c) Reintroduce at the position of the "identical columns" obtained in step 1 the values of the points at the corresponding indices which remained in the RP.

In Appendix A we will illustrate the use of this algorithm by an easy example.

3. Application of the algorithm to three different systems

Now we show, that the algorithm works well for very different systems, both dynamical and stochastic ones. The algorithm does not depend on the dynamics of the underlying system.

(A) We start with the reconstruction of the time series of a logistic map ($x_{n+1} = 4x_n(1-x_n)$), Fig. 2(a). Given only the RP from a simulation of the logistic map (in this case we used $\varepsilon = 0.1$, length of the time series $N = 1000$), we reconstruct based on the upper algorithm the time series which is displayed in Fig. 2(b). Plotting the original and the reconstructed time series one on top of the other, one obtains Fig. 2(c). Note, that the reconstruction is much more precise than the error bounds given by the threshold ε. Actually, the precision increases with the length of the underlying time series. The main dynamical properties of the time series (e.g., the correlation entropy and

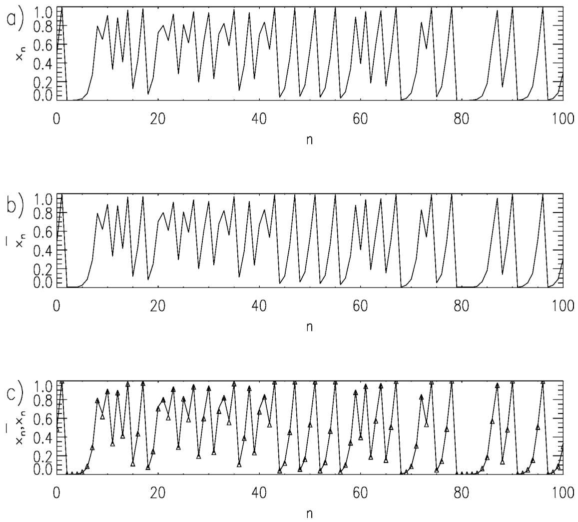

Fig. 2. Reconstruction of a time series of a logistic map from an RP. (a) Original time series, (b) reconstructed time series, (c) both time series: original one (line) reconstructed one (points).

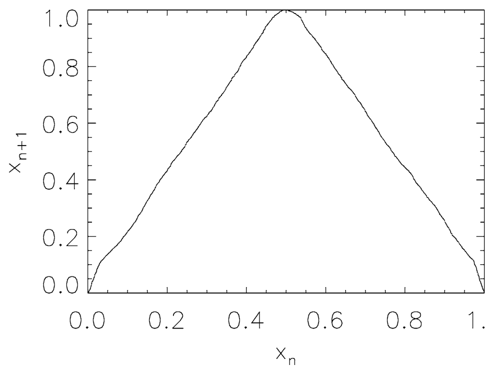

Fig. 3. x_n vs. x_{n+1} diagram for the reconstructed logistic map. Due to the different distribution of the time series it is not a parabola.

dimension) are captured. This can also be seen in an x_n vs. x_{n+1} plot (Fig. 3). Plotting this diagram for the original time series yields a parabola. For the reconstructed time series one finds a graph which seems to be a continuously deformed parabola. This effect is mainly due to the different distribution of the original and the reconstructed time series. However, it is still a one humped map and also its correlation entropy is unchanged.

(B) The reconstruction also works for continuous systems, such as the Rössler system [20]. Fig. 4(a) shows a portion of the time series of the x-component of the Rössler attractor with standard parameters.

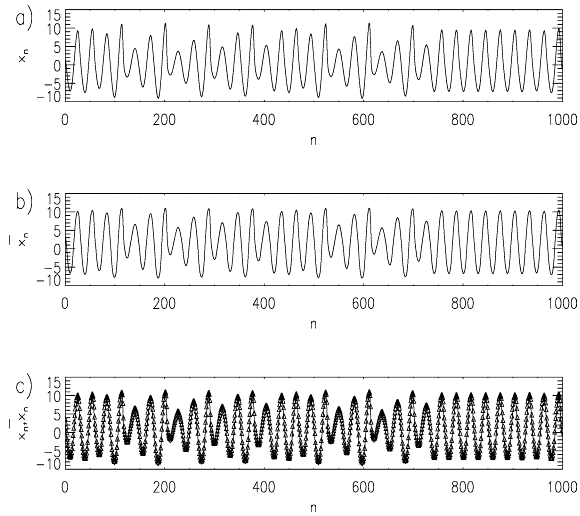

Fig. 4. Reconstruction of a time series of the x-component of the Rössler system from a recurrence plot. (a) Original time series, (b) reconstructed time series (for better comparability the standard deviation and the mean have been adapted), (c) both time series: original one (line) reconstructed one (points).

The reconstruction coincides nearly perfectly with the original time series, Fig. 4(c).

(C) The next case we want to present is independent uniformly distributed noise. This system is not dynamical, but the reconstruction of the trajectory based on the algorithm still works. Fig. 5(a) represents the original time series. Fig. 5(b) and (c) represent the reconstructed time series. There is nearly perfect coincidence.

In all the cases the algorithm succeeded in reconstructing the time series from the RP. In the next section we will discuss more of the characteristics and limitations of the algorithm.

4. Discussion of the algorithm

The reconstruction algorithm works by considering the neighbourhoods of the points of the time series. Condition for the reconstructability assures, that the neighbourhood overlap sufficiently. This makes it possible to reconstruct the time series. It is equivalent to the condition that in the projection of the values there is no ε-interval void of points.

Assuming that the values are uniformly distributed, one can estimate the number of points which

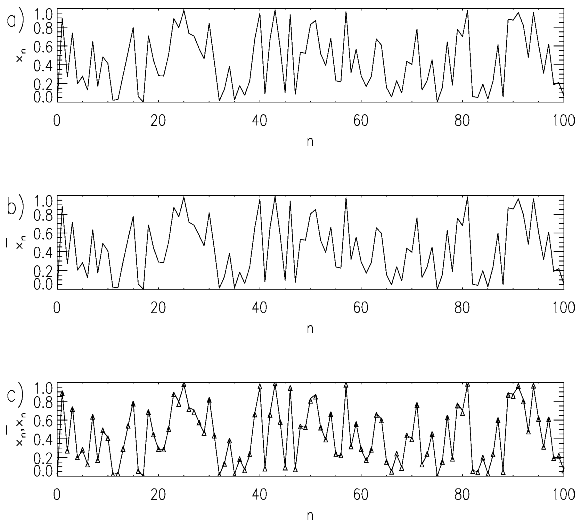

Fig. 5. Reconstruction of a time series of independent uniformly distributed noise. (a) Original time series, (b) reconstructed time series, (c) both time series: original one (line) reconstructed one (points).

are needed to reconstruct the time series for a given threshold ε. The distance d of two neighbouring points in the interval of the values $[x_{\min}, x_{\max}]$ is then exponentially distributed

$$p(d) = N \cdot e^{-N*d}, \tag{3}$$

where N is the length of the time series. Let us, without loss of generality, assume that the interval in which the values are distributed is the unit interval. Then there are $N + 1$ intervals, which have to be all smaller than ε. One obtains the following relation between the number N of points in the time series, the threshold ε and the probability to find a void interval which is larger than the threshold p.

$$P = \left(1 - e^{-N\varepsilon}\right)^{N+1}. \tag{4}$$

This formula allows to estimate that in order to be able (with a probability of about 0.999) to reconstruct the time series (and the attractor) for an RP which has a recurrence rate of 1%, one should have more than about 1400 points in the time series. Using $\varepsilon = 0.1$ one only needs about 90 points to reconstruct the time series.

Hence, if ε is larger, one needs less points for the algorithm to reconstruct the time series. If, on the other hand, ε is too large, the reconstruction algorithm works but cannot distinguish different points properly. Let us take example (C) of values which are distributed uniformly in the unit interval. Then, if ε is $0.5 + \delta$, a band of width 2δ around 0.5 has all points of the time series as neighbours. All these points have equal columns in the RP and are not distinguishable. (They are "canceled" in the first step of the algorithm.) Whenever, $\varepsilon < 0.5$, i.e., half the interval width of the value of the time series, the time series can be reconstructed as accurately as one wishes by considering sufficiently long time series.

Note, that based on Takens theorem [15], it is possible to reconstruct the attractor from the reconstructed time series. Hence, the attractor can be recovered from the RP (of only one component) of the system, at least topologically.

This means, that the RP contains all topological information of the underlying system, even though it is only computed from one of its components. Fig. 6 shows the reconstruction of the Rössler attractor from the reconstructed time series. However, the reconstruction of the time series from the RP of more than one component of the system, e.g., the three-dimensional vectors of the Rössler system, is not possible with this algorithm. Such a plot is the point-wise product of the RP of the single components, if one uses the maximum norm in Eq. (1). Hence, one loses information. The open question is, if it is possible to reconstruct the attractor from such a multidimensional RP. This is an important problem as the RP of only one component (i.e., the projection of the attractor onto one coordinate axis) contains seemingly less information than the whole n-dimensional phase space. It even does not represent real recurrences but due to the projection also false ones ("false nearest neighbours").

5. Conclusions

In this Letter we have shown—based on Takens theorem—that it is possible to reconstruct an attractor from its RP obtained from one component. This result is an extension of the work [3], where it was shown that the attractor can be reconstructed from the unthresholded RP. It must be noted that with our algorithm it is also possible to reconstruct time series which have a stochastic component from an RP, i.e., it does not matter whether the time series is stochastic, dynamical or a dynamical system corrupted by noise.

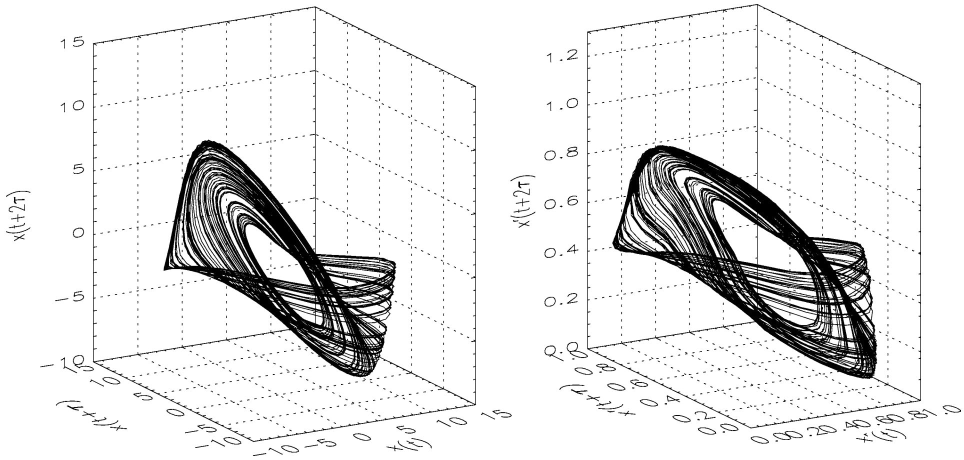

Fig. 6. (a) Reconstruction of the phase space by Takens theorem, based on the x-component of the Rössler system (b) reconstruction of the attractor of the Rössler system from the RP of a time series of its x-component.

An interesting point is that the RP of only one component is sufficient to reconstruct the attractor. This is linked to the fact presented in [10] that the RP of one component is sufficient to estimate both the correlation entropy and dimension. Also note, that the algorithm presented in this Letter cannot reconstruct the attractor from a RP of a vector valued, i.e., embedded time series. It is an open question if such an algorithm can be constructed.

Our results are of mainly theoretical relevance as they show that also in a binary RP all topological information is conserved. The algorithm to reconstruct the time series allows to state some fundamental relations for information theoretical considerations, e.g., how to choose the threshold ε so that based on recurrences the attractor can be reconstructed.

However, our results could be the basis for a forthcoming study on generating of surrogates from an RP.

Beyond the topological information, there is also some information about the distribution of the data contained in the RP. To extract this information, too, is an open problem and will be the objective of a forthcoming paper.

Acknowledgements

This project was supported by the "DFG Priority Program 1114" and the "Internationales Promotionskolleg Computational Neuroscience of Behavioral and Cognitive Dynamics".

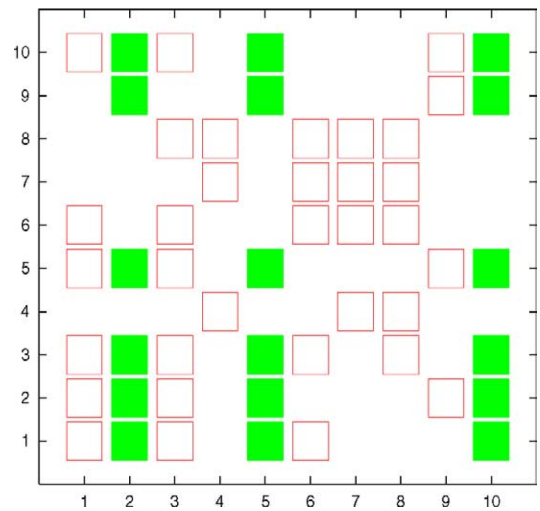

Fig. 7. Toy recurrence plot obtained from a time series of ten entries.

Appendix A. Practical reconstruction a time series from an RP

To illustrate how the algorithm works, we apply it to the RP of the following toy time series $\{3, 2, 4, 8, 2, 5, 7, 6, 0, 2\}$. The threshold is chosen $\varepsilon = 2$. The corresponding RP is represented in Fig. 7. Now we apply the algorithm described in the last sec-

Table 1

i	1	1	1	2	2	2	3	3	3	3	4	4	6	6	6	6	7	7	7	8	8	8	8	9
j	2	3	6	1	3	9	1	2	6	8	7	8	1	3	7	8	4	6	8	3	4	6	7	2
$n_{i,j}$	1	1	2	1	2	0	0	1	1	2	1	2	1	1	1	1	0	2	1	2	0	1	0	2

tion. First, we notice that the columns 2, 5, 10 are identical (here marked in grey). Following the instruction in step “1” we cancel columns 5 and 10, i.e., we will ignore them carrying out the next steps of the algorithm.

Next, we compute the $n_{i,j}$ for all neighbours. See Table 1.

Following step “3” we search the two indices j_1 and j_2 for which $n_{i,j_{1,2}} = 0\ \forall i$. We find $j_1 = 9$ and $j_2 = 4$. This means that one of the values x_9 and x_4 is the largest and one is the smallest of the time series. We choose one of them, e.g., $j_1 = 9$ respectively x_9 (step 4).

Then (step 5), we search all neighbours of x_9. There is only one such value x_2. This is the next value in the rank order. Hence, we have $x_9 < x_2$ (or $x_9 > x_2$).

Next, we search all neighbours of x_2. These are x_1, x_3, x_9 and x_9 is already arranged. Hence, we consider $n_{2,1} = 1$ and $n_{2,3} = 2$. The minimum is $n_{2,1} = 1$, i.e., x_1 is next in the series. We have $x_9 < x_2 < x_1$ (or $x_9 > x_2 > x_1$).

The neighbours of x_1 are x_2, x_3, x_9. x_2 is already arranged in the reconstructed time series. Then, we consider $n_{1,3} = 1$ and $n_{1,6} = 2$. This means that x_3 is the next value in the time series: $x_9 < x_2 < x_1 < x_3$ (or $x_9 > x_2 > x_1 > x_3$). Proceeding with the neighbours of x_3, we find $n_{3,6} = 1$ and $n_{3,8} = 2$. Hence, we have $x_9 < x_2 < x_1 < x_3 < x_6$ (or $x_9 > x_2 > x_1 > x_3 > x_6$).

The next step is different from the last steps. The two relevant neighbours of x_6 are x_7 and x_8. As $n_{6,7} = 1$ and $n_{6,8} = 1$, we have to apply step “6” of the algorithm and consider $n_{7,6} = 2$ and $n_{8,6} = 1$. The minimum is $n_{8,6} = 1$. Hence, we get $x_9 < x_2 < x_1 < x_3 < x_6 < x_8$ (or $x_9 > x_2 > x_1 > x_3 > x_6 > x_8$).

In the next iteration we consider $n_{8,4} = 0$ and $n_{8,7} = 0$. Again, we must follow step “6”. The minimum of $n_{4,8} = 2$ and $n_{7,8} = 1$ is $n_{7,8}$. Hence, the next point is x_7 and the last one x_4. We obtain the time series $x_9 < x_2 < x_1 < x_3 < x_6 < x_8 < x_7 < x_4$ (or $x_9 > x_2 > x_1 > x_3 > x_6 > x_8 > x_7 > x_4$).

Next, one generates 9 random numbers and orders them. Then x_9 is identified with the smallest number, x_2 with the smallest but one and so on.

Finally, one reintroduces x_5 and x_{10} which have been canceled in the first step. Both are set equal to x_2.

Then, the algorithm is finished and one has a time series $\{x_i\}$ which is reconstructed from the RP.

References

[1] J.-P. Eckmann, S.O. Kamphorst, D. Ruelle, Europhys. Lett. 4 (1987) 973.
[2] J.P. Zbilut, C.L. Webber, Phys. Lett. A 171 (1992) 199.
[3] G. McGuire, N.B. Azar, M. Shelhamer, Phys. Lett. A 237 (1997).
[4] C.L. Webber, J.P. Zbilut, J. Appl. Physiol. 76 (1994) 965.
[5] N. Thomasson, C.L. Webber, J.P. Zbilut, Int. J. Comput. Appl. 9 (2002) 1.
[6] J.P. Zbilut, N. Thomasson, C.I. Webber, Med. Engin. Phys. 24 (2002) 53.
[7] J. Kurths, U. Schwarz, C.P. Sonett, U. Parlitz, Nonlinear Processes Geophys. 1 (1994) 72.
[8] N. Marwan, M. Thiel, N.R. Nowaczyk, Nonlinear Processes Geophys. 9 (2002) 325.
[9] N. Marwan, J. Kurths, Phys. Lett. A 302 (5–6) (2002) 299.
[10] M. Thiel, M.C. Romano, J. Kurths, Chaos, in press.
[11] P. Faure, H. Korn, Physica D 122 (1998) 265.
[12] A. Rényi, Probability Theory, North-Holland, Amsterdam, 1970, Appendix.
[13] C. Beck, F. Schlöl, Thermodynamics of Chaotic Systems, Cambridge Univ. Press, 1993.
[14] M. Thiel, M.C. Romano, J. Kurths, R. Meucci, E. Allaria, T. Arecchi, Physica D 171 (3) (2002) 138.
[15] F. Takens, in: D.A. Rand, L.-S. Young (Eds.), Dynamical Systems and Turbulence, in: Lecture Notes in Mathematics, vol. 898, Springer, Berlin, 1980.
[16] M.B. Kennel, R. Brown, H.D.I. Abarbanel, Phys. Rev. A 45 (6) (1992) 3403.
[17] D. Broomhead, K. King, Physica D 20 (1986) 217.
[18] P. Grassberger, T. Schreiber, C. Schaffrath, Int. J. Bifur. Chaos 1 (3) (1991) 521.
[19] M. Thiel, M.C. Romano, J. Kurths, Appl. Nonlinear Dynamics, in press.
[20] O. Rössler, Phys. Lett. A 57 (1976) 397.