PHYSICAL REVIEW E VOLUME 62, NUMBER 1 JULY 2000

Surrogate data test for nonlinearity including nonmonotonic transforms

D. Kugiumtzis*
Department of Statistics, University of Glasgow, Glasgow G12 8QW, United Kingdom
(Received 13 January 2000)

It is shown that monotonicity of the transform in the surrogate data test, which diminishes the applicability of the test, is not necessary and concerns only the prominent algorithm of amplitude adjusted Fourier transform (AAFT) for surrogate data generation. The failure of AAFT under nonmonotonicity is explained and a modified algorithm appropriate for nonmonotonic transforms, called corrected AAFT (CAAFT), is proposed. The superiority of CAAFT over AAFT is demonstrated with simulated and real data and compared also to the iterated AAFT algorithm.

PACS number(s): 05.45.Tp, 05.10.Ln

Nonlinear analysis of time series, and especially methods based on chaos theory, have evidently contributed towards a better understanding of many physical phenomena observed often through a single quantity [1]. On the other hand, some classes of linear stochastic processes can fool the nonlinear methods to give estimates of low-dimensional determinism, nonlinearity, and chaos [2]. The surrogate data test has been invented in order to rule out this possibility before the arsenal of nonlinear methodology can be put forward [3,4].

The most general null hypothesis H_0 for the test is that the original scalar data $\boldsymbol{x}=[x_1, \ldots, x_n]'$ are generated by a Gaussian (normal) process measured through a static and possibly nonlinear transform h, i.e., $x_i=h(s_i)$, $i=1, \ldots, n$, and $\boldsymbol{s}=[s_1, \ldots, s_n]'$ the normal time series. To test H_0, M surrogate data representing H_0 are generated, a nonlinear method is applied, and if the derived estimate q_0 on the original data does not lie in the distribution of the estimates $q_1, \ldots, q_M$ on the surrogate data H_0 is rejected.

The monotonicity of h has been considered as an additional prerequisite for the implementation of the test and the prominent algorithm for generating surrogate data, the so-called amplitude adjusted Fourier transform (AAFT), is based on this assumption [3,5,6]. However, monotonicity of h cannot be asserted when dealing with real data. In this Rapid Communication, we argue against this constraint and propose a correction to the AAFT algorithm (CAAFT), to be used instead of AAFT for any transform h. So, through the use of CAAFT we establish the applicability of the test in real applications. The important concept in our approach is that for any time series, one can construct a normal time series, which under a monotonic transform possesses the autocorrelation and amplitude distribution of the given data.

According to H_0, a surrogate data set $\boldsymbol{z}=[z_1, \ldots, z_n]'$ must preserve the sample autocorrelation r of the original data, i.e., $r_x(\tau)=r_z(\tau)$, $\tau=1, \ldots, \tau_{\max}$, and the sample amplitude distribution, i.e., $F_x(x)=F_z(z)$, where $F_x(x)$ is the sample cumulative density function (CDF) of x_i. It was recently shown that AAFT does not fulfill the first condition when h is nonmonotonic and favors rejection of H_0 even when nonlinearity is not evident [7]. Thus, reported rejections using AAFT in numerous applications (e.g., see Ref. [8]) should be reexamined as suggested in [9].

The AAFT algorithm assumes that h is monotonic and first simulates h^{-1} (by reordering normal white noise to the rank order of $\boldsymbol{x}$, call it $\boldsymbol{y}$), then destroys any possible nonlinear dynamics (making phase randomization, call the derived data set $\boldsymbol{y}^{\mathrm{FT}}$) and finally simulates h (reordering the original data $\boldsymbol{x}$ to the rank order of $\boldsymbol{y}^{\mathrm{FT}}$) to get the surrogate data set $\boldsymbol{z}$.

Let us suppose first that the given time series is according to H_0 and h is monotonic. Then AAFT performs well because the approximation of h^{-1} and h in steps 1 and 3, respectively, is successful. If F_0 denotes the normal CDF (the same for s_i, y_i, and y_i^{FT}), the reordering in step 1 reads [10]

$$y_i=g^{-1}(x_i)=F_0^{-1}(F_x(x_i)), \tag{1}$$

and consequently g^{-1} coincides with h^{-1}, up to a normalization factor, due to the uniqueness of the inverse of a monotonic function h transforming a normal time series to the observed time series [11]. The reconstruction of h^{-1} in Eq. (1) gives a normal time series $\boldsymbol{y}$ with $r_y(\tau)=r_s(\tau)$ and the same holds for $\boldsymbol{y}^{\mathrm{FT}}$, so that $\boldsymbol{s}$, $\boldsymbol{y}$, and $\boldsymbol{y}^{\mathrm{FT}}$ are all realizations of the same normal process. Moreover, $r_s(\tau)$ is related to $r_x(\tau)$ through a function ϕ depending on g, i.e., $r_x(\tau)=\phi(r_s(\tau))$ [12].

Accordingly, the reordering in step 3 reads

$$z_i=g(y_i^{\mathrm{FT}})=F_x^{-1}(F_0(y_i^{\mathrm{FT}})), \tag{2}$$

and $h=g$, $r_z(\tau)=\phi(r_{y^{\mathrm{FT}}}(\tau))=\phi(r_s(\tau))$. Thus $r_z(\tau)=r_x(\tau)$ and the conditions of H_0 are satisfied.

Let us suppose now that h is nonmonotonic. In this case, g is different from h, g^{-1} is well defined as both g and g^{-1} are monotonic by construction, and $\boldsymbol{y}$ has still normal marginal distribution, but now its joint distribution is not in general normal. This can be verified formally using a test for joint normal distribution [13], or simply by testing whether an odd joint moment such as $\mu_{12}=\langle y_i, y_{i-\tau}^2 \rangle$ is different from zero. On the other hand, $\boldsymbol{y}^{\mathrm{FT}}$ has normal joint distribution by construction and has still $r_{y^{\mathrm{FT}}}(\tau)=r_y(\tau)$. Obviously, applying g to the components of $\boldsymbol{y}$ will give $\boldsymbol{x}$, but when g is applied to the components of $\boldsymbol{y}^{\mathrm{FT}}$ will give $\boldsymbol{z}$, but now $r_z(\tau)$

*Email address: dimitris@stats.gla.ac.uk

$\neq r_x(\tau)$, because the output of g^{-1} [$\boldsymbol{y}$ in Eq. (1)] and the input of g [$\boldsymbol{y}^{\mathrm{FT}}$ in Eq. (2)] are time series from different processes (non-normal and normal, respectively). So, the flaw in the AAFT algorithm when h is nonmonotonic is that, though $\boldsymbol{s}$ may be normal, a non-normal $\boldsymbol{y}$ (but with normal marginal distribution) is obtained in the first step and then the match of linear correlations fails when a normal $\boldsymbol{y}^{\mathrm{FT}}$ having the same autocorrelation as $\boldsymbol{y}$ is transformed back in step 3.

The assumption of a normal $\boldsymbol{s}$, treated so far, is consistent with the null hypothesis, but the examined time series may as well stem from a non-normal process, e.g., from a chaotic process. In this case and regardless of the monotonicity of h, the result after step 1 of the AAFT algorithm is the same as for the case of normal $\boldsymbol{s}$ and non-monotonic h: $\boldsymbol{y}$ has normal marginal distribution but not normal joint distribution.

We present now a correction of the AAFT algorithm, CAAFT, so that the original autocorrelation is matched even when $\boldsymbol{y}$ is not normal. Interpreting H_0 slightly differently than in AAFT, we want to find a normal time series $\boldsymbol{u}$ with autocorrelation r_u (different in general from r_s) such that through a transform like g it gives $\boldsymbol{w}$ and $r_x(\tau)=r_w(\tau)=\phi(r_u(\tau))$ for $\tau=1,\ldots,\tau_{\max}$, where ϕ depends on g. Note that basically the problem is to find the correct r_u. Then a normal time series $\boldsymbol{u}$ can be generated from r_u and the desired surrogate data $\boldsymbol{w}$ is directly derived through the g transform defined in Eq. (2).

Given r_x, the problem is formed as to find the approximate inverse functional relation to ϕ, call it ψ, where $\psi=\phi^{-1}$ if ϕ^{-1} exists, so that $r_u(\tau)=\psi(r_x(\tau))$. We have already established $r_z(\tau)=\phi(r_y(\tau))$, where $\boldsymbol{z}$ is the AAFT surrogate, and thus ψ can be approximated from the graph of $r_y(\tau)$ versus $r_z(\tau)$ for $\tau=0,\ldots,\tau_{\max}$. An estimate $\hat{\psi}$ can be obtained simply by linear interpolation, i.e., $\hat{\psi}$ is a piecewise linear function. Note that, though $\hat{\psi}$ is not monotonic by definition, our simulations hint at monotonic $\hat{\psi}$, so that $\psi=\phi^{-1}$ appears to hold approximately. We found linear interpolation superior to cubic spline interpolation, but other techniques may estimate $\hat{\psi}$ better.

Given $r_u(\tau)$ from $r_u(\tau)=\hat{\psi}(r_x(\tau))$, the coefficients of the corresponding autoregressive (AR) model can be found from the Yule-Walker equations, solved efficiently through the Levinson algorithm [14]. The AR model can be stabilized first, if any roots of the characteristic polynomial lie out of the unit circle, and then be used to generate $\boldsymbol{u}$.

In practice, the accuracy in $r_x(\tau)\simeq r_w(\tau)$ depends heavily on the quality of the approximation of ψ. This, in turn, does not seem to depend so much on the fitting method as on the sparsity of the data pairs $(r_z(\tau),r_y(\tau))$. Increasing $\tau_{\max}$ beyond lags of significant autocorrelation does not really enhance the approximation as the additional data pairs are ordered at the same range of (r_z,r_y) values. So, a good choice for $\tau_{\max}$ is at the range of lags for which r_x levels off to zero. To maintain good accuracy in the autocorrelation, we propose to run the algorithm a number of times K and pick out the r_u that gives r_w closest to r_x [15]. Then we use the corresponding AR model to generate the M surrogate data sets. This approach is close to the so-called ''typical realization'' approach found to have less power in rejecting H_0 in [16], but this is not actually an undesired property for real applications [9].

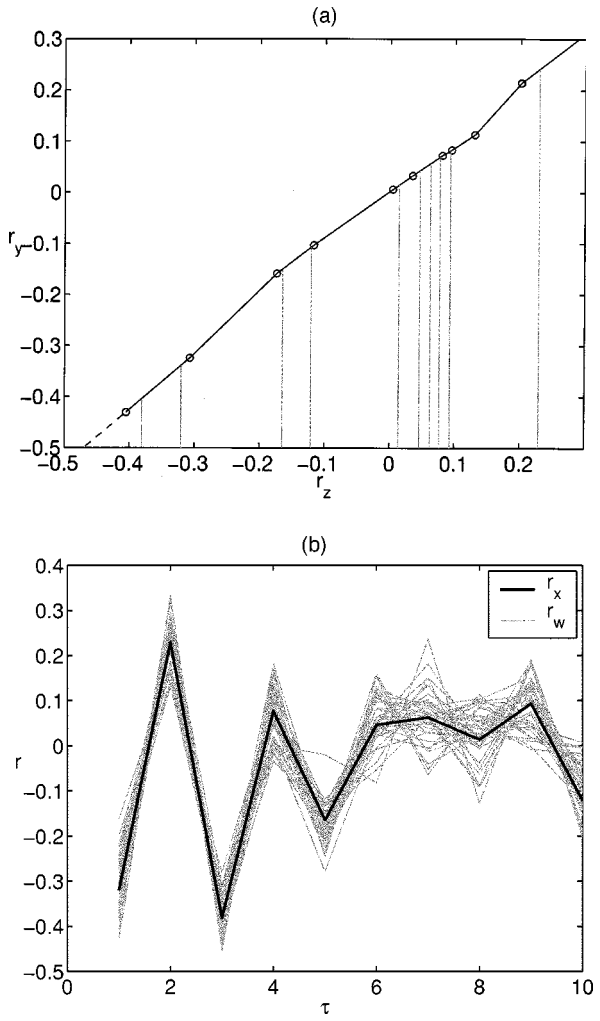

FIG. 1. (a) Linear interpolation $\hat{\psi}$ (solid line) of $(r_z(\tau),r_y(\tau))$, $\tau=0,\ldots,10$ time steps (open circles) computed on the Henon data [pair (1,1) for $\tau=0$ is not shown]. The mapping $r_u(\tau)=\hat{\psi}(r_x(\tau))$ is shown by a gray vertical line for each $r_x(\tau)$, $\tau=1,\ldots,10$. For extrapolation towards -1 (dashed line), $\hat{\psi}$ is arbitrarily defined as $r_u(\tau)=r_x(\tau)[r_y(\tau^*)/r_z(\tau^*)]$, where τ^* is the lag of the smallest r_z. (b) The autocorrelation for the original and the 40 surrogate data (computed in step 10), as denoted in the legend.

The steps of the complete algorithm CAAFT are as follows:

(1) Make the three steps of AAFT to generate $\boldsymbol{y}$, $\boldsymbol{y}^{\mathrm{FT}}$, and $\boldsymbol{z}$.

(2) Choose $\tau_{\max}$ and compute $r_z(\tau)$, $r_x(\tau)$ and $r_y(\tau)$ for $\tau=1,\ldots,\tau_{\max}$.

(3) Find the linear interpolation $\hat{\psi}$ of $r_y(\tau)$ as a function of $r_z(\tau)$.

(4) Compute $r_u(\tau)$ from $r_u(\tau)=\hat{\psi}(r_x(\tau))$.

(5) Choose a suitable order p for AR (e.g., $p=\tau_{\max}$), and estimate the coefficients of AR(p) from r_u.

(6) Generate $\boldsymbol{u}$ using the stabilized AR model.

(7) Transform $\boldsymbol{u}$ to $\boldsymbol{w}$ $[w_i=F_x^{-1}(F_0(u_i)),\ i=1,\ldots,n]$.

(8) Repeat the steps 1–7 K times to generate $\boldsymbol{w}^1,\ldots,\boldsymbol{w}^K$.

(9) Compute $\{r_w^1,\ldots,r_w^K\}$ and find the one, r_w^k, closest to r_x.

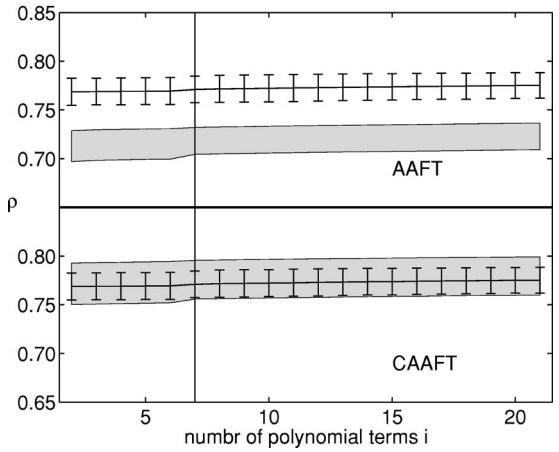

FIG. 2. The correlation coefficient ρ of the fit with Volterra polynomials ($m=5$) as a function of the polynomial terms i, for the square transformed AR(1) model [$s_{i+1}=0.3+0.8s_i+e_i$, where $e_i \sim \mathcal{N}(0,1-0.8^2)$ and white] and its AAFT and CAAFT surrogates (upper and lower panel, respectively). The average ρ from 100 realizations of 2048 samples from the AR(1) model is shown as a black line and the corresponding SD as error bars. The gray zones denote the average diversity of ρ for the surrogate data (first the mean and SD of ρ from the 40 surrogate data for each realization is computed and then averaged over the 100 realizations; these averaged mean and SD form the gray area). The vertical line distinguishes linear from nonlinear polynomial terms.

(10) Use the AR model that corresponds to the selected k trial and generate M surrogate data by repeating the steps 6 and 7 M times.

To elucidate, the algorithm is illustrated for a time series of 512 samples from the Henon map, for which the AAFT algorithm generates surrogate data with biased autocorrelation, e.g., $r_x(1)=-0.32$ and $\langle r_z(1)\rangle=-0.41$ with standard deviation (SD) 0.02 for 40 AAFT surrogates. In one replication of the CAAFT algorithm, the desired r_u is approximated using the linear interpolation of the pairs $(r_z(\tau),r_y(\tau))$, as shown in Fig. 1(a). Repeating steps 1–7 K times, K candidate stabilized AR models (here $K=5$ and $p=\tau_{\max}=10$) are designed from the respective r_u, and the one giving rise to the surrogate $\boldsymbol{w}$ with the closest autocorrelation to the original is selected (see [15]). Based on this AR model, M surrogates are generated and, as shown in Fig. 1(b), their autocorrelations match well the original one with somehow larger variance than for the AAFT case, e.g., $\langle r_w(1)\rangle=-0.31$ with SD 0.06.

Note that the computational load depends mainly on K. Overall, CAAFT is not more computationally intensive than AAFT when $K<M$. For $K=M=40$, used in the examples below, CAAFT is only about twice slower than AAFT.

We applied the CAAFT algorithm to different known systems (linear stochastic and chaotic) and different transforms h (monotonic and nonmonotonic) and confirmed that it always preserves the original autocorrelation with some variance, which depends on the data size.

In Fig. 2, we compare AAFT and CAAFT for the square of an AR(1) process using as discriminating statistics for the test the correlation coefficient ρ of the fit with Volterra polynomials [i.e., polynomials on $(x_i,\ldots,x_{i-(m-1)})$ of a maximum degree d, here $d=2$], in order to view differences in both the linear and nonlinear correlations (the polynomials of the first $m+1$ terms are linear) [7,17]. Obviously, the AAFT surrogate data mismatch the original linear correlations, but the CAAFT surrogates do not. We found for both the linear and nonlinear polynomials that H_0 was rejected always for AAFT but not for CAAFT, using either the significance S ($S>1.96$ suggests rejection at the 95% confidence level) or the rank ordering.

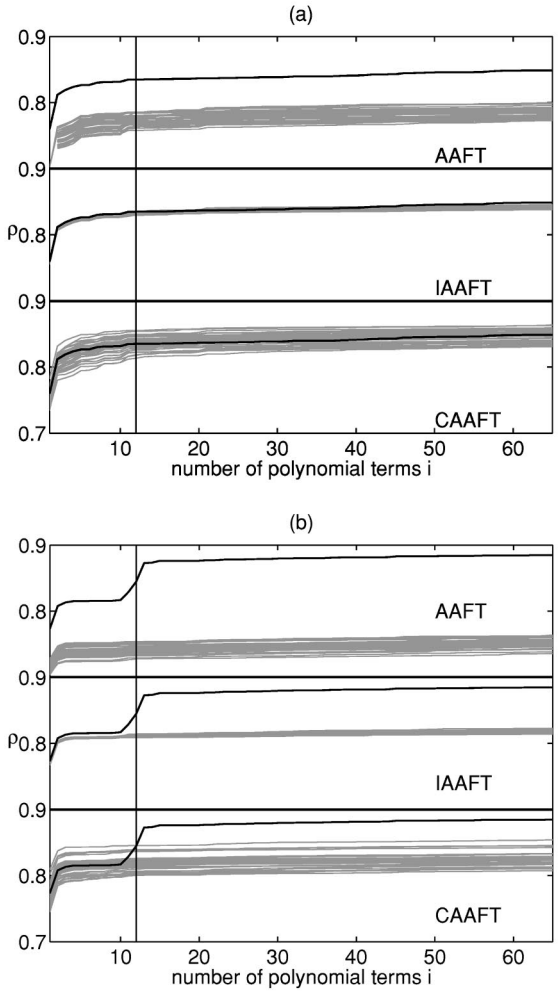

FIG. 3. (a) The correlation coefficient ρ of the fit with Volterra polynomials ($m=10$) as a function of the polynomial terms i, for the first EEG data set (black line) and its AAFT, IAAFT, and CAAFT surrogates (gray lines in the upper, middle, and lower panels, respectively). The vertical line distinguishes linear from nonlinear polynomial terms. (b) The same for the second EEG data set.

We apply the test to two electroencephalogram (EEG) data sets under different physiological conditions, i.e., an EEG recording (sampling time $\tau_s=0.01$ s, $n=2048$) many hours before an epileptic seizure of a human subject, which is believed to be high-dimensional and thus hard to detect nonlinearity [9], and another during an epileptic seizure ($\tau_s=0.01$ s, $n=1700$), which was reported to be low-dimensional and nonlinear [18]. We use the statistics from the Volterra polynomials, AAFT and CAAFT surrogates, as well as surrogates generated by the iterated AAFT algorithm (IAAFT) [4], which is supposed to match well the original linear autocorrelations but gives little insight into the process

that generates the surrogate data. For the first data set, no nonlinearity is observed from the fit, as shown in Fig. 3(a). This is confirmed with CAAFT as the fit for the surrogates is equally good. On the other hand, the nonlinear fit for the AAFT surrogates is clearly worse suggesting the rejection of H_0, erroneously as the same discrimination holds for the linear fit. In the case of IAAFT, H_0 is also rejected for some polynomials (for all linear and the last 20 nonlinear polynomials, $S>2$ was found), due to the small variance and the slight bias in the linear fit. For the epileptic EEG evidence of nonlinearity is implied by the improvement of the fit with the inclusion of the first couple of nonlinear polynomial terms, as shown in Fig. 3(b). The increase in ρ for the original data is discriminated clearly from all three types of surrogates, but at different levels, determined by the bias in the linear correlations (large for AAFT, small for IAAFT, and none for CAAFT), and the variance (particularly small for IAAFT). For AAFT, the problem is that H_0 is rejected at about the same large confidence level using linear and nonlinear polynomials, while for CAAFT rejections are properly obtained only with the nonlinear fit.

In conclusion, we have established that monotonicity of the transform is not required for the implementation of the surrogate data test for nonlinearity and we proposed a correction to the AAFT algorithm (CAAFT) to generate proper surrogate data. The rationale with CAAFT is to circumvent the problem of nonmonotonicity by finding a suitable normal process, which under a monotonic transform gives surrogate data with approximately the same autocorrelation as the original one, even if the original data can be seen as nonmonotonic transform of a normal process. The approximation of the original autocorrelation is unbiased, so that the test with CAAFT is more accurate than with AAFT. The variance in this approximation is often as large as for AAFT and always larger than for IAAFT. Therefore, the test with CAAFT turns out to be more conservative than with IAAFT. However, this should not be considered as a drawback of the CAAFT algorithm, but rather as a welcome property, preventing immature rejections of H_0. On the other hand, if the time series passes the test with CAAFT the evidence for nonlinearity is more reliable than when using AAFT or IAAFT surrogates.

The author thanks Pål Larsson for providing the EEG data and Holger Kantz for his valuable comments.

[1] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis* (Cambridge University Press, Cambridge, England, 1997).
[2] A. R. Osborne and A. Provenzale, Physica D **35**, 357 (1989).
[3] J. Theiler *et al.*, Physica D **58**, 77 (1992).
[4] T. Schreiber and A. Schmitz, Phys. Rev. Lett. **77**, 635 (1996).
[5] T. Schreiber, Phys. Rev. Lett. **80**, 2105 (1998).
[6] T. Schreiber and A. Schmitz, Physica D (to be published).
[7] D. Kugiumtzis, Phys. Rev. E **60**, 2808 (1999).
[8] K. P. Yip, D. J. Marsh, and N. H. Holstein-Rathlou, Physica D **80**, 95 (1995); C. J. Stam *et al.*, Biol. Cybern. **77**, 247 (1997); R. B. Govindan, K. Narayanan, and M. S. Gopinathan, Chaos **8**, 495 (1998).
[9] D. Kugiumtzis, in *Proceedings of the Workshop ''Chaos in Brain?''*, edited by K. Lehnertz *et al.* (World Scientific, Singapore, 2000), p. 230.
[10] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd ed. (McGraw-Hill, New York, 1991), p. 113.
[11] The normalization factor here means that under normalization $\boldsymbol{s}$ and $\boldsymbol{y}$ are identical if n is infinite, or ϵ-close with ϵ decreasing as n increases.
[12] N. Johnson and S. Kotz, *Distributions in Statistics, Continuous Multivariate Distributions* (John Wiley and Sons, New York, 1972), p. 15.
[13] K. V. Mardia and J. T. Kent, Biometrika **78**, 355 (1991); S. W. Looney, Am. Stat. **49**, 64 (1995).
[14] S. M. Kay, *Modern Spectral Estimation: Theory and Applications* (Prentice Hall, New Jersey, 1988).
[15] We found that a robust way to achieve this is to compute the error $\Sigma_{i=1}^{\tau}(r_x(i)-r_w^j(i))^2$ for $j=1,\ldots,K$ and $\tau=1,\ldots,\tau_{\max}$ and then select the trial k that gives the minimum error most times, where $\tau_{\max}$ minimums are totally computed, each time over the K trials.
[16] J. Theiler and D. Prichard, Physica D **94**, 221 (1996).
[17] B. Barahona and Ch.-S. Poon, Nature (London) **6579**, 215 (1996).
[18] D. Kugiumtzis, Physica D **95**, 13 (1996).