Smooth Local Subspace Projection for Nonlinear Noise Reduction

David Chelidze
University of Rhode Island, chelidze@uri.edu

Smooth Local Subspace Projection for Nonlinear Noise Reduction

Smooth Local Subspace Projection for Nonlinear Noise Reduction

David Chelidze[1, a)]

Department of Mechanical, Industrial & Systems Engineering

University of Rhode Island, Kingston, RI 02881

Many nonlinear or chaotic time series exhibit an innate broad spectrum, which makes noise reduction difficult. Local projective noise reduction is one of the most effective tools. It is based on proper orthogonal decomposition (POD), and works for both map-like and continuously sampled time series. However, POD only looks at geometrical or topological properties of data and does not take into account the temporal characteristics of time series. Here we present a new smooth projective noise reduction method. It uses smooth orthogonal decomposition (SOD) of bundles of reconstructed short-time trajectory strands to identify smooth local subspaces. Restricting trajectories to these subspaces imposes temporal smoothness on the filtered time series. It is shown that SOD-based noise reduction significantly outperforms the POD-based method for continuously sampled noisy time series.

Keywords: nonlinear noise reduction, smooth orthogonal decomposition, projective noise reduction

---

[a)]chelidze@mail.uri.edu; mcise.uri.edu/chelidze

Noise reduction in nonlinear or chaotic time series is problematic due to the inherent broad spectrum of the deterministic component in a signal. Projective noise reduction based on proper orthogonal decomposition has been shown to be effective for low noise levels. Here, we study a generalization of the local projective noise reduction method based on smooth orthogonal decomposition, which accounts for both topological and temporal characteristics of the time series. Through the analysis of synthetic chaotic time series contaminated by various levels of noise, we show that our method significantly outperforms the original and works even for low signal-to-noise ratios. Thus, it provides an effective tool for noise reduction in continuously sampled chaotic time series.

I. INTRODUCTION

Many natural and engineered systems generate nonlinear deterministic time series that are contaminated by random measurement/dynamical noise. Chaotic time series have inherently broad spectra and are not amenable to conventional spectral noise filtering. Two main methods for filtering noisy chaotic time series[1–3] are: (1) model based filtering[4], and (2) projective noise reduction[2]. While model based reduction has some merit for systems with known analytical models[4], projective noise reduction provides a superior alternative[2] for experimental data. There are other methods like adaptive filtering or the wavelet shrinkage[5] method that can work in certain cases but require an opportune selection of parameters or some *a priori* knowledge of the system or the noise mechanisms[6].

In this paper, we describe a new method for nonlinear noise reduction that is based on a smooth local subspace identification[7,8] in the reconstructed phase space[9,10]. In contrast to identifying the tangent subspace of an attractor using proper orthogonal decomposition (POD) of a collection of nearest neighbor points (i.e., as in local projective noise reduction scheme), we identify a smooth subspace that locally embeds the attractor using smooth orthogonal decomposition (SOD) of bundle of nearest neighbor trajectory strands. This new method accounts for not only geometrical information in the data, but its temporal

2

characteristics too. As we demonstrate later, this dramatically increases the ability to filter low signal-to-noise-ratio (SNR) time series.

In the following section ideas behind nonlinear noise reduction are discussed, followed by the description of the SOD-based algorithm. Models generating test time series are introduced next, along with the methods used in the algorithm evaluation, which is followed by the description of results and discussion.

II. SMOOTH PROJECTIVE NOISE REDUCTION

Both POD-based local projective noise reduction[2], and SOD-based smooth projective noise reduction work in the reconstructed phase space[9] of the system generating the noisy time series. The basic idea is that, at any point on the attractor, noise leaks out into higher dimensions than the dimension of the attractor's tangent space at that point. Thus, by embedding data into the higher d-dimensional space and then projecting it down to the tangent subspace of k-dimensions reduces the noise in the data. The differences between the methods are in the way this tangent space is identified.

For both noise reduction methods, the noisy time series $\{x_i\}_{i=1}^{n}$ is embedded into a d-dimensional global phase space using delay coordinate embedding generating the vector-valued trajectory $\{\mathbf{y}_i\}_{i=1}^{n-(d-1)\tau}$:

$$\mathbf{y}_i = \begin{bmatrix} x_i, \ x_{i+\tau}, \ \ldots \ x_{i+(d-1)\tau} \end{bmatrix}^T, \tag{1}$$

where τ is the delay time, which is usually determined using the first minimum of the average mutual information[18] for the time series, by some other nonlinear correlation statistic[17], or by visual inspection.

The selection of the global embedding dimension d, and the dimension of the tangent subspace k are important steps in both noise reduction schemes. For a noise-free time series, a method of false nearest neighbors[11,12] is usually used to estimate the minimum *necessary* embedding dimension D. However, the efficacy of this method is degraded by the presence of noise[13]. Alternatively, the embedding theorem provides[9] a measure of minimum *sufficient* embedding dimension which should be greater than twice the *fractal dimension*[14] of the attractor. The most commonly used estimate of the fractal dimension is a correlation

dimension[15], which is also not very robust with respect to noise. However, for moderate noise levels, one can still get rough approximation to the correlation dimension[16], which could serve as the upper bound on the projection dimension. In practical applications, the use of methods that are robust with respect to noise (e.g., the minimum description length principle[13] or continuity statistic[17]) are more appropriate. In this paper, however, we use synthetic time series which are artificially contaminated with additive noise. Therefore, the size of global embedding dimension is taken to be a slightly greater than the minimum required embedding dimension estimated using the clean time series and the false nearest neighbors method.

A.   Local Projection Subspace

In the local projective noise algorithm[19], for each point $\mathbf{y}_i$ in the reconstructed d-dimensional phase space, r temporally uncorrelated nearest neighbor points $\{\mathbf{y}_i^j\}_{j=1}^r$ are determined. The POD is applied to this cloud of nearest neighbor points, and the optimal k-dimensional projection of the cloud is obtained providing the needed filtered $\bar{\mathbf{y}}_i$ (which is also adjusted to account for the shift in the projection due to the trajectory curvature at $\mathbf{y}_i$[3]). The filtered time series $\{\bar{x}_i\}_{i=1}^n$ is obtained by averaging the appropriate coordinates in the adjusted vector-valued time series $\{\bar{\mathbf{y}}_i\}_{i=1}^{n-(d-1)\tau}$.

In contrast, smooth projective noise reduction works with short strands of the reconstructed phase space trajectory. These strands are composed of $(2l+1)$ time-consecutive reconstructed phase space points, where l is a small natural number. Namely, each reconstructed point $\mathbf{y}_k$ has an associated strand $\mathbf{s}_k = [\mathbf{y}_{k-l}, \dots, \mathbf{y}_{k+l}]$, and an associated bundle of r nearest neighbor strands $\{\mathbf{s}_k^j\}_{j=1}^r$, including the original. This bundle is formed by finding $(r-1)$ nearest neighbor points $\{\mathbf{y}_k^j\}_{j=1}^{r-1}$ for $\mathbf{y}_k$ and the corresponding strands. SOD is applied to each of these strands in the bundle and the corresponding k-dimensional smoothest approximations to the strands $\{\tilde{\mathbf{s}}_k^j\}_{j=1}^r$ are obtained. The filtered $\bar{\mathbf{y}}_k$ point is determined by the weighted average of points $\{\tilde{\mathbf{y}}_k^j\}_{j=1}^r$ in the smoothed strands of the bundle. Finally, the filtered time series $\{\bar{x}_i\}_{i=1}^n$ are obtained just as in the local projective noise reduction. The procedure can be applied repeatedly for further smoothing or filtering. For completeness,

4

the details of SOD and smooth subspace approximation are given in the following sections.

B.   Smooth Orthogonal Decomposition

The objective of POD is to obtain the best low-dimensional approximate description of high-dimensional data in a least squares sense[20–22]. In contrast, SOD is aimed at obtaining the smoothest low-dimensional approximations of high-dimensional dynamical processes[7,23,24]. While POD only focuses on the spatial characteristics of the data, SOD considers both its temporal and spatial characteristics.

Consider a case where a field, y, is defined for some d state variables $\{x_i\}_{i=1}^d$ (e.g., a system of d ordinary differential equations, d sensor measurements from some system, etc.). Assume that this field is sampled at exactly n instants of time (e.g., n sets of simultaneous measurements at d locations). We arrange this data into a $n \times d$ matrix $\mathbf{Y}$, such that element Y_{ij} is the sample taken at the ith time instant from the jth state variable. We also assume that each column of $\mathbf{Y}$ has zero mean (or, alternatively, we subtract the mean value from each column). We are looking for a linear coordinate transformation of this matrix $\mathbf{Y}$:

$$\mathbf{Q} = \mathbf{Y}\boldsymbol{\Psi},\tag{2}$$

where the columns of $\mathbf{Q} \in \mathbb{R}^{n \times d}$ are new *smooth orthogonal coordinates* (SOCs) and *smooth projective modes* (SPMs) are given by the columns of $\boldsymbol{\Psi} \in \mathbb{R}^{d \times d}$. In this transformation, $\mathbf{Q}$ should contain time coordinates sorted by their roughness.

The SOD is accomplished by the following generalized eigenvalue problem:[36]

$$\boldsymbol{\Sigma}\,\psi_i = \lambda_i\,\dot{\boldsymbol{\Sigma}}\,\psi_i,\tag{3}$$

where $\boldsymbol{\Sigma} = \frac{1}{n}\mathbf{Y}^T\mathbf{Y} \in \mathbb{R}^{d \times d}$ and $\dot{\boldsymbol{\Sigma}} = \frac{1}{n}\dot{\mathbf{Y}}^T\dot{\mathbf{Y}} \in \mathbb{R}^{d \times d}$ are auto-covariance matrices for d states and their time derivatives, respectively. Eigenvalues λ_i are called *smooth orthogonal values* (SOVs), and $\psi_i \in \mathbb{R}^d$ are individual SPMs $(i = 1, \ldots, d)$. The time derivative of the matrix $\mathbf{Y}$ is either known analytically or can be derived numerically $\dot{\mathbf{Y}} = \mathbf{D}\mathbf{Y}$, where $\mathbf{D}$ is some finite difference differential operator.

5

C. Smooth Subspace Identification and Data Projection

Equation (3) can be solved using a *generalized singular value decomposition* of the matrix pair $\mathbf{Y}$ and $\dot{\mathbf{Y}}$:

$$\mathbf{Y} = \mathbf{UC\Phi}^T , \quad \dot{\mathbf{Y}} = \mathbf{VS\Phi}^T , \quad \mathbf{C}^T\mathbf{C} + \mathbf{S}^T\mathbf{S} = \mathbf{I} , \tag{4}$$

where *smooth orthogonal modes* (SOMs) are given by the columns of $\mathbf{\Phi} \in \mathbb{R}^{d \times d}$, SOCs are given by the columns of $\mathbf{Q} = \mathbf{UC} \in \mathbb{R}^{n \times d}$ and SOVs are given by the term-by-term division of $\mathrm{diag}(\mathbf{C}^T\mathbf{C})$ and $\mathrm{diag}(\mathbf{S}^T\mathbf{S})$. The resulting SPMs are the columns of the inverse of the transpose of SOMs: $\mathbf{\Psi}^{-1} = \mathbf{\Phi}^T \in \mathbb{R}^{d \times d}$. In this paper, we will assume that the SOVs are arranged in descending order $(\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d)$. The magnitude of the SOVs quadratically correlates with the smoothness of the SOCs[7]. Please note that if the identity matrix is substituted for $\dot{\mathbf{\Sigma}}$ in Eq. (3), it reduces to the POD eigenvalue problem that can be solved by the singular value decomposition of the matrix $\mathbf{Y}$.

The smoothest k-dimensional approximation to $\mathbf{Y}$ $(k < d)$ can be obtained by retaining only k columns of interest in the matrices $\mathbf{U}$ and $\mathbf{\Phi}$, and reducing $\mathbf{C}$ to the corresponding $k \times k$ minor. In effect, we are looking for a projection of $\mathbf{Y}$ onto a k-dimensional smoothest subspace. The embedding of this k-dimensional projection into d-dimensional space $(\bar{\mathbf{Y}})$ is constructed using the corresponding reduced matrices: $\bar{\mathbf{U}} \in \mathbb{R}^{n \times k}$, $\bar{\mathbf{C}} \in \mathbb{R}^{k \times k}$, and $\bar{\mathbf{\Phi}} \in \mathbb{R}^{d \times k}$, as follows

$$\bar{\mathbf{Y}} = \bar{\mathbf{U}}\bar{\mathbf{C}}\bar{\mathbf{\Phi}}^T . \tag{5}$$

D. Data Padding to Mitigate the Edge Effects

SOD-based noise reduction is working on $(2l+1)$-long trajectory strands. Therefore, for the first and last l points in the matrix $\mathbf{Y}$, we will not have full-length strands for calculations. In addition, the delay coordinate embedding procedure reconstructs the original n-long time series into $\mathbf{Y}$ which has only $[n-(d-1)\tau]$-long columns. Therefore, the first and last $(d-1)\tau$ points in $\{x_i\}_{i=1}^n$ will not have all d components represented in $\mathbf{Y}$, while other points will have their counterparts in each column of $\mathbf{Y}$. To deal with these truncations, which cause unwanted edge effects in noise reduction, we pad both ends of $\mathbf{Y}$ by $[l + (d-1)\tau]$-long
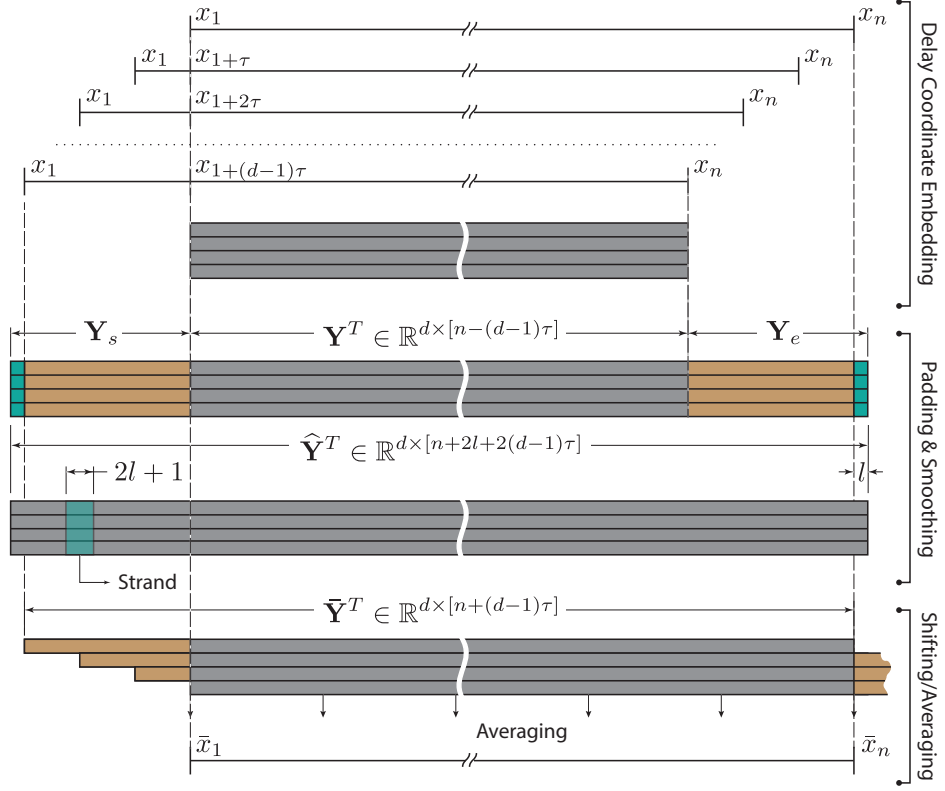
FIG. 1. Schematic illustration of smooth projective noise reduction algorithm

trajectory segments. These trajectory segments are identified by finding the nearest neighbor points to the starting and the end points inside $\mathbf{Y}$ itself (e.g., $\mathbf{y}_s$ and $\mathbf{y}_e$, respectively). Then, the corresponding trajectory segments are extracted from $\mathbf{Y}$: $\mathbf{Y}_s = [\mathbf{y}_{s-l-(d-1)\tau}, \ldots, \mathbf{y}_{s-1}]^T$ and $\mathbf{Y}_e = [\mathbf{y}_{e+1}, \ldots, \mathbf{y}_{e+l+(d-1)\tau}]^T$, and are used to form a padded matrix $\widehat{\mathbf{Y}} = [\mathbf{Y}_s; \mathbf{Y}; \mathbf{Y}_e]$. Now, the procedure can be applied to all points in $\widehat{\mathbf{Y}}$ starting at $l+1$ and ending at $n+l$ points.

III. SMOOTH NOISE REDUCTION ALGORITHM

Smooth noise reduction algorithm is schematically illustrated in Fig. 1. While we usually work with data in column form, in this figure—for illustrative purpose—we arranged arrays in row form. The particular details are explained as follows:

1. Delay Coordinate Embedding

   (a) Estimate the appropriate delay time τ and the minimum *necessary* embedding

7

dimension D for the time series $\{x_i\}_{i=1}^{n}$.

(b) Determine the local (i.e., projection), $k \leq D$, and the global (i.e., embedding), $d \geq D$, dimensions for trajectory strands.

(c) Embed the time series $\{x_i\}_{i=1}^{n}$ using global embedding parameters (τ, d) into the reconstructed phase space trajectory $\mathbf{Y} \in \mathbb{R}^{(n-(d-1)\tau) \times d}$.

(d) Partition the embedded points in $\mathbf{Y}$ into a kd-tree for fast searching.

2. Padding and Filtering:

(a) Pad the end points of $\mathbf{Y}$ to have appropriate strands for the edge points, which results in $\widehat{\mathbf{Y}} \in \mathbb{R}^{[n+2l+2(d-1)\tau] \times d}$.

(b) For each point $\{\hat{\mathbf{y}}_i\}_{i=l+1}^{n+l}$ construct a $(2l+1)$-long trajectory strand $\mathbf{s}_i^1 = [\hat{\mathbf{y}}_{i-l}; \ldots; \hat{\mathbf{y}}_{i+l}] \in \mathbb{R}^{(2l+1) \times d}$.

(c) For the same points $\hat{\mathbf{y}}_i$, look up $(r-1)$ nearest neighbor points that are temporarily uncorrelated, and the corresponding nearest neighbor strands $\{\mathbf{s}_i^j\}_{j=2}^{r}$.

(d) Apply SOD to each of the r strands in this bundle, and obtain the corresponding k-dimensional smooth approximations to all d-dimensional strands in the bundle $\{\tilde{\mathbf{s}}^j\}_{j=1}^{r}$.

(e) Approximate the base strand by taking the weighted average of all these smoothed strands $\bar{\mathbf{s}}_i = \langle \tilde{\mathbf{s}}_i^j \rangle_j$, using weighting that diminishes contribution with the increase in the distance from the base strand.

3. Shifting and Averaging:

(a) Replace the points $\{\hat{\mathbf{y}}_i\}_{i=1+l}^{n+l}$ at the center of each base strand by its approximation $\bar{\mathbf{y}}_i$ determined above to form $\bar{\mathbf{Y}} \in \mathbb{R}^{[n+(d-1)\tau] \times d}$.

(b) Average each d smooth adjustment to each point in the time series to estimate the filtered point:
$$\bar{x}_i = \frac{1}{d} \sum_{k=1}^{d} \bar{\mathbf{Y}}(k, i + (k-1)\tau).$$

4. Repeat the first three steps until data is smoothed out, or some preset criterion is met.

8

IV. EVALUATING THE ALGORITHM

In this section we evaluate the performance of the algorithm by testing it on a time series generated by Lorenz model[29] and a double-well Duffing oscillator[28]. The Lorenz model used to generate chaotic time series is:

$$\dot{x} = -\frac{8}{3}x + yz, \quad \dot{y} = -10\left(y - z\right), \quad \text{and} \quad \dot{z} = -xy + 28y - z. \tag{6}$$

The chaotic signal used in the evaluation is obtained using the following initial conditions $(x_0, y_0, z_0) = (20, 5, -5)$, and the total of 60,000 points were recorded using 0.01 sampling time period. The double-well Duffing equation used is:

$$\ddot{x} + 0.25\,\dot{x} - x + x^3 = 0.3\cos t. \tag{7}$$

The steady state chaotic response of this system was sampled thirty times per forcing period and a total of 60,000 points were recorded to test the noise reduction algorithms.

In addition, a 60,000 point, normally distributed, random signal was generated, and was mixed with the chaotic signals in 1/20, 1/10, 1/5, 2/5, and 4/5 standard deviation ratios, which respectively corresponds to 5, 10, 20, 40, and 80 % noise in the signal or SNRs of 26.02, 20, 13.98, 7.96, and 1.94 dB. This results in total of five noisy time series for each of the models. The POD-based projective and SOD-based smooth noise reduction procedures were applied to all noise signals using total of ten iterations.

To evaluate the performance of the algorithms, we used several metrics that include improvements in SNRs and power spectrum, estimates of the correlation sum and short-time trajectory divergence rates (used for estimating the correlation dimension and short-time largest Lyapunov exponent, respectively[30]). In addition, phase portraits were examined to gain qualitative appreciation of noise reduction effects.

The largest Lyapunov exponent λ_1 [31] characterizes the exponential growth of the distance between two points on the attractor that are very close initially. If this small initial distance between points is δ_0 and the distance at a later discrete time n is denoted as δ_n, then $\delta_n \sim \delta_0 \exp\left(\lambda_1 t\right)$ as $\delta_0 \to 0$. Here we used a modified algorithm by Wolf *et al.*[32,33], where for a set of reconstructed points $\mathbf{y}_i$ on their fiducial trajectory, we look up the r temporarily

uncorrelated nearest neighbor points and track their average divergence from the fiducial trajectory.

The correlation dimension characterizes the cumulative distribution of distances on the attractor[15,34,35], and indicates the lower bound on the minimum number of variables needed to uniquely describe the dynamics on the attractor. Here, we used the modified algorithm described in[15], where the correlation sum is estimated for the different length scales ϵ as:

$$C(\epsilon) = \frac{2}{(N-s)(N-s-1)} \sum_{i=1}^{N} \sum_{j=i+s+1}^{N} \Theta\left(\epsilon - \|\mathbf{y}_i - \mathbf{y}_j\|\right) \sim \epsilon^{D_2}, \tag{8}$$

where Θ is a Heavyside step function, N is the total number of points, s is the interval needed to remove temporal correlations, and D_2 is the correlation dimension.

V. RESULTS

A. Lorenz Model Based Time Series

Using average mutual information estimated from the clean Lorenz time series, a delay time of 12 sampling time periods is determined. The false nearest neighbors algorithm (see Fig. 2) indicates that the attractor is embedded in $D = 4$ dimensions for teh noise-free time series. Even for the noisy time series the trends show clear qualitative change around four or five dimensions. Therefore, six is used for global embedding dimension ($d = 6$), and $k = 3$ is used as local projection dimension. The reconstructed phase portraits for the clean Lorenz signal and the added random noise are shown in Fig. 3 (a) and (b), respectively. The corresponding power spectral densities for the clean and the noise-added signals are shown in Fig. 3 (c). The phase portraits of POD- and SOD-filtered signals with 5% noise are shown in Fig. 3 (d) and (e), respectively. Fig. 3 (f) shows the corresponding power spectral densities. In all these and the following figures, 64 nearest neighbor points are used for POD-based algorithm, and bundles of 9 eleven-point-long trajectory strands are used for SOD-based algorithm. The filtered data shown is for 10 successive applications of the noise reduction algorithms. The decrease in the noise floor after filtering is considerably more dramatic for the SOD algorithm when compared to the POD.
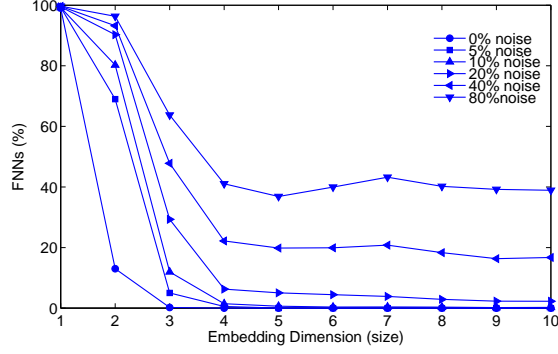
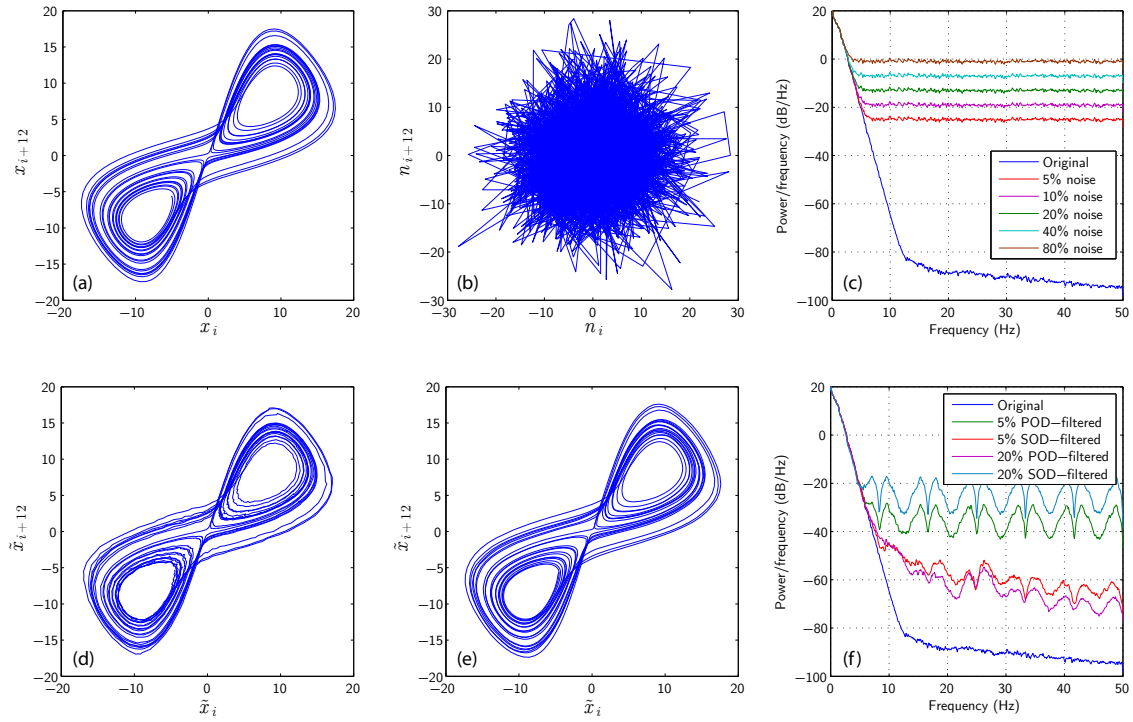FIG. 2. False nearest neighbor algorithm results for Lorenz time series

FIG. 3. Reconstructed phase portrait from Lorenz signal (a); random noise phase portrait (b); power spectrum of clean and noisy signals (c); POD-filtered phase portrait of 5% noise added data (d); SOD-filtered phase portrait of 5% noise added data (e); and the corresponding power spectrums (f).

The successive improvements in SNRs after each iteration of the algorithms are shown in Fig. 4, and the corresponding numerical values are listed in Table I for the 5% and 20% noise added signals. While the SNRs are monotonically increasing for the SOD algorithm after each successive application, they peak and then gradually decrease for the POD algorithm.
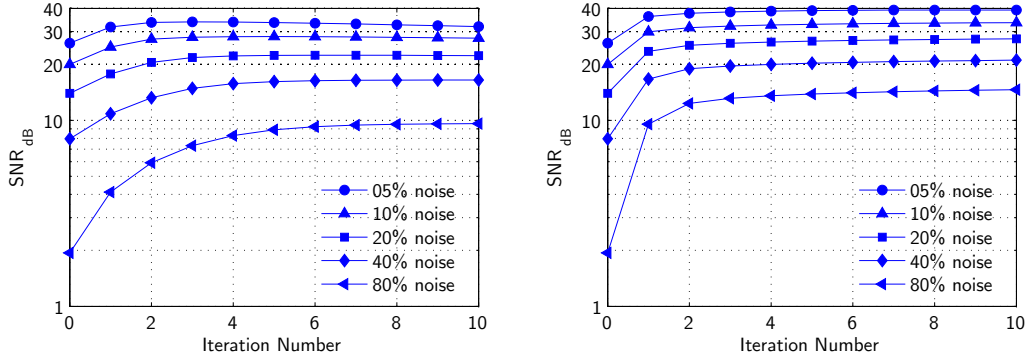
11

FIG. 4. Noisy Lorenz time series SNRs versus number of application of POD-based (left) and SOD-based (right) nonlinear noise reduction procedures

In addition, the rate of increase is considerably larger for the SOD algorithm compared to the POD, especially during the initial applications. As seen from the Table I, the SOD-based algorithm provides about $13\,\mathrm{dB}$ improvement in SNRs, while POD-based algorithm manages only $7 \sim 8\,\mathrm{dB}$ improvements.

The estimates of the correlation sum and short-time trajectory divergence for the noise-reduced data and the original clean data are shown in Fig. 5. For the correlation sum, both POD- and SOD-based algorithms show similar scaling regions and improve considerably on the noisy data. For the short-time divergence both methods manage to recover some of the scaling regions, but SOD does a better job at small scales, where POD algorithm yields large local fluctuations in the estimates.

The qualitative comparison of the phase portraits for both noise reduction methods are shown in Fig. 6. While POD does a decent job al low noise levels, it fails at higher noise

TABLE I. SNRs for the noisy time series from Lorenz model, and for noisy data filtered by POD- and SOD-based algorithms

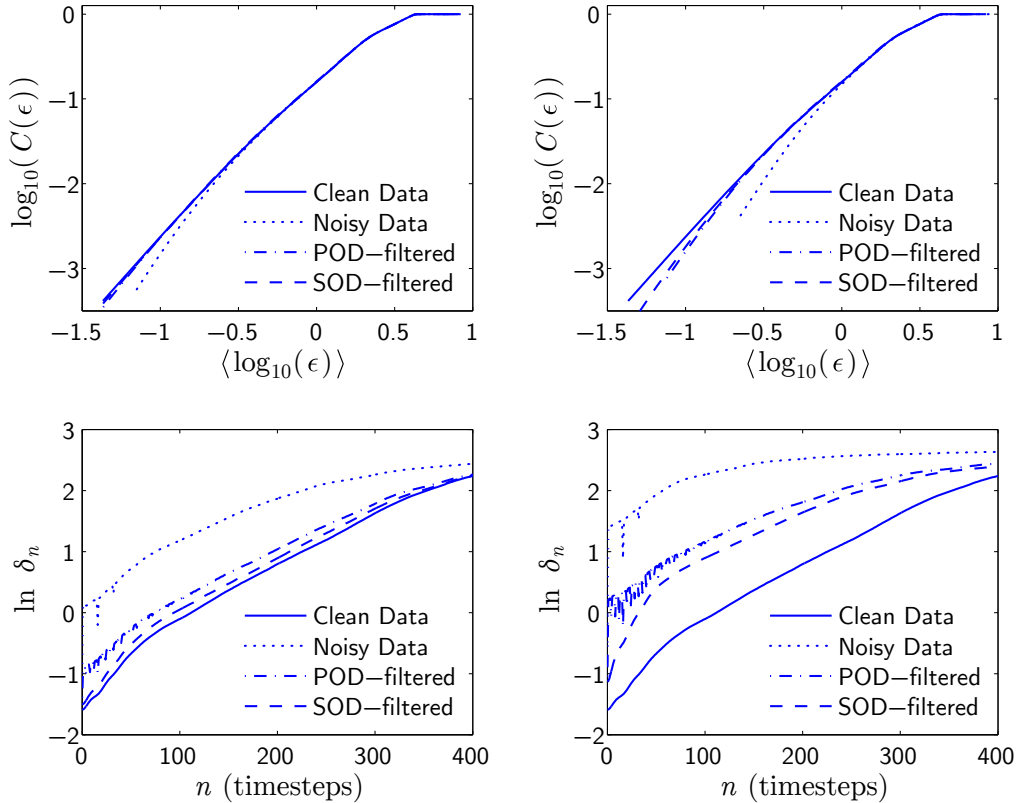| Signal	Lorenz + 5% Noise			Lorenz + 20% Noise		
Algorithm	None	POD	SOD	None	POD	SOD
SNR (dB)	26.0206	33.8305	39.1658	13.9794	22.3897	27.4318

FIG. 5. Correlation sum (top plots) and short-time divergence (bottom plots) for 5% (left plots) and 20% (right plots) noise contaminated Lorenz time series

levels where no deterministic structure is recovered at 80% noise level. In contrast, SOD provides smoother and cleaner phase portraits. Even at 80% noise level, the SOD algorithm is able to recover large deterministic structures in the data. While SOD still misses small scale features at 80% noise level, topological features are still similar to the original phase portrait in Fig. 3(a).

B. Duffing Equation Based Time Series

Using the noise-free Duffing time series and the average mutual information, a delay time of seven sampling time periods is determined. In addition, false nearest neighbors algorithm indicates that the attractor is embedded in $D = 4$ dimensions for 0% noise (see Fig. 7). Six is used for global embedding dimension ($d = 6$), and $k = 3$ is used as local projection dimension. The reconstructed phase portraits for the clean Duffing signal and the added
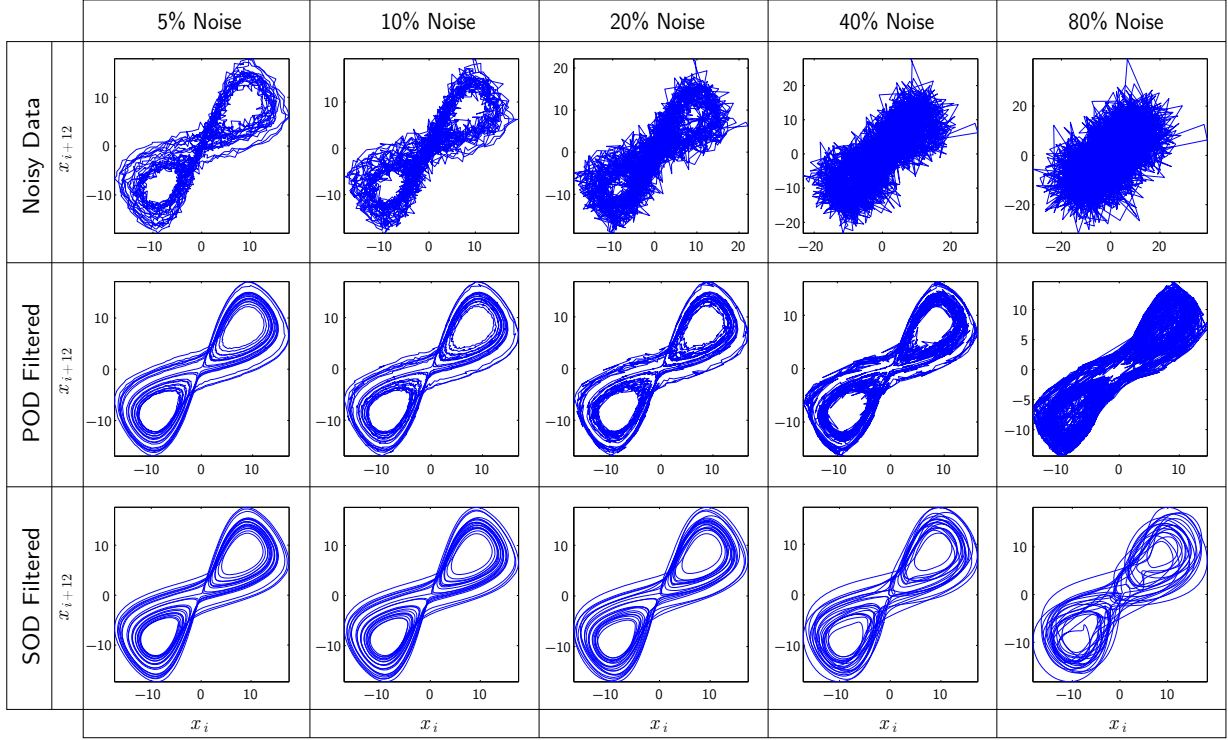
FIG. 6. Phase space reconstructions for noisy and noise-reduced Lorenz time series after ten iterations of the POD- and SOD-based algorithms

random noise are shown in Fig. 8 (a) and (b), respectively. The corresponding power spectral densities for the clean and noise-added signals are shown in Fig. 8 (c). The phase portraits of POD- and SOD-filtered signals with 5% noise are shown in Fig. 8 (d) and (e), respectively. Fig. 8 (f) shows the corresponding power spectral densities. In all these and the following figures, 64 nearest neighbor points were used for POD-based algorithm, and bundles of nine 11-point-long trajectory strands were for SOD-based algorithm. The filtered data shown is after 10 successive applications of the noise reduction algorithms. As before, the decrease in the noise floor after filtering is considerably more dramatic for the SOD algorithm when compared to the POD.

The successive improvements in SNRs after each iteration of the algorithms are shown in Fig. 9, and the corresponding numerical values are listed in Table II for 5% and 20% noise added signals. Here, SNRs for both methods are peaking at some point and then gradually decrease. In addition, the rate of increase is considerably larger for the SOD compared to
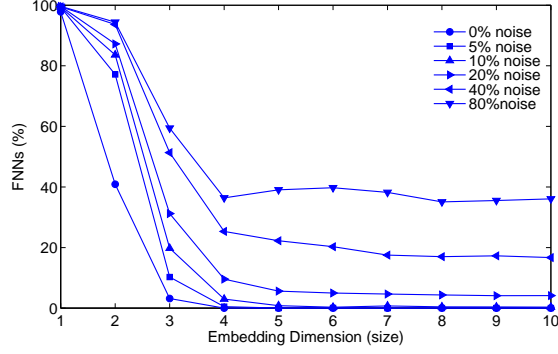
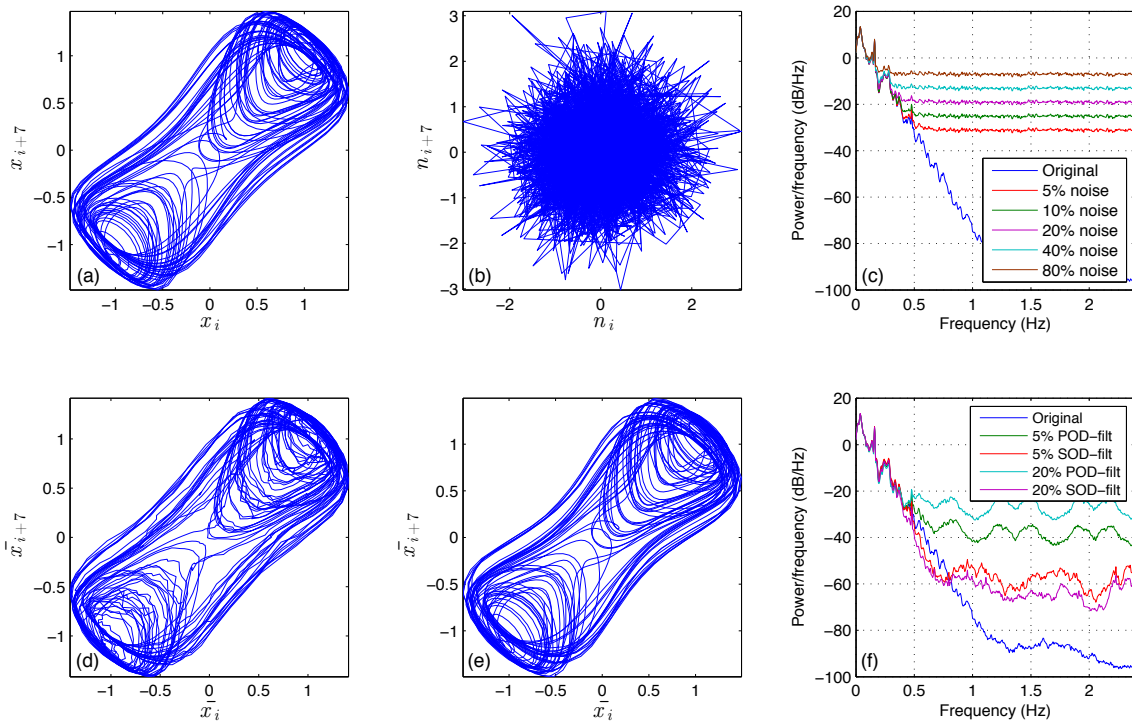FIG. 7. False nearest neighbor algorithm results for Duffing time series

FIG. 8. Reconstructed phase portrait from Duffing signal (a); random noise phase portrait (b); power spectrum of clean and noisy signals (c); POD-filtered phase portrait of 5% noise added data (d); SOD-filtered phase portrait of 5% noise added data (e); and the corresponding power spectrums (f).

the POD algorithm, especially during the initial iterations. SOD is usually able to increase SNR by about $10\,\mathrm{dB}$, while POD provides only about $4\,\mathrm{dB}$ increase on average.

The estimates of the correlation sum and short-time trajectory divergence for the noisy, noise reduced data, and the original clean data from Duffing oscillator are shown in Fig. 10.
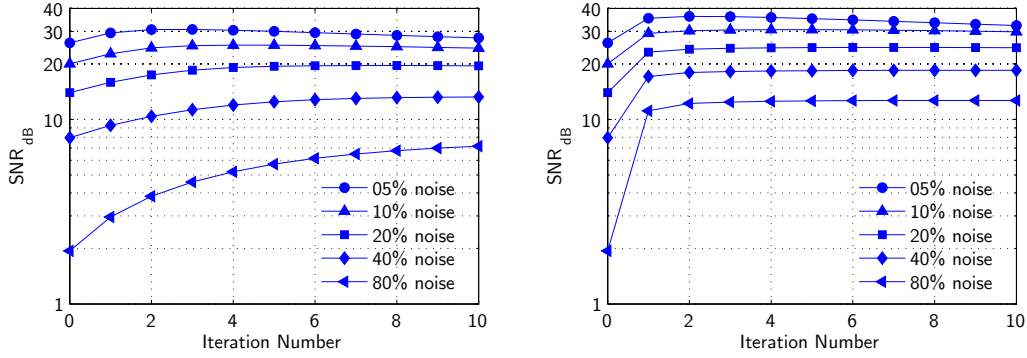
15

FIG. 9. Noisy Duffing time series SNRs versus number of application of POD-based (left) and SOD-based (right) nonlinear noise reduction procedures

For the correlation sum, both POD- and SOD-based algorithms show similar scaling regions and improve substantially on the noisy data, with SOD following the noise trend closer than POD. For the short-time divergence rates both methods manage to recover some of the scaling regions, but SOD does a better job at small scales, where POD algorithm causes large local fluctuations in the estimates.

The qualitative comparison of the Duffing phase portraits for both noise reduction methods are shown in Fig. 11. For consistency all noise-reduced phase portraits are obtained after 10 consecutive applications of the algorithms. While POD does a decent job at low noise levels, it fails at high noise levels where no deterministic structure is recovered at 40% or 80% noise levels. In contrast, SOD provides smoother and cleaner phase portraits at every noise level. Even at 80% noise level, the SOD algorithm is able to recover some large deterministic structures in the data, providing topologically similar phase portrait.

TABLE II. SNRs for the noisy time series from Duffing model, and for noisy data filtered by POD- and SOD-based algorithms

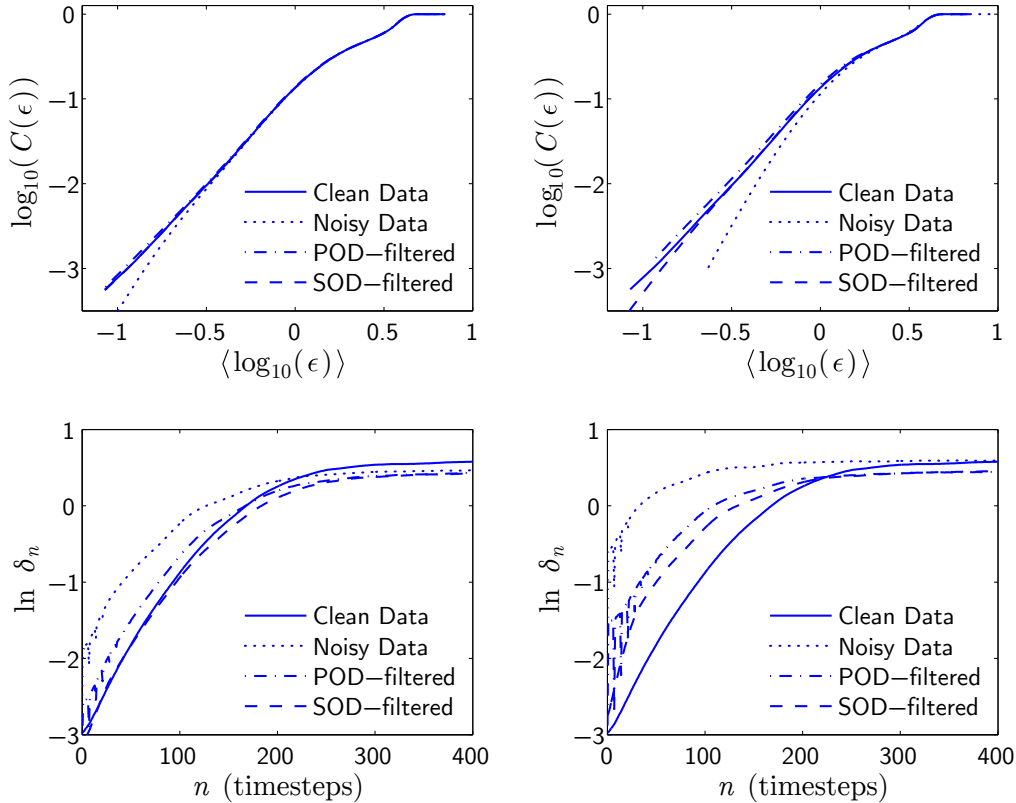| Signal	Duffing + 5% Noise			Duffing + 20% Noise		
Filtering	None	POD	SOD	None	POD	SOD
SNR (dB)	26.0206	30.7583	36.1687	13.9794	19.5963	24.5692

16

FIG. 10. Correlation sum (top plots) and short-time divergence (bottom plots) for 5% (left plots) and 20% (right plots) noise contaminated Duffing time series

VI. DISCUSSION

The new smooth nonlinear noise reduction (SOD-based) method was contrasted with the local projective nonlinear noise reduction (POD-based) for continuously sampled time series. Two sample chaotic time series were used in the analyses: one from double-well Duffing oscillator and another from the Lorenz system. Both time series were contaminated by an additive random noise signal to obtain five different SNR noisy time series for each model. The performance of the noise reduction methods were tested by comparing the resulting (1) power spectral densities, (2) SNRs, (3) correlation sum, (4) short-time trajectory divergence, and (5) by visually inspecting reconstructed phase portraits.

The decrease in the noise floor observed in the power spectral density for the SOD-based method was at least $20\,\mathrm{dB}$ larger then for the POD-based method, which itself improved on the original signals noise floor at best by $10\,\mathrm{dB}$. SNRs show similar trends, where SOD-based
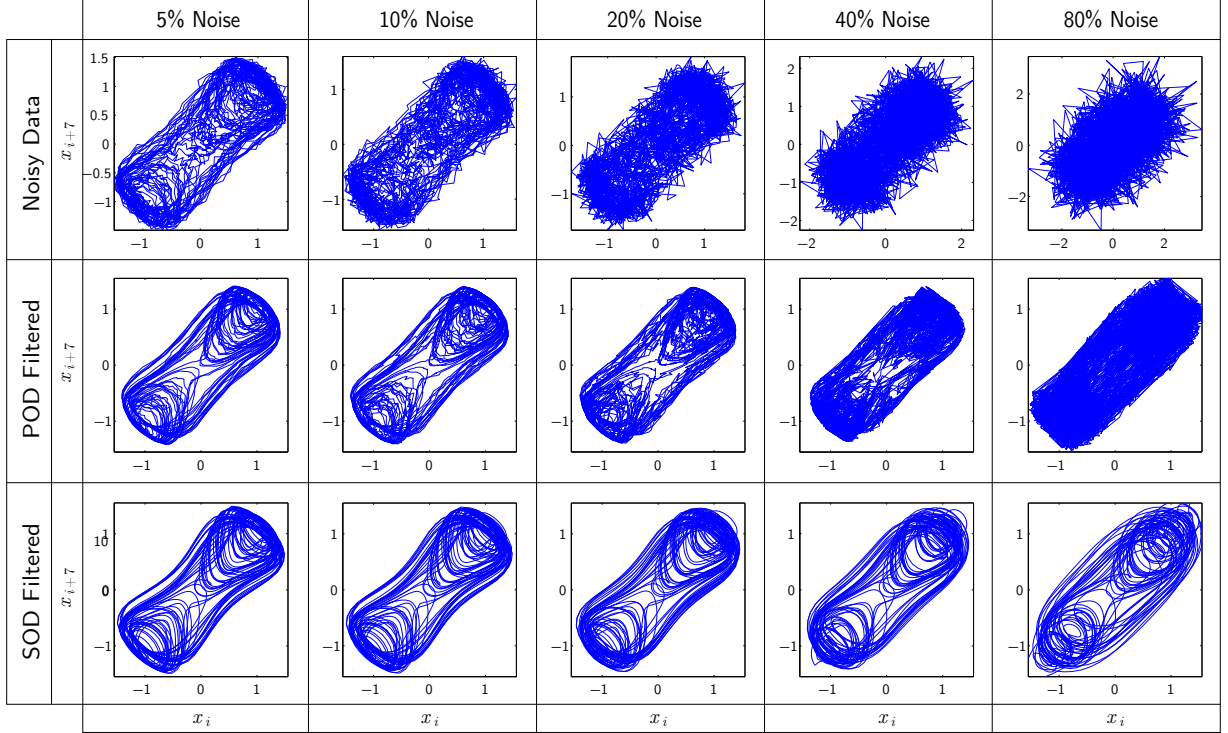
FIG. 11. Phase space reconstructions for noisy and noise-reduced Duffing time series after ten iterations of the POD- and SOD-based algorithms

method manages to improve it by about $10\,\mathrm{dB}$, while POD-based manages only $4 \sim 5\,\mathrm{dB}$ increase. In addition, SOD accomplishes this with considerably fewer applications of the noise reduction scheme, while POD works more gradually and requires many more iterations at higher noise levels.

Estimating nonlinear metrics used in calculations of correlation dimension and short-time Lypunov exponent showed that both methods are adequate at low noise levels. However, SOD performed better at small scales, where POD showed large local fluctuations. At larger noise levels both methods perform similarly for the correlation sum, but SOD provides larger scaling regions for short-time trajectory divergence rates. POD still suffers from large variations at small length/time scales for larger noise levels.

The above quantitative metrics do not tell a whole story, however; we also need to look at trajectories themselves. Visual inspection of the reconstructed phase portraits showed superior results for SOD. POD trajectories had small local variations even at low noise levels, and completely failed to recover any large dynamical structures at large noise levels.

In contrast, SOD method provides very good phase portraits up to 20% noise levels, and for larger noise levels the reconstructions still have clear topological consistency with the original phase portrait. However, SOD also fails to capture finer phase space structures for 80% noise level.

In summary, the SOD-based noise reduction procedure provides considerable improvement over the POD-based method for continuously sampled noisy time series. This was demonstrated using temporal, spatial and spectral characteristics of noise reduced signals. The nature of SOD itself is well suited for identifying smooth local subspaces for noise reduction. However, precisely this feature makes SOD-based method not suitable for direct application to map-like noisy time series. Finally, the SOD-based algorithm is able to handle noise at both large and small scales, while POD is only good for large scale structures in the data.

ACKNOWLEDGEMENT

REFERENCES

[1] E. Kostelich, T. Schreiber, Noise reduction in chaotic time-series data: a survey of common methods, Physical Review E 48 (3) (1993) 1752.

[2] H. Kantz, T. Schreiber, Nonlinear time series analysis, Vol. 7, Cambridge University Press, 2003.

[3] H. Kantz, T. Schreiber, I. Hoffmann, T. Buzug, G. Pfister, L. Flepp, J. Simonet, R. Badii, E. Brun, Nonlinear noise reduction: A case study on experimental data, Physical Review E 48 (2) (1993) 1529.

[4] J. Brocker, U. Parlitz, M. Ogorzalek, Nonlinear noise reduction, Proceedings of the IEEE 90 (5) (2002) 898–918.

[5] P. Tikkanen, Nonlinear wavelet and wavelet packet denoising of electrocardiogram signal, Biological cybernetics 80 (4) (1999) 259–267.

[6] J. Gao, H. Sultan, J. Hu, W. Tung, Denoising nonlinear time series by adaptive filtering and wavelet shrinkage: a comparison, Signal Processing Letters, IEEE 17 (3) (2010) 237–240.

[7] D. Chelidze, W. Zhou, Smooth orthogonal decomposition-based vibration mode identification, Journal of Sound and Vibration 292 (3) (2006) 461–473.

[8] U. Farooq, B. Feeny, Smooth orthogonal decomposition for modal analysis of randomly excited systems, Journal of Sound and Vibration 316 (1) (2008) 137–146.

[9] T. Sauer, J. Yorke, M. Casdagli, Embedology, Journal of Statistical Physics 65 (3) (1991) 579–616.

[10] W. Liebert, K. Pawelzik, H. Schuster, Optimal embeddings of chaotic attractors from topological considerations, EPL (Europhysics Letters) 14 (6) (2007) 521.

[11] C. Rhodes, M. Morari, False-nearest-neighbors algorithm and noise-corrupted time series, Physical Review E 55 (5) (1997) 6162–6170. `doi:10.1103/PhysRevE.55.6162`.

[12] H. Abarbanel, Local false nearest neighbors and dynamical dimensions from observed chaotic data, Physical Review E 47 (5) (1993) 3057–3068. `doi:10.1103/PhysRevE.47.3057`.

[13] I. Ya. Molkov, D. N. Mukhin, E. M. Loskutov, A. M. Feigin, Using the minimum description length principle for global reconstruction of dynamic systems from noisy time series, Physical Review E 80 (4) (2009) 046207.

[14] J. D. Farmer, E. Ott, J. A. Yorke, The dimension of chaotic attractors, Physica D 7, (1983) 153–180.

[15] P. Grassberger, I. Procaccia, Measuring the strangeness of strange attractors, Physica D: Nonlinear Phenomena 9 (1) (1983) 189–208.

[16] R. L. Smith, Estimating Dimension in Noisy Chaotic Time Series, Journal of the Royal Statistical Society B 54 (2) (1992), 329–351.

[17] L. M. Pecora, L. Moniz, J. Nichols, T. L. Carroll, A unified approach to attractor reconstruction, Chaos 17 (2007) 013110.

[18] A. M. Fraser, H. L. Swinney, Independent coordinates for strange attractors from mutual information, Physical review A 33 (2) (1986) 1134.

[19]N. Jevtic, J. Schweitzer, P. Stine, Optimizing nonlinear projective noise reduction for the detection of planets in mean-motion resonances in transit light curves, CHAOS2010 (2011) 191.

[20]A. Chatterjee, An introduction to the proper orthogonal decomposition, Current Science: Computational Science 78 (7) (2000) 808–817.

[21]M. Rathinam, L. R. Petzold, A new look at proper orthogonal decomposition, SIAM J. Numer. Anal. 41 (5) (2003) 1893–1925.

[22]G. Kerschen, J.-C. Golinval, A. F. Vakakis, L. A. Bergman, The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview, Nonlinear Dynamics 41 (2005) 147–169.

[23]D. Chelidze, M. Liu, Reconstructing slow-time dynamics from fast-time measurements, Philosophical Transaction of the Royal Society A 366 (2008) 729–3087.

[24]A. Chatterjee, J. P. Cusumano, D. Chelidze, Optimal tracking of parameter drift in a chaotic system: Experiment and theory, Journal of Sound and Vibration 250 (5) (2002) 877–901.

[25]L. Wiskott, P. Berkes, Is slowness a learning principle of the visual cortex?, in: Proc. Jahrestagung der Deutschen Zoologischen Gesellschaft 2003, Berlin, June 9-13, 2003, special issue of Zoology, 106(4):373-382.

[26]L. Wiskott, Unsupervised learning of invariances in a simple model of the visual system, in: Proc. 9th Annual Computational Neuroscience Meeting, CNS 2000, Brugge, Belgium, July 16–20, 2000, p. 157.

[27]L. Wiskott, Learning invariance manifolds, in: Proc. Computational Neuroscience Meeting, CNS'98, Santa Barbara, 1999, special issue of Neurocomputing, 26/27:925–932.

[28]I. Kovacic, M. J. Brennan, The duffing equation: Nonlinear oscillators and their behaviour, Wiley, 2011.

[29]I. Stewart, Mathematics: The lorenz attractor exists, Nature 406 (6799) (2000) 948–949.

[30]T. Ivancevic, L. Jain, J. Pattison, A. Hariz, Nonlinear dynamics and chaos methods in neurodynamics and complex data analysis, Nonlinear Dynamics 56 (1-2) (2009) 23–44.

[31]J. B. Dingwell, Lyapunov exponents, Wiley Encyclopedia of Biomedical Engineering.

[32] A. Wolf, J. Swift, H. Swinney, J. Vastano, Determining lyapunov exponents from a time series, Physica D: Nonlinear Phenomena 16 (3) (1985) 285–317.

[33] A. Wolf, Quantifying chaos with lyapunov exponents, Chaos (1986) 273–290.

[34] P. Grassberger, I. Procaccia, Characterization of strange attractors, Physical review letters 50 (5) (1983) 346–349.

[35] P. Grassberger, Generalized dimensions of strange attractors, Physics Letters A 97 (6) (1983) 227–230.

[36] Concept similar to SOD called *slow feature analysis* has been used for pattern analysis in neural science[25–27].