Optimizing of recurrence plots for noise reduction

Lorenzo Matassini* and Holger Kantz

Max-Planck-Institut für Physik komplexer Systeme, Nöthnitzer Strasse 38, D-01187 Dresden, Germany

Janusz Hołyst

Faculty of Physics, Warsaw University of Technology, Koszykowa 75, PL-00662 Warsaw, Poland

Rainer Hegger

Institute for Physical and Theoretical Chemistry, J. W. Goethe-University, Marie Curie Strasse 11, D-60439 Frankfurt, Germany

We propose a way to automatically detect the best neighborhood size for a local projective noise reduction filter, where a typical problem is the proper identification of the noise level. Here we make use of concepts from the recurrence quantification analysis in order to adaptively tune the filter along the incoming time series. We define an index, to be computed via recurrence plots, whose minimum gives a clear indication of the best size of the neighborhood in the embedding space. Comparison of the local projective noise reduction filter using this optimization scheme with the state of the art is also provided.

I. INTRODUCTION

Noise reduction means that one tries to decompose a time series into two components, one of which supposedly contains the signal and the other one contains random fluctuations. We want to focus our attention here on the local projective noise reduction scheme presented in Ref. [1], optimizing the most important parameter, namely, the neighborhood size. This step was lacking and, therefore, it was usually performed through a visual inspection of the sample. Since we will mainly discuss the noise reduction for speech signals, we start with a short introduction of the sound generating mechanism of humans. In particular, as reported in Ref. [2], we make use of the result that the reconstruction of attractors and the estimation of their properties indicate low dimensionality of the system generating the signal (this result has been achieved through an analysis of a two-mass model of vocal-fold vibrations with methods from nonlinear dynamics). On the other hand, human speech is highly nonstationary, since inside the logical units, called phonemes, a typical speech signal is almost periodic, but the concatenation of different phonemes does not represent a low-dimensional deterministic dynamical system. In Ref. [1] a local projective noise reduction scheme is presented, based on the identification and exploration of quasideterminitic structure in the voice signal, in such a way that tools developed in the framework of nonlinear time series analysis [3] (which relies on the hypothesis of deterministic chaos, namely, the signal has to reflect the complex dynamics of a purely deterministic and often with few degrees of freedom system) can be profitably used.

The reason for the success of the method relies on the fact that human speech can be considered as a dynamical phenomenon driven by few parameters, with an instantaneous dynamics with limit cycle solutions. As reported in Ref. [4],

if a D-dimensional deterministic dynamical system depends on P parameters with slow time dependence, then delay vectors of sufficient embedding dimension [5,6] are approximately confined to a $(D+P)$-dimensional manifold. The confinement is violated on length scales of the order of the standard deviation of the data times the average parameter change per unit step. The use of embedding dimensions larger than optimal solves the problem of nonstationarity in many applications, included the noise reduction for human speech. Data belonging to different parameter settings (i.e., different phonemes) populate different regions of the embedding space and are thus distinguishable. When the parameter variation is nondeterministic, these changes cannot be predicted from the data, but can be implicitly adjusted through the selection of the neighbors. The hypothesis of slow time dependence of the P parameters is fulfilled here, since a phoneme comprises several repetitions (usually 10–20) of a subunit called *pitch* and within a phoneme there is essentially no parameter variation. The nonstationarity is involved in the concatenation of phonemes to build up words and sentences.

To get an impression of how the local projective noise reduction scheme works, assume that one has to eliminate noise from music stored on an old-fashioned long playing (LP) record, induced by scratches on the black disc. The task becomes almost trivial if one can make use of several samples of this LP. When playing them synchronously, the signal part of the different tracks is identical, whereas the noise part is independent: As a consequence of that, already a simple averaging would enhance the sound quality. In deterministic chaotic signals, this redundancy is stored in the past. Similar initial conditions will behave in a similar way, at least for short periods. In human voice signals there is no need to suppose a chaotic behavior, since every phoneme is made up of pitches in an almost periodic fashion. This means that every logical unit provides all the redundancy required for its filtering.

Careful investigation of time and length scales shows that the sound wave characterizing a single phoneme has a char-

---

*Email address: lorenzo@mpipks-dresden.mpg.de

acteristic profile on about 5–10 ms. Since a phoneme has a duration of about 100 ms, the redundancy consists of 10–20 repetitions of pitches. Searching for neighbors in equal phonemes belonging to other words introduces large numerical efforts, requires longer sentences and does not improve the situation much, since changed amplitudes and dilatation or compression in time destroy the similarity. Thus the local projective noise reduction scheme works with just intraphoneme neighbors, allowing an almost online execution. For a detailed description of the algorithm, see Ref. [1]. Here we want to address an optimization problem, namely, we want to provide a mechanism able to automatically identify the best neighborhood size, strictly related to the noise level.

II. LOCAL PROJECTIVE NOISE REDUCTION SCHEME

Deterministic dynamical systems are defined by equations of motion in a vector valued space. Since in most experiments only a single observable is measured, one needs a way to reconstruct vector valued time series from scalar time series. Embedding techniques are employed for this purpose. The reconstructed sequence of vectors can be interpreted as a sample of a trajectory of a dynamical system in a reconstructed (artificial) phase space, which is related to the unknown space of the underlying dynamical system by some smooth coordinate transform, if the measurement function was smooth. As a consequence, its invariants such as attractor dimensions, Lyapunov exponents and entropies are the same.

Let us consider a dynamical system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ in a phase space $\Gamma \subset \mathbb{R}^d$, a measurement function $h: \mathbb{R}^d \to \mathbb{R}$, and a sampling interval Δt. Denote the scalar measurements obtained through the sampling by $s_n := h(\mathbf{x}(t = n\Delta t))$. Delay vectors are constructed as follows:

$$\mathbf{s}_n = [s_n, s_{n-\tau}, s_{n-2\tau}, \ldots, s_{n-(m-1)\tau}], \tag{1}$$

where m is the dimension of the vector and $\tau \in \mathbb{N}$ a delay that makes no mathematical difference, but quite important in practice. If $m > 2D_f$, the m-dimensional delay embedding space is equivalent to the original unobserved phase space of the dynamical system, since, in particular, the dynamics of s is deterministic. The requirement of $m > 2D_f$ is related to geometry and guarantees that self-intersections of the reconstructed sets due to nonlinearities are nongeneric. For more details, see [1,3,4].

When looking for neighbors, we have to restrict our search to a subset of the embedding space. The size of it plays a crucial role. A very small value will provide nothing but the point itself and, therefore, the filter will produce no effect on the time series. A very big value will identify all the points as possible neighbors and the algorithm will perform just a global averaging, destroying completely the original voice. From a computational point of view, the smaller the size of the neighborhood is, the faster the program runs. But there is a lower limit for the size of the subspace, given by the noise level. As depicted in Fig. 1, the diameter of the neighborhood has to be bigger than the size of the cloud of points contaminated by noise. We want to filter the point at
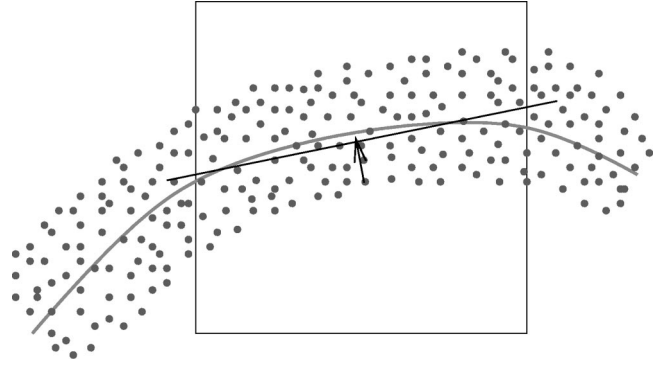

FIG. 1. How the local projective noise reduction scheme works. The curve represents a branch of the attractor in a proper embedding space (qualitative illustration), which shares with the phase space all the topological properties of the system. After the contamination with noise every point is moved away from the original curve in a random way. Imagine now that we want to remove noise from the sample at the starting point of the arrow. The idea is to collect neighbors of this point up to a certain distance (the parameter we want to optimize here), getting a linear local approximation of the original attractor. A projection of the selected point onto the straight line performs the noise reduction.

the base of the arrow. For this purpose we look for its neighbors in a subset of the embedding space indicated by the big square. The bold curve represents the original attractor and the cloud of points is the effect of the contamination with noise. The straight line is the best linear approximation of the attractor given the collected set of neighbors. The arrow indicates what the filter does, namely, the projection of the actual point onto the local linear reconstruction of the attractor.

The local projective noise reduction algorithm can, therefore, be summarized in the following scheme.

(1) For every delay vector $\mathbf{s}_n$, all neighbors (with respect to a given neighborhood size to be optimized) are collected.

(2) The covariance matrix $C_{ij} = \Sigma_{\mathcal{U}_n}(\hat{\beta}_k)_i (\hat{\beta}_k)_j$, where $\mathcal{U}_n$ represents the neighborhood, is computed.

(3) The vectors corresponding to the largest singular values are supposed to represent the directions spanning the hyperplane that approximates the dynamics.

(4) The projection onto these dominant directions performs the noise reduction.

The noise reduction algorithm can also be seen as a minimization problem with the following cost function:

$$L = \sum_{n' \in \mathcal{U}_n} \left[\sum_{q=1}^{Q} \mathbf{a}^q \cdot (\mathbf{a}^q \cdot \mathbf{s}_{n'}) \right]^2 - \sum_{q,q'} \lambda_{q,q'}(\mathbf{a}^q \cdot \mathbf{a}^q - \delta_{q,q'}). \tag{2}$$

It has to be minimized with respect to $\mathbf{a}^q$ (orthonormal vectors such that the local projection onto these vectors is minimal) and $\lambda_{q,q'}$ (Lagrange multipliers introduced in order to impose the orthogonality of the $\mathbf{a}^q\mathbf{s}$.).
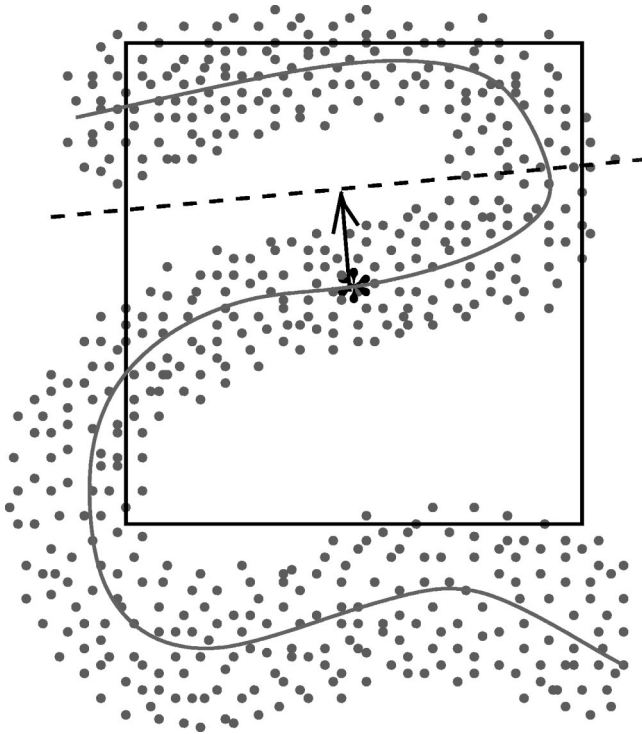
FIG. 2. Effect of a too big neighborhood, qualitative illustration. The algorithm is not able to correctly identify the original manifold because two different branches of the attractor are erroneously considered as neighbors. Therefore, the projection is performed onto a wrong direction and the quality of the filtering is bad.

III. IDENTIFICATION OF THE BEST NEIGHBORHOOD SIZE

The effect of a too big neighborhood is reported in Fig. 2, where the identification of the original manifold cannot be correctly performed and, therefore, the projection of the actual point does not act along the proper direction. This happens because of the minimization procedure. We have to find a local linear approximation of the attractor that minimizes the sum of distances from the noisy points. With such a neighborhood we consider too many points, including false ones (because belonging to another branch of the attractor) and the resulting manifold is far away from the correct one. Also in the case depicted in Fig. 3, a clear identification of the original manifold is not possible. Here the solution of the problem is not unique, due to the fact that the size of the neighborhood is too small and the points are distributed almost uniformly and the projection almost random.

It is thus evident that one needs a mechanism to decide which is the best size of the neighborhood to be taken into consideration. As a further example let us have a look at the lower panel of Fig. 3: Here we consider a whistle, one of the simplest acoustic signals that an human being can generate. The signal is almost sinusoidal and, therefore, with the proper parameters, the attractor looks like a circle. The lower left panel of Fig. 3 refers to the reconstructed embedding space related to this signal. We proceed now by adding a 30% noise to the whistle and filtering the new signal with a wrong set of parameters, namely, with a too small value of
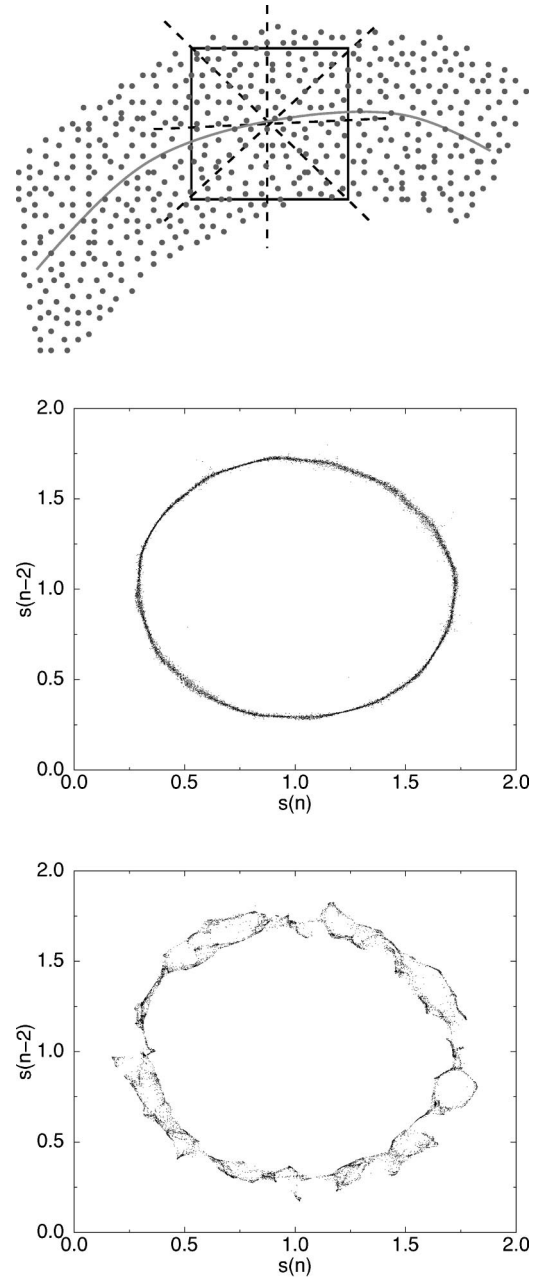

FIG. 3. Effect of a too small neighborhood. Upper panel: The original manifold cannot be correctly identified, since the distribution of points inside the neighborhood is almost uniform. Middle panel: Attractor of a clean whistle in a proper embedding space. Bottom panel: Attractor of the same whistle after the contamination with a 30% additive noise and the filtering with a too small neighborhood size. The result is the born of completely artificial structures due to projections onto random directions.

the neighborhood size. The lower right panel of Fig. 3 shows how the reconstructed attractor looks. We do not report the picture of the attractor after the correct filtering, since within the resolution of this paper it would be almost indistinguishable from the original. The reason for the strange shape of the right panel is the following. Once considered the embedding space for the noisy signal, we look for neighbors in such small regions that the distributions of points
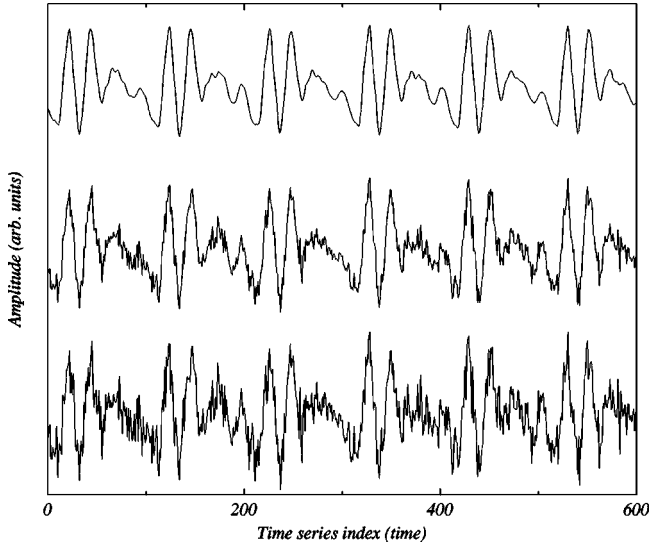
FIG. 4. Example of a single phoneme. Upper panel: Original time series, without noise. Middle panel: With 30% noise. Lower panel: With 50% noise. Time along the x axis, amplitude of the microphone signal in arbitrary units along the y axis.

within them is almost uniform; therefore, we are not able to identify the original manifold and locally we perform projections onto random, hence wrong, directions.

We proceed now adding noise to a phoneme, as illustrated in Fig. 4 (time along the x axis, amplitude of the microphone signal in arbitrary units along the y axis). Starting from the upper panel we have the original time series plus 0%, 30%, and 50% of noise. The voice was recorded with a 20-kHz sampling rate, so that 2000 points correspond to 100 ms. The recurrence plot [7] of the noise-free time series is reported in Fig. 5. Recurrence plots are a qualitative tool with several potential applications. They consist of plotting the following square matrix:
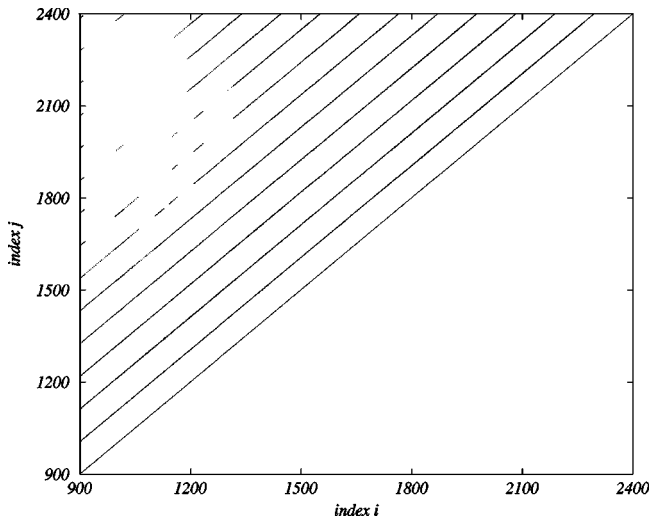

FIG. 5. Recurrence plot of the time series reported on the upper panel of Fig. 4. A dot in position (i,j) reflects the condition $|s_i - s_j| < \epsilon$. The observation of lines parallel to the main diagonal is a significant signature of determinism.
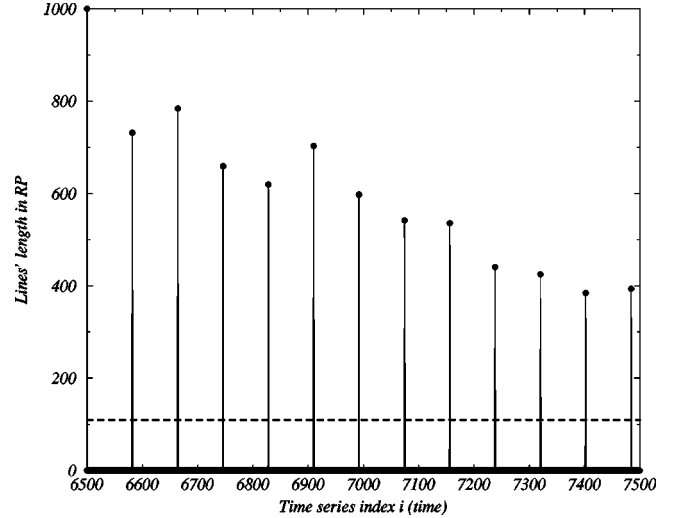

FIG. 6. Histogram derived from the recurrence plot (RP) of Fig. 5 according to Eq. (4). In this case we get $N_p(\epsilon) = 13$.

$$r_{ij} = \Theta(\epsilon - |\mathbf{s}_i - \mathbf{s}_j|), \qquad (3)$$

where ϵ is a predefined tolerance level and $\Theta(\ )$ is the Heaviside step function, $\Theta(x) = 1$ if $x > 0$ and $\Theta(x) = 0$ elsewhere. Hence, the matrix elements are unity for all pairs of indices i, j whose corresponding delay vectors have a distance smaller than ϵ. If a recurrence plot as in Fig. 5 occurs, this is a clear indication that delay vectors really represent meaningful states. The line structure shows the approximate periodicity inside phonemes and the number of intraphoneme neighbors.

A point in the recurrence plot mirrors a recurrence of the dynamical process and the plot can be considered as a global picture of the autocorrelation structure of the system. Consequently, a recurrence plot visualizes the distance matrix, which in turn represents the autocorrelation present in the series at all possible time scales. A recurrence can, in principle, be observed by chance whenever the system explores two nearby points of its state space. On the contrary, the observation of recurrence points consecutive in time (and then forming lines parallel to the main diagonal) is an important signature of deterministic structuring [8–10]. The noise does not destroy the qualitative structure of Fig. 5, provided one is able to identify the correct set of parameters. The distinction between signal and noise based on the deterministic content is still possible and, therefore, also the noise reduction is not inhibited.

Starting from the recurrence plot (defined here only in the 1000 central points to avoid edge effects), we define the following quantities.

(1) $N_p(\epsilon)$: We compute the histogram along the main diagonal direction,

$$h_i = \sum_{k-j=i} r_{jk}, \qquad (4)$$

where r_{jk} is a point in the recurrence plot and h_i the histogram we get after this computation, reported in Fig. 6. Since we want to count the number of peaks and to be sure they are sharp, we define a threshold (dashed line of Fig. 6) as the
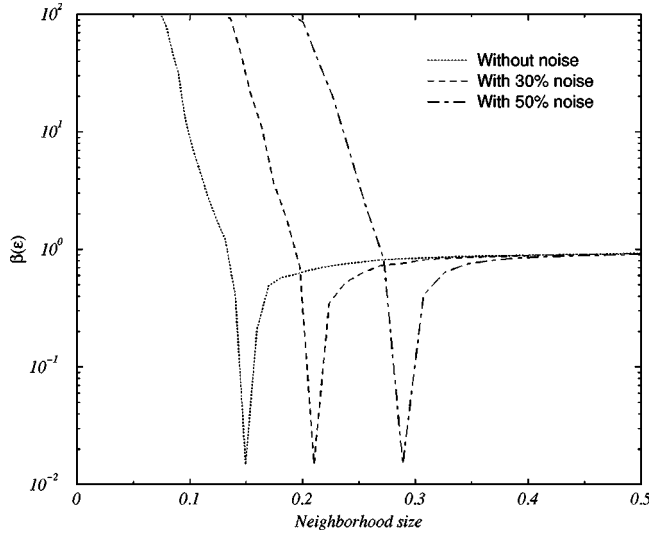
FIG. 7. Identification of the best ϵ for the three cases of Fig. 4. It is very intuitive that in increasing the noise level one has to consider a bigger neighborhood size.

FIG. 8. Performance of the optimized local projective noise reduction scheme. Upper panel: Original ''clean'' speech signal. Middle panel: Numerical noise added to the time series. Lower panel: Residual noise (difference between the filtered noisy signal and the original one). The amplitude of the signal along the y axis is given in arbitrary units. It is interesting to note peaks in the residual noise corresponding to transitions between phonemes, a typical limitation of this approach.

average height of the histogram plus three times the standard deviation. The number of peaks $N_p(\epsilon)$ is then given by the number of h_i such that $h_i >$ threshold and $h_{i-1} <$ threshold (if a peak is made of several $h_i s$, not the case of Fig. 6, we want to count it just once).

(2) $N_\perp(\epsilon)$: We compute $\Sigma_{i,j} r_{ij}/N$, the average number of neighbors that points have.

Of course these two quantities depend on ϵ: The best value of it is the one that maximizes the number of peaks and produces a value of $N_\perp(\epsilon)$ as close as possible to $N_p(\epsilon)$. The optimal recurrence plot is the one where lines are long but not fat, in order to avoid phase identification problems. So the task is finding the value of ϵ such that the following quantity is minimized:

$$\beta(\epsilon) = \frac{|N_\perp(\epsilon) - N_p(\epsilon)|}{N_\perp(\epsilon)} \qquad (5)$$

The purpose of $N_\perp(\epsilon)$ as denominator of Eq. (5) is to get a better identification of the minimum of $\beta(\epsilon)$: In Fig. 7 we can see the result of such a computation for the three different noise levels depicted in Fig. 4, namely, 0%, 30%, and 50%. Not surprisingly, a bigger noise requires a bigger neighborhood. For small values of ϵ we have almost no point outside the main diagonal; therefore, $N_\perp(\epsilon)$ is close to zero but this is not the case for $N_p(\epsilon)$, since the few points are considered as isolated peaks: Hence $\beta(\epsilon)$ is very big. Big values of ϵ are such that the recurrence plot is almost full of points: $N_\perp(\epsilon)$ is close to N (the length of the time series under observation) and there is no isolated peak. Consequently $\beta(\epsilon) \approx 1$. The optimal situation is when $N_\perp(\epsilon) \approx N_p(\epsilon)$, namely, the lines are neither fragmented nor fat: All the recurrence points belong to line structures and the quantity is maximized.
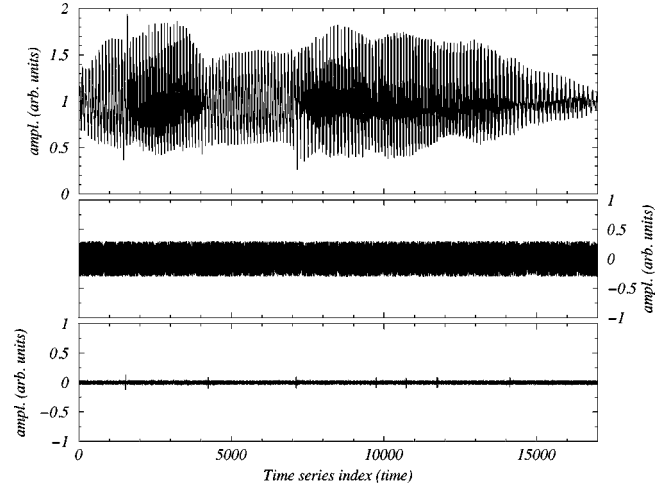
IV. RESULTS AND COMPARISONS

Now we want to show some results of the application of the method and a comparison with the state of the art. A typical result is provided by Fig. 8, where the three panels show the original signal (upper panel), the additive noise used to contaminate it (middle panel) and the difference between the filtered noisy signal and the clean one (lower panel, what we call residual noise). The variance of the added noise is 60% the variance of the signal, giving rise to a signal-to-noise ratio (SNR)$=2.2$ dB. The variance of the residual noise is 4.6% the variance of the original signal, i.e., SNR$=13.4$ dB. In the case depicted in Fig. 8 the algorithm is, therefore, able to provide a gain of 11.2 dB.

The residual noise shows peaks corresponding to transitions between phonemes, as clearly visible from the lower panel of Fig. 8. This is a limitation of the local projective noise reduction scheme. This filter is working with intraphoneme neighbors and the quantity close to the transitions is smaller than in the middle of a phoneme because of nonstationary effects. The quality of the filter is, therefore, poorer there, as reflected by the peaks in the residual noise.

The state of the art noise reduction scheme is based on the Ephraim and Malah filter, making use of spectral subtraction techniques. Spectral subtraction is a popular method of speech enhancement, if the speech signal is corrupted by additive noise. It is based on the manipulation of the magnitude of the noisy-speech spectrum. The application proposed by [11,12] makes use of two filter banks with bark-scaled frequency bands. A discrete wavelet transformation and a nonuniform polyphase filterbank. The basic idea is the following: Signal enhancement is performed in the frequency domain by operating on the Fourier transformation of the observed samples. The estimation of the noise-reduced
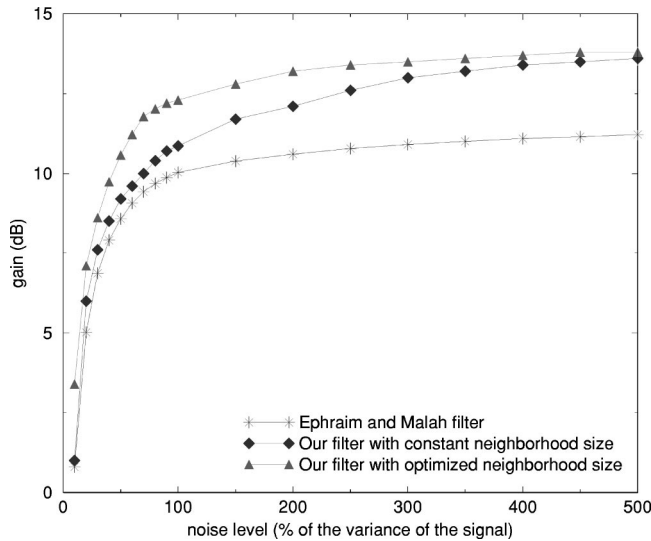
FIG. 9. Comparison of the optimized local projective noise-reduction scheme with the Ephraim and Malah filter, state of the art.

speech spectrum is obtained by subtracting an estimated mean spectral magnitude of the noise from the spectral magnitude of the signal containing speech plus noise. It is a kind of spectral weighting, followed by an inverse Fourier transformation to come back to the time domain. For more details see [12].

In Fig. 9 we show the performance of the filter compared to the Ephraim and Malah noise reduction scheme as implemented by Gülzow and collaborators in [12]. Also the improvement of this optimized ϵ (compared to the case where ϵ is kept constant) is visible. As a measure of performance, we use the gain in decibels, given by

$$(\text{gain}) = 10 \log\left(\frac{\Sigma(\hat{y}_k - s_k)^2}{\Sigma(y_k - s_k)^2}\right), \qquad (6)$$

where s_k is the clean signal, y_k the noisy signal, and $\hat{y}_k$ the signal after noise reduction. Fig. 9 is the summary of several cases like Fig. 8, where the variance of the noise is ranging from 10% up to 500% the variance of the original speech signal. The improvement obtained when using the indication of Eq. (5) is particularly significant when one has to filter a full sentence, where the noise may affect different words in a very different fashion. In such a case the use of a constant ϵ is far from the optimal solution and the new tool we have implemented plays a fundamental role in the quality one can achieve.

V. CONCLUSIONS

Noise is almost ubiquitous and the task concerned with its remotion is a very challenging one. We have introduced here the local projective noise reduction scheme, an algorithm exploiting deterministic structures of the signal in order to separate it from the noise, which is supposed not to be correlated with the clean time series. The main problem when using this filter is the tuning of its parameters, in particular, the size of the neighborhood. With this work we have proposed a way to optimize this value with an automatic scheme. The optimal neighborhood size is the one that provides a recurrence plot whose lines are neither fragmented nor fat. The algorithm becomes now a double minimization problem, the first being related to the cost function involved in the projection, the second involving the minimization of the index $\beta(\epsilon)$ with respect to ϵ.

The results presented show the robustness of the method in the correct identification of the noise level present in the time series. With such an approach it is no longer necessary to visually inspect the recorded noisy sample during the filtering in order to get the best result with the minimum efforts, namely, within the shortest possible computational time.

[1] R. Hegger, H. Kantz, and L. Matassini, Phys. Rev. Lett. **84**, 4092 (2000).

[2] I. Steinecke and H. Herzel, J. Acoust. Soc. Am. **97**, 1874 (1995).

[3] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis* (Cambridge University Press, Cambridge, 1997).

[4] R. Hegger, H. Kantz, L. Matassini, and T. Schreiber, Phys. Rev. Lett. **84**, 3197 (2000).

[5] F. Takens, in *Dynamical Systems and Turbulence*, edited by D. A. Rand and L.-S. Young, Lecture Notes in Mathematics Vol. 898 (Springer, Berlin, 1981), pp. 366–381.

[6] T. Sauer, J. A. Yorke, and M. Casdagli, J. Stat. Phys. **65**, 579 (1991).

[7] M. C. Casdagli, Physica D **108**, 12 (1997) ; J. P. Eckmann, S. O. Kamphorst, and D. Ruelle, Europhys. Lett. **4**, 973 (1987).

[8] C. Manetti, M. A. Ceruso, A. Giuliani, C. L. Webber, and J. P. Zbilut, Phys. Rev. E **59**, 992 (1999).

[9] J. P. Zbilut, A. Giuliani, and C. L. Webber, Phys. Lett. A **246**, 122 (1998).

[10] L. L. Trulla, A. Giuliani, J. P. Zbilut, and C. L. Webber, Phys. Lett. A **223**, 43 (1996).

[11] S. F. Boll, IEEE Trans. Acoust., Speech, Signal Process. **27**, 113 (1979).

[12] T. Gülzow, A. Engelsberg, and U. Heute, Signal Process. **64**, 5 (1998).