Nonlinear Speech Analysis Using Models for Chaotic Systems

Iasonas Kokkinos, *Student Member, IEEE,* and Petros Maragos, *Fellow, IEEE*

***Abstract*—In this paper, we use concepts and methods from chaotic systems to model and analyze nonlinear dynamics in speech signals. The modeling is done not on the scalar speech signal, but on its reconstructed multidimensional attractor by embedding the scalar signal into a phase space. We have analyzed and compared a variety of nonlinear models for approximating the dynamics of complex systems using a small record of their observed output. These models include approximations based on global or local polynomials as well as approximations inspired from machine learning such as radial basis function networks, fuzzy-logic systems and support vector machines. Our focus has been on facilitating the application of the methods of chaotic signal analysis even when only a short time series is available, like phonemes in speech utterances. This introduced an increased degree of difficulty that was dealt with by resorting to sophisticated function approximation models that are appropriate for short data sets. Using these models enabled us to compute for short time series of speech sounds useful features like Lyapunov exponents that are used to assist in the characterization of chaotic systems. Several experimental insights are reported on the possible applications of such nonlinear models and features.**

***Index Terms*—Chaos, nonlinear systems, speech analysis.**

I. Introduction

FOR SEVERAL DECADES, the traditional approach to speech modeling has been the linear (source-filter) model where the true nonlinear physics of speech production are approximated via the standard assumptions of linear acoustics and one-dimensional (1-D) plane wave propagation of the sound in the vocal tract. The linear model has been applied to speech coding, synthesis and recognition with limited success; to build successful applications, deviations from the linear model are often modeled as second-order effects or error terms. However, there is strong theoretical and experimental evidence [1]–[7] for the existence of important nonlinear aerodynamic phenomena during the speech production that cannot be accounted for by the linear model. We view the linear model only as a first-order approximation to the true speech acoustics which also contain second-order and nonlinear structure. The investigation of speech nonlinearities can proceed in several directions. In our on-going research [7], [8] we focus on a nonlinear signal processing approach, which consists in developing efficient computational models for detecting nonlinear phenomena in speech and extracting related acoustic signal features. One such important nonlinear phenomenon is turbulence during speech production. One viewpoint from which turbulence can be explored is its nonlinear dynamics aspect, which leads us to the theory of *chaos* [9], [10]. Previous research in the existence and analysis of chaotic behavior that the speech production system can display includes [7], [8], [11]–[21].

Attempting to explore the link between speech turbulence and chaos, in this paper we use concepts and methods from chaotic systems to model and analyze nonlinear dynamics in speech signals. The modeling is done not on the scalar speech signal, but on its reconstructed multidimensional attractor by embedding the scalar signal into a phase space. Such embeddings are often used in chaotic systems analysis [8]–[10], [22]. In this setting, modeling the system dynamics is interpreted as a function approximation task, using an observed set of input-output pairs. One of the main problems of a function approximation task is how to strike a balance between fidelity to the observed input-output pairs and generalization ability; this is particularly prominent in our case where we are provided with few data residing in a high-dimensional space. As part of our contributions, we have analyzed and compared a variety of nonlinear models for the purpose of approximating the dynamics of the speech production system on its reconstructed attractor. The models we have experimented with include approximations based on global or local polynomials, which are commonly used to model chaotic systems, as well as nonlinear function approximations inspired from machine learning such as radial basis function networks, fuzzy-logic systems and support vector machines.

Our focus has been on facilitating the application of the methods of chaotic signal analysis even when a short time series is available. In most of the previous work [12], [14], [15], [17], [19] sustained vowels have been used, and the major part of research in modeling of chaotic systems assumes that a long time-series is available -10^4 samples or more. Instead, we have used relatively short time-series, on average 800 samples, which introduced an increased degree of difficulty that was dealt with by resorting to sophisticated function approximation models that are appropriate when a short data set is available. Using these models has enabled us to compute useful features that are used to characterize chaotic systems and the nonlinear dynamics of speech, like Lyapunov Exponents, even when less than 10^3 points are available. Since we are not familiar with any comparison of the models we have used in the time-series

Manuscript received December 20, 2002; revised September 10, 2004. This work was supported in part by the European STREP ‘‘HIWIRE" and NoE ‘‘Muscle" and the Greek research program "HRAKLEITOS", which is co-funded by the European Social Fund (75%) and National Resources (25%). The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Juergen Schroeter.

The authors are with the School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece (e-mail: jkokkin@cs.ntua.gr; maragos@cs.ntua.gr).

Digital Object Identifier 10.1109/TSA.2005.852982

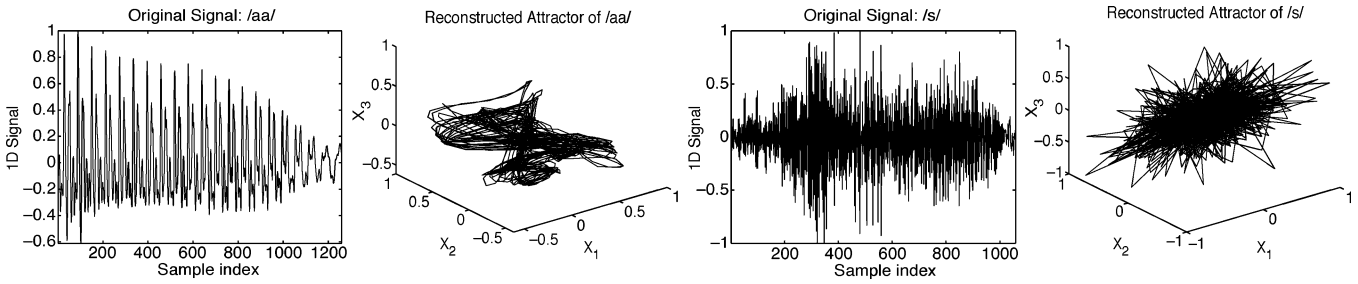

Fig. 1. Waveforms and attractors for /aa/ and /s/. (Only the first three dimensions of the multidimensional attractor are plotted).

analysis literature using short data sets, we also present in this paper our experience gained from working with a wide repertoire of these models.

Our contributions in this paper are twofold: 1) investigating the appropriateness of complex nonlinear function approximation methods for speech modeling and 2) experimentally validating the feasibility and potential merits of carrying out speech analysis using methods for chaotic systems.

Organization of the paper: Section II briefly summarizes the embedding procedure we used, that enables us to reconstruct the geometric structure of the attractor of a chaotic system. Section III presents Lyapunov Exponents and the problems associated with their computation. In Section IV we review the most important nonlinear models that we have used, report our conclusions concerning their suitability and select the most efficient model. In Section V we present our experimental results from applying the previous methods for chaotic systems analysis to speech signals.

II. Speech Signal Embedding and Attractor

We assume that the speech production system can be viewed in discrete time n as a low-dimensional nonlinear dynamical system $Z(n+1) = H(Z(n))$, where the phase space of $Z(n)$ is multidimensional. The observed speech signal segment $s(n)$, $n = 1, \ldots, N$, can be considered as a 1D projection via a vector function V applied to the unknown dynamic variables $Z(n)$, i.e., $s(n) = V(Z(n))$. It is possible that the apparent complexity or randomness in the scalar signal could be due to loss of information during the projection. According to a well-established signal *embedding* theory (see [9] for a comprehensive presentation), under mild assumptions about V, the vector

$$X(n)=[s(n), s(n-T_d), s(n-2T_d), \ldots, s(n-(D_e-1)T_d]$$

formed by samples of the original signal delayed by multiples of a constant time delay T_d defines a motion in a reconstructed D_e-dimensional space that has many common aspects with the original phase space of $Z(n)$. Thus, by studying the constructible dynamical system $X(n) \rightarrow X(n+1)$ we can uncover useful information about the original unknown dynamical system $Z(n) \rightarrow Z(n+1)$ provided that the unfolding of the dynamics is successful. However, the embedding theorem does not specify a method to determine the required parameters (T_d, D_e); hence, procedures to estimate good values of these parameters are essential.

T_d is related to the correlation among speech samples: a small T_d is equivalent to a large correlation between $s(n)$ and $s(n - T_d)$; on the other hand a large T_d can result in $s(n)$ and $s(n-T_d)$ being independent and hence $s(n - T_d)$ yields no useful part of the system's history [9]. In order to achieve a compromise between these two conflicting arguments, the following measure of nonlinear correlation is used:

$$I(T)=\sum_{n=1}^{N-T} P\left(s(n), s(n+T)\right) \cdot \log_2\left[\frac{P\left(s(n), s(n+T)\right)}{P\left(s(n)\right) P\left(s\left(n+T\right)\right)}\right]$$

where $P(\cdot)$ denotes probability. $I(T)$ measures the *average mutual information* between samples that are T positions apart, i.e., the amount of information we get on average about $s(n+T)$ from $s(n)$. As in [8], [9] we choose T_d as the location of the first minimum of $I(T)$.

After fixing T_d, the next step is to select the embedding dimension D_e of the reconstructed vectors. Projection of the system attractor to a lower dimensional space may cause some of the points on the attractor that were initially far apart to come close to each other; such pairs of points are called *false neighbors*. As in [8], [9], we find the embedding dimension D_e by increasing its value until the percentage of false neighbors reaches zero (or is minimized in the presence of noise). A true vs. false neighbor criterion is formed by comparing the distance between two points S_i, S_j embedded in successive increasing dimensions. If their distance $d_D(S_i, S_j)$ in a space of dimension D is significantly different to their distance $d_{D+1}(S_i, S_j)$ in a space of dimension $D+1$, then they are considered to be a pair of *false neighbors*. Alternatively, if $(d_{D+1}(S_i, S_j) - d_D(S_i, S_j))/(d_D(S_i, S_j))$ exceeds a threshold (usually in the range of [10–15]), then the two points are false neighbors. The dimension D after which the percentage of false neighbors no longer decreases is chosen as D_e. After choosing T_d and D_e, the task of embedding the speech signal in a multidimensional phase space and reconstructing its *attractor* can be accomplished, as shown in Fig. 1 for a vowel and a fricative sound from the TIMIT database.

The embedding procedure enables us to reconstruct the geometrical structure of the original attractor of the system that produced the observed signal and to recover the determinism of an apparently random signal, in case this is possible. This allows us to construct accurate models of the system dynamics and to compute characteristics of the original system, like Lyapunov Exponents, which may prove to be useful features for the purpose of speech analysis, since they characterize the behavior of the system that produced the observed time-series.

III. Lyapunov Exponents

Lyapunov Exponents (LEs) are characteristics of a dynamical system that remain intact by the embedding procedure, and can be characterized as a measure of the system's "degree of chaos." A chaotic system is characterized by extreme sensitivity on initial conditions and rapid divergence of nearby orbits on its phase-space; LEs measure the exponential rate of divergence or convergence of orbits. A positive LE signifies divergence of nearby orbits and respectively a negative LE means convergence of orbits, while a LE equal to 0 is characteristic of a flow. For a conservative system the sum of LEs has to be negative, so that the orbits are bounded, while a chaotic system has at least one positive LE, which is the best known characteristic of chaotic systems, namely the exponential divergence of nearby orbits which results in long-term unpredictability.

Let $X(n+1) = f(X(n))$ be a discrete-time multidimensional dynamical system. In order to quantify the rates of divergence of its orbits, assume an initial state $X(0)$, which is slightly perturbed by ΔX to a new one $X'(0) = X(0) + \Delta X$. If we consider two orbits passing through $X(0)$ and $X'(0)$ at $t = 0$ the orbits will differ at the following step by $X'(1) - X(1) = f(X(0) + \Delta X) - f(X(0)) \approx J(X(0))\Delta X$ where $J(X(0))$ is the Jacobian of f at $X(0)$. By iterating for k steps we get:

$$\begin{aligned} X'(k) - X(k) &\approx J_k(X(0))\,\Delta X, \quad \text{where} \\ J_k(X(0)) &\triangleq J(X(k-1)) \cdots J(X(0)) \rightarrow \\ \|X'(k) - X(k)\|^2 &= \Delta X^T J_k^T(X(0))\, J_k(X(0))\, \Delta X \end{aligned}$$

where $\|\cdot\|$ is the Euclidean norm of a vector. In the limit $k \rightarrow \infty$, $[J_k(X(0))J_k^T(X(0))]^{1/2k}$ converges to a matrix known as the *Oseledec matrix*

$$\mathbf{OSL} = \lim_{k\rightarrow\infty} \left(J_k(X(0))\, J_k^T(X(0))\right)^{\frac{1}{2k}}. \tag{1}$$

The logarithms of the eigenvalues of the Oseledec matrix are equal to the LEs of the system whose dynamics are described by f. Since we usually do not have enough data to calculate the limit as $k \rightarrow \infty$, we use an approximation of **OSL** which involves only the first k matrices, from which we compute the so called *local* LEs [9].

The computation of LEs using experimental data is a nontrivial task for which various computational schemes have been proposed. Even though geometrical approaches to the problem are intuitively appealing, it turns out that these are extremely sensitive to noise and research in this direction has not advanced. The approach we follow is the one advocated in [23] and detailed in [9], which explicitly calculates the eigenvalues of the Oseledec matrix in (1) using an approximation F to the map f in order to calculate f's Jacobian.

Our main problem is that of data sparsity: contrary to previous attempts [19] to compute LEs of speech signals, where sustained vowels have been used (which facilitates the construction of local linear models), we used models that can learn the essential part of the system dynamics even when a short data set is available.

A second problem that arises when calculating the eigenvalues of the Oseledec matrix is its ill-conditioned nature which causes numerical inaccuracies. As a remedy to this problem, in [9], [23] the recursive QR decomposition technique has been applied, which breaks the problem into smaller ones: The matrix **OSL** can be viewed as the product of $2k$ matrices, $A_{2k}A_{2k-1}\cdots A_1$, for each of which we can use the recursive expression: $A_jQ_{j-1} = Q_jR_j$, $j = 1,2,\ldots,2k$, $Q_0 = I$, where Q_j, R_j result from the QR-decomposition of A_jQ_{j-1}; Q is an orthogonal matrix and R is upper diagonal. We can thus simplify the diagonalization of **OSL** as follows: We start with $A_1 = A_1Q_0 = Q_1R_1$ and $A_2Q_1 = Q_2R_2$, which yield $A_2A_1 = Q_2R_2R_1$. By iterating we get

$$A_{2k}A_{2k-1}\cdots A_1 = Q_{2k}R_{2k}R_{2k-1}\cdots R_1.$$

Q_{2k} tends to the identity matrix as $k \rightarrow \infty$ and therefore the eigenvalues of the LHS matrix shall be equal to the eigenvalues of the product of the $R_{1\ldots2k}$ matrices. Since $R_{1\ldots2k}$ are upper diagonal, the eigenvalues of their product shall equal the product of their diagonal elements. Thus, for the LEs we have

$$\lambda_i = \frac{1}{2k}\log\left(\prod_{j=1}^{2k} d_{ji}\right) = \frac{1}{2k}\sum_{j=1}^{2k}\log(d_{ji}), \quad i = 1\ldots D_e \tag{2}$$

where d_{ji} is the ith element of the diagonal of R_j.

Another problem we may encounter is due to the fact that the embedding dimension is not necessarily the intrinsic dimension of the system, but can be a larger one, namely the one which guarantees the unfolding of the attractor. As a by-product of the embedding process, more LEs than the true ones are computed and these are called *spurious* exponents. One approach [9], [23] to resolve this problem is to reverse the time sequencing of the observed data, i.e., to model the dynamics of the system $X(n+1) \rightarrow X(n)$ and compute once more the LEs of the "inverse" system. The true LEs should flip sign, since convergence of nearby orbits now becomes divergence and vice-versa. The spurious exponents are an artifact of the embedding process and should stay negative; this observation can be exploited as a validation technique for the LEs that are computed. This process, however, is prone to noise and care must be taken to determine the rejection thresholds for a spurious exponent. In addition, due to numerical problems large negative exponents rarely keep their magnitude while changing sign, when the time sequencing is reversed; as a consequence this procedure can be used only to validate a positive, zero, or small negative exponent.

An alternative approach to the same problem, presented in [24] and extended to deal with noisy and short time series in [25] is based on projecting the data in each neighborhood onto the most densely populated regions of the attractor. This is achieved using SVD analysis of the neighborhood data, which helps project out superfluous dimensions that are needed to *globally* embed the data.

IV. Nonlinear Modeling of System Dynamics

The task of predicting a chaotic signal that has been produced by a system whose dynamics are described by a function f defined on the reconstructed phase space can be formulated as that of multivariate function approximation based on the

input-output pairs $f(X(n)) = X(n+1), n = 1 \dots N$ where N is the number of available input-output pairs on the attractor. Numerous techniques have been proposed, ranging from simple extensions of the linear model to complex neural networks [22]. Our focus is on models that can approximate f even when supplied with short data-sets, since this is the main problem we are confronted with when dealing with a speech time series whose length does not exceed a phoneme's duration. Henceforth, unless otherwise mentioned, the output of the approximating functions is assumed to be multi-dimensional.

A. Global Polynomials

Global polynomials are a straightforward extension of the linear model. Instead of assuming that the next value of the state-space vector can be expressed as a linear combination of the previous values of the signal we can use an expression that uses higher order terms, i.e., a *global polynomial*. The parameters of a global polynomial that fits the data in an optimal way can be calculated using a family $\Phi = \{\phi_i\}$ of *orthonormal multivariate polynomials*

$$\phi_i(X) = \sum_{m=1}^{i} A_{m,i} \prod_{j=1}^{D_e} X_j^{e_j(m)}$$

$$\sum_{n=1}^{N} \phi_i\left(X(n)\right) \phi_k\left(X(n)\right) = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases}$$

$E(m) = \{e_1(m), e_2(m), \dots, e_{D_e}(m)\}$ is a 1-1 function $Z \rightarrow Z^{D_e}$ such that: i) $e_i(0) = 0\ \forall i$, ii) $e_i(k) + 1 \geq e_i(k+1) \geq e_i(k)\ \forall i$, iii) $\sum_i e_i(k+1) = 1 + \sum_i e_i(k)$. The expressions for orthogonality involving $A_{m,i}$ become complicated when dealing with multivariate polynomials, unless a principled approach is used, like the one in [26]. Assuming $A_{m,i}$ have been calculated, the normal equations [27] for the optimal expansion of f on the orthonormal basis Φ becomes

$$\begin{aligned} F(X) &= \sum_{i=1}^{M} C_i \phi_i(X) \\ &\approx f(X) \\ C_i &= \sum_{n=1}^{N} \phi_i\left(X(n)\right) f\left(X(n)\right) \\ &= \sum_{n=1}^{N} \phi_i\left(X(n)\right) X(n+1). \end{aligned}$$

Global polynomials are well suited when a crude model of the dynamics is desired, using a very small number of parameters and when very few input-output pairs are available; however, they are inappropriate for accurately capturing the dynamics of the system.

B. Local Polynomials

Local polynomials constitute the mainstream approach to approximating the dynamics of a system on its reconstructed attractor. The main idea is to construct for each point $X(i)$ on the attractor a linear model, using information about the behavior of neighboring points and minimizing

$$E_i = \sum_{j=1}^{k} \left\| X\left(n_j^i + 1\right) - \left(AX\left(n_j^i\right) + B\right) \right\|^2 \tag{3}$$

which results in the normal equations [27]; $X(n^i_{1 \dots k})$ are the k nearest neighbors of $X(i)$. The phase space is thus broken up in small neighborhoods, for each of which the system dynamics are expressed through a local model.

A refinement of this method [24], [25] uses local linear models in D_p dimensions, where $D_p < D_e$, by projecting the input and output points on the most densely populated directions in the two neighborhoods. If D_p is the intrinsic dimension of the system, only the necessary part of the system dynamics is modeled, using fewer points and better conditioned matrices in the normal equations. However, it is hard to determine the correct D_p when few and noisy points are available, given that the attractors of the systems are usually curved manifolds. This results in inaccurate dimension estimates when points from a relatively large neighborhood are used to determine D_p, since for its calculation it is assumed that locally the attractor resides in a linear subspace of the embedding space.

Other refinements that have been proposed, like clustering the data and then constructing local linear models for each cluster [9], can be seen as the limiting case of TSK models (presented later) when the spread of the membership functions tends to zero, and so they will not be presented here.

C. RBF Networks

RBF networks rank among the most popular approaches to multivariate function approximation, combining several appealing and useful properties, like continuity, differentiability and the ability to locally approximate f while minimizing globally the prediction error. Suppose we have found M clusters of data points on the reconstructed attractor by using, e.g., the k-means clustering algorithm, and we use the center C_i of each cluster center as the center of a Radial Basis Function like the Gaussian. We can then approximate f by the function

$$F(X) = \sum_{i=1}^{M} B_i \cdot G_i(X), \quad G_i(X) = \exp\left(-\frac{\|X - C_i\|^2}{\sigma_i^2}\right). \tag{4}$$

In the above equation B_i can be interpreted as a constant model of the system dynamics around the center of the ith cluster. The B_i's can be calculated by minimizing the mean square prediction error

$$E = \sum_{n=1}^{N} \left\| X(n+1) - \sum_{i=1}^{M} B_i \cdot \exp\left(-\frac{\|X(n) - C_i\|^2}{\sigma_i^2}\right) \right\|^2$$

which yields again the normal equations. Determining the number of clusters and their spreads is usually based on heuristics, which can be complemented by statistical methods for model validation like cross-validation tests.

D. TSK Models

TSK [28] models constitute the Fuzzy-Logic approach to function approximation. TSK models, even though relatively simple and straightforward, can approximate complicated functions using a small number of parameters. In brief, the idea is to partition the input space of the function in M fuzzy sets, i.e., sets having no crisp borders, to fit a local model to each subdivision of the input space and to express the approximation of the function as a weighted average of local models:

$$F(X) = \frac{\sum_{i=1}^{M} \mu_i(X) l_i(X)}{\sum_{i=1}^{M} \mu_i(X)} \tag{5}$$

where μ_i measures the degree of membership of X in the ith fuzzy set and $l_i(X)$ is the local model of the system dynamics for the ith fuzzy set. If constants are used as local models, i.e., $l_i(X) = B_i$, the model is called a TSK-0 model. If we use a linear expression as a local model, i.e., $l_i(X) = A_i X + B_i$, the model is called TSK-1. For the special case of a TSK-0 model with radial membership functions we obtain a model equivalent with the Normalized Radial Basis Function network proposed in [29] and used for chaotic time series prediction in [30], [31]. The models proposed in [32] under the names of Weighted Constant/Local Predictor can be identified with the TSK-0/1 models, respectively. One can also interpret TSK models as a mixture-of-experts architecture [33] for which efficient Expectation Maximization [34] training algorithms have been proposed.

As with RBF networks the number of membership functions and their spreads (if they are Gaussians) have to be determined, as well as A_i, B_i. If the centers and the spreads are known, the optimal A_i and B_i can be calculated by the normal equations and if A_i and B_i are considered constant, the centers and the σ's can be learned using a gradient descent algorithm. Combined with the widely used SVD-QR [27] technique for elimination of unnecessary membership functions we have a powerful technique for function approximation which is very similar to the ANFIS system [35]. Since this is the model we finally used in our experiments with speech signals, some implementation details are given in a following section.

E. Support Vector Machines (SVM)

SVMs for regression [36] are based on novel ideas from the field of machine learning and have proven to give good results when applied to chaotic signals [37]. All the previous approaches are prone to overfitting, that is, instead of learning the function f, the predictor may learn the training data. This may result in small prediction MSE with the training set, but unless care is taken our predictors shall fail to capture the system dynamics. Instead, SVMs are constructed so as to minimize the *expected generalization error*, so a fairly accurate model of the system dynamics that is not biased in favor of the training data can be made.

Assume $f : R^{D_e} \rightarrow R$ can be expressed as $f(X, W) = \sum_{i=1}^{\infty} W_i \phi_i(X)$. The method proposed in [36] for approximating f suggests minimizing with respect to W the cost functional

$$C(W) = \frac{1}{N} \sum_{i=1}^{N} |y_i - f(X(i), W)|_\epsilon + \gamma W^T W \tag{6}$$

where $y_{1\ldots N}$ are the desired outputs and $|\cdot|_\epsilon$ punishes only deviations of $f(X(i), W)$ from y_i larger than ϵ by $|y_i - f(X(i), W)| - \epsilon$. The first term in the sum punishes prediction error while the second term accounts for the complexity of the predictor. A small norm of W means that fewer terms will contribute significantly in the expression for f, while γ is a constant determining the tradeoff between predictor complexity and accuracy. By using the ϕ_k's we can form *inner-product kernels* of the type $K(X, Y) = \sum_{k=1}^{\infty} c_k \phi_k(X) \phi_k(Y)$. Then, it can be shown (see references in [36]) that the function minimizing C is of the form

$$F(X) = \sum_{i=1}^{N} (\alpha_i - \alpha_i^*) K(X, X(i)) + b \tag{7}$$

where $\alpha_i, \alpha_i^* \geq 0$, $\alpha_i \cdot \alpha_i^* = 0$. Examples of inner product kernels K include the Gaussian function, splines, and the sigmoid activation function. By substituting the expression for F in (7) in (6) we get an optimization problem, the dual of which is

$$\begin{aligned} \text{Maximize}: \quad & -\frac{1}{2} \sum_{i,j=1}^{N} \left[(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*) K(X(i), X(j)) \right] \\ & - \epsilon \sum_{i=1}^{l} (\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i (\alpha_i - \alpha_i^*) \\ \text{subject to}: \quad & \begin{cases} \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) = 0 \\ \alpha_i, \alpha_i^* \in [0, C], C = \frac{1}{2\gamma} \end{cases}. \end{aligned}$$

The nature of this optimization problem leads to using a sparse set of data points $X(i)$ to approximate the function f, by punishing nonzero α_i s and by including only those $K(X, X(i))$ in the final expression (7) for which $\alpha_i^* - \alpha_i \neq 0$. Those $X(i)$ for which α_i or α_i^* are different from 0 are called *Support Vectors* and are the optimal set of points to express F as in (7).

Despite its elegance and successful application to many fields of machine learning, the SVM model is not a panacea; choosing ϵ, γ, the kernel used and the kernel-specific parameters are still done heuristically, while our experience has shown that solving the optimization problem can become slow, even when using efficient algorithms.

F. Dealing With Short Time-Series

In all of the previous analysis it is assumed that the dynamical system that produces the observed time series is stationary; consequently only phonemes can be used to construct models of the dynamical system that produced them, since utterances of whole words are inappropriate. Therefore only short time series are available to approximate high-dimensional functions, which necessitates resorting to more parsimonious models than the commonly used local linear models.

For example, suppose the observed time series, $x(n)$, has been embedded in 6 dimensions and we have 800 pairs $X(n) \rightarrow$

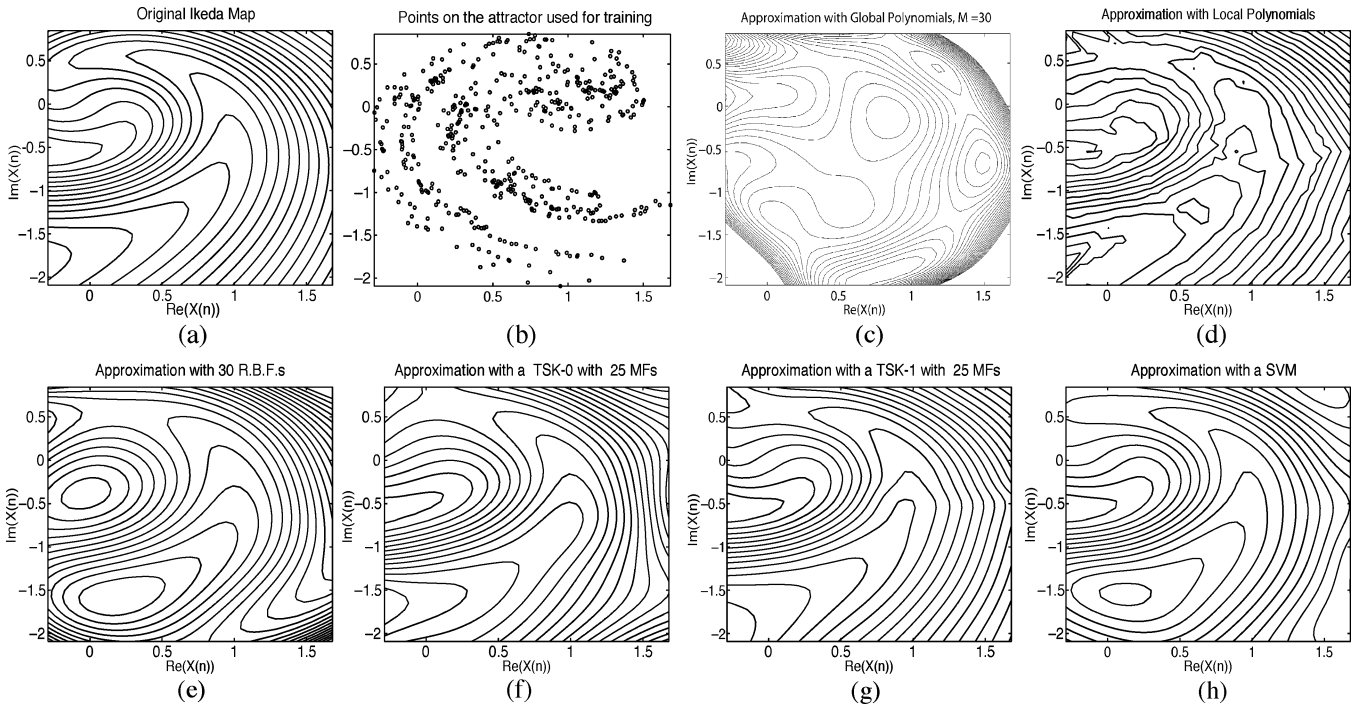

Fig. 2. Approximation quality of the real part of the Ikeda map using the models presented in the text. For illustration we use the level sets of the original and approximating functions, namely the sets $\{X : f(X) = c_i\}$, for $c_i = -2.7 + 0.13i$ $i \in \{0 \dots 25\}$. (a) Level sets of the real part of the Ikeda map (b) 500 data points, lying on the attractor of the system that serve as the training set (c)–(h) level sets of the approximations to the Ikeda dynamics, using the models described above, trained with 500 points. When a region is left blank (as, e.g., with global polynomials) this means the outputs have surpassed a given threshold. Accumulation of level sets in a region means that the function has a high gradient there and vice versa.

$X(n+1)$ (a typical reconstructed attractor of a phoneme). Even if we consider that our data are clean and the system is perfectly deterministic, at least $6 + 6 \times 6 = 42$ neighboring data points are necessary to build a linear approximation of the system dynamics around each point, which is 1/20 of the attractor points. Therefore, in order to construct local models in sparsely populated areas of the attractor it is necessary to use points that lie outside the actual neighborhood of $X(n)$ which results in inaccurate local models of the system dynamics. In order to cope with noise and/or system randomness even more points are necessary, making the problem worse.

As a first step toward choosing an appropriate model for nonlinear speech modeling, we performed extensive experiments to evaluate the ability of these models to approximate nonlinear functions using a limited set of input-output pairs. A characteristic example is depicted in Fig. 2, where 500 input-output pairs of the complex Ikeda map

$$X(n+1) = 1 + 0.9\, X(n) \exp\left(0.4 \cdot j - \frac{6 \cdot j}{1 + \|X(n-1)\|^2}\right)$$

have been used to learn the dynamics of the system using the models presented previously. We observe that SVM networks and TSK models combine smoothness and fidelity to the observed data. Instead local polynomials and RBF networks are inaccurate in areas where there are not many data available, even though the latter are much smoother. Global polynomials give a smooth, yet inaccurate approximation, especially in areas where there have been no points available. It is not only the value of the function at a specific point that we are interested in approximating correctly, but also its Jacobian at that point, so a smooth approximation is also needed. We have made similar experiments using noisy input-output pairs, larger data sets, different nonlinear mappings, and the results are qualitatively similar. The parameter settings used to produce these approximations have not been chosen manually for every experiment, but were based on some heuristics that gave reasonable performance when used with other test signals and time series of different lengths.

A second and more quantitative test we performed was that we probed our models with synthetic signals, and compared the Lyapunov Exponents (LEs) that were calculated using each model. For this purpose we used embeddings of chaotic, sinusoidal and random signals and checked the LEs estimated for a wide range of parameters settings (number of clusters, neighbors, learning iterations etc.). We used time series consisting of 1000 samples from the Lorenz System, the Ikeda System, the Henon System [9] as well as a two-tone sinusoidal signal and a gaussian noise signal. We have considered that a model is not appropriate for LE estimation when it gives ambiguous results, which means that i) there are no clearly validated exponents, ii) there are more validated exponents than there should be, or iii) their estimates are not close to the true values of the exponents. RBF and TSK-0 models worked well when many data points were available but broke down in more challenging situations. Local and global polynomials, lying on the two extremes of model complexity, proved to have the worst performance for the purpose of computing LEs from a short time series. SVM networks gave very good results, as long as a cross-validation procedure was used, that would check the cross-validation error

of the model, using a separate test set. Since many parameter settings for SVM networks are ad-hoc, a single heuristic would not always work. TSK-1 models on the other hand gave good results with all types of signals, using some heuristics described in the following section, while taking a relatively short time to train. This led us to choose them among the previously presented repertoire of models for the purpose of speech modeling.

V. Applications to Speech

From a dynamical systems perspective a speech signal can be regarded as an observable of the speech production system, which can be used to uncover and model its dynamics, using the previously presented techniques. Modeling these dynamics can be useful for speech prediction/synthesis, while invariants of the dynamical system, like its Lyapunov Exponents or the fractal dimension of its reconstructed attractor could be useful for speech recognition.

A. Dynamics Model Implementation Details

Based on the experiments presented in the previous subsection, we decided to use TSK-1 models for the purpose of nonlinear speech analysis on the reconstructed attractor, because they seem to provide the best tradeoff between model complexity and performance.

TSK models are based on building simple models of the system dynamics around clusters of data, and expressing system dynamics at each point as a combination of the dynamics corresponding to each of the clusters it belongs to. Typically some simple clustering algorithm like K-means is used to locate the centers of the clusters, and the subsequent steps of determining the TSK model parameters are assumed to make up for any initial inaccuracies of the clustering procedure.

For the case of nonlinear dynamical systems the data occupy a small portion of the embedding space and their variance depends on their location on the attractor. Some more sophisticated data clustering procedures are therefore necessary, which can simultaneously assign in a soft way each data point to some clusters and determine the parameters used to express the membership functions. A natural framework for dealing with such problems is the expectation–maximization (EM) algorithm [34], where at each step the parameters of the membership functions are used to estimate the membership of the data to each cluster, and subsequently these parameters are determined so as to maximize the likelihood of the data, given their memberships. For the sake of simplicity we used a single parameter expressing the spread of the multidimensional Gaussian membership functions, as in (4), even though it is obvious that when expressing the plane-like distribution of the data around each cluster with Gaussian distributions one should use a covariance matrix that is neither diagonal, nor has equal elements. In order to somehow make up for this inaccuracy, the spreads of the Gaussian filters we used were set equal to the maximal variance of the data belonging to each cluster along all axes; otherwise the fact that the data lie locally along a plane results in lower spread estimates, systematically causing modeling errors. Finally, if some cluster centers lie closer to each other than a threshold, fixed at 0.01 for data normalized to lie in [0, 1], they are merged into a single one. The "correct" number of clusters is usually hard to find so we have used a heuristic prescription which fits well the manually estimated optimal number of clusters, for data sets of the size we used in our experiments (500–2000 points): $M = 10\log_2(N/125)$, where N is the number of points on the attractor, so for 10^3 points we have 30 clusters, for 500 points 20 clusters and so on.

After the initial clustering stage, the design matrix of the activation signals used in the normal equations is formed, on which the SVD-QR [27] procedure is used to determine which columns can be left out in order to render the matrix well-conditioned. The threshold on the singular values of the design matrix was set equal to 1/100 of the maximal singular value of the matrix. After these initial settings, a gradient-descent algorithm has been used in order to fine-tune the parameters of the predictor model; a robust variant of the back-propagation algorithm has been very helpful, namely Resilient Propagation (RPROP) [38], which uses only the sign of the error gradient (and not its value) to update the network parameters. The parameters for this algorithm were set to $\nu^- = .5$, $\nu^+ = 1.05$ and ten iterations of the RPROP algorithm were used; some readily derivable expressions for the partial derivatives of the error functions w.r.t. the model parameters can be found in [39]. After updating the centers and spreads the local linear models are reestimated using the normal equations, thereby resulting in a loop that decreases the prediction error, without overfitting thanks to the preceding SVD-QR procedure that eliminates superfluous activation signals. In order to avoid numerical problems, after each updating step the spreads of the Gaussian functions are thresholded by a lower and upper bound, set to 0.01 and 0.25 respectively, for data normalized to lie in [0, 1].

An interesting alternative procedure to train a TSK model relies on the EM algorithm described in [33] since, as already mentioned, TSK models are equivalent with Mixture-of-Experts architectures [33] for which the EM algorithm applies. We note that for this specific architecture an on-line EM algorithm has been proposed in [40], thereby offering the possibility to continuously update the model of the system dynamics which could facilitate the nonlinear prediction and analysis of continuous signals.

B. Experiments With Lyapunov Exponents

In order to guarantee the time invariance of the time series representing the observable of the system we wish to model, we applied our methods to isolated phonemes from the TIMIT database, that is short time series corresponding to the same sound so that one can assume the underlying dynamics are time-invariant. Even though in previous work, like [13], [19] much more stringent criteria have been used in order to guarantee the stationarity of the analyzed signal, in our case we tried to reconcile the fact that we had to model naturally uttered speech sounds with the desire to have stationary signals, so there are still some transients, like at the beginnings or endings of vowels. In that case we expect the estimated values of the LEs to be close to the ones that would be estimated using the stationary part of the signal, based on the fact that most of the terms involved in the averaging in (2) are estimated at points with stationary dynamics. In practice, manually cutting off the transient parts of

TABLE I
THREE FIRST VALIDATED LYAPUNOV EXPONENTS FOR SPEECH PHONEMES

Phon./LEs	/aa/(70)	/eh/ (64)	/ih/ (59)	/ow/ (56)	/w/ (39)	/m/ (36)
λ_1	0.047 ± 0.028	0.093 ± 0.040	0.084 ± 0.045	0.069 ± 0.042	0.036 ± 0.024	0.029 ± 0.034
λ_2	-0.004 ± 0.018	-0.014 ± 0.027	-0.001 ± 0.041	0.052 ± 0.025	-0.009 ± 0.015	-0.096 ± 0.068
λ_3	-0.078 ± 0.038	-0.139 ± 0.048	-0.156 ± 0.079	-0.083 ± 0.052	-0.096 ± 0.042	-0.289 ± 0.142
Phon./LEs	/r/ (52)	/l/ (39)	/f/ (50)	/s/ (102)	/b/ (37)	/t/ (35)
λ_1	0.074 ± 0.038	0.048 ± 0.035	-0.561 ± 0.249	-0.312 ± 0.157	-0.012 ± 0.152	-0.296 ± 0.254
λ_2	-0.012 ± 0.030	-0.013 ± 0.022	-0.772 ± 0.260	-0.504 ± 0.172	-0.047 ± 0.277	-0.492 ± 0.293
λ_3	-0.118 ± 0.096	-0.099 ± 0.069	-0.997 ± 0.274	-0.725 ± 0.217	-0.361 ± 0.303	-0.710 ± 0.323

Next to each phoneme is given the number of time series from which the statistics have been calculated; for robustness the median and the mean absolute deviation from the median are used instead of the mean and the standard deviation. The phonemes have been uttered by 11 speakers. For all phonemes, approximately the same number of pronunciations is used from every speaker. For all the vowels/semivowels in this table the exponents have been validated using the LEs of the inverse time series. For fricatives and unvoiced stops these are not validated, but used merely as features for classification; no conclusions should be drawn from these. One should note the increase in the variation of the LEs for the latter classes.

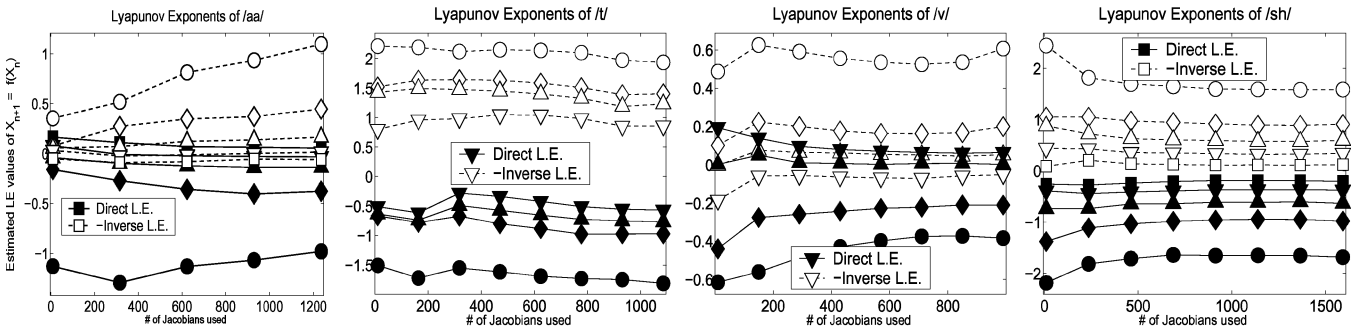

Fig. 3. Direct and inverse Lyapunov exponents of a vowel, an unvoiced stop sound, a voiced and an unvoiced fricative. Please note that the only validated Lyapunov exponents are the ones close to zero, where a negative LE validates a positive LE.

vowels resulted in qualitatively the same estimated LEs. The results of our methods for *inherently* nonstationary phonemes like stop sounds should be treated with care to ensure we can exploit the results for the rest of the phonemes where our approach is expected to work well.

The time-series data have been normalized to lie in the interval [0, 1] and have been embedded using the procedure described in Section II, resulting in typically low-dimensional attractors, with dimensions lying in the range [4–8]. Specifically we used at least 35 pronunciations of each phoneme belonging to the following classes:

- Vowels: /aa/, /ae/, /ao/, /er/, /eh/, /ow/, /oy/, /ih/, /ix/
- Fricatives: /f/, /s/, /sh/, /th/, /z/, /dh/, /v/
- Stops: /p/, /k/, /t/, /b/, /d/, /g/
- Semivowels: /r/, /l/, /w/, /j/
- Nasals: /m/, /n/

by the following speakers:

- Male: “dab,” “jsw,” “reb,” “rjo,” “sjs,” “stk,” “wbt.”
- Female: “aks,” “dac,” “elc,” “jem.”

from dialect 1 in the TIMIT database. For all phonemes, approximately the same number of pronunciations is used from every speaker. The length of the time series ranges from 500 samples to 2000 or more, with a typical sample size being around 800 points. We discarded shorter time series, since they typically gave erroneous results when using synthetic data.

1) Lyapunov Exponents of Speech Signals: We computed the Lyapunov Exponents for different phonemes using TSK-1 models, in order to draw some meaningful and useful conclusions about their values. The main results are the following (see also Table I and Fig. 3):

- Vowels typically have a small positive exponent, one equal to zero and one or more negative exponents.
- Unvoiced fricatives give no validated exponents; in particular, we observed a phenomenon that happens with noise signals: all the direct and inverse exponents are negative (hence no exponents are validated). This may be a consequence of the highly noisy nature of unvoiced fricatives that causes the methods of chaotic analysis to break down. Even though more data and higher embedding dimensions could lead to accurate models of the system dynamics that would potentially give some validated exponents, it was not possible using the short time series available.
- Validated exponents of voiced fricatives are usually higher than those of vowels; this is somehow expected since fricatives are less predictable than vowels. However, it seems that a strong noise component in the signal usually causes none of its exponents to be validated.
- Stop sounds were found to be vaguely separated in two clusters: the first consisted of {/p/, /k/, /t/} (unvoiced stops) and the second was formed by {/b/, /d/, /g/}(voiced stops). For the first group it was impossible to find any validated exponents, while the exponents of the second group were validated occasionally. Really short and nonstationary time series correspond to stop sounds and it is therefore not safe to

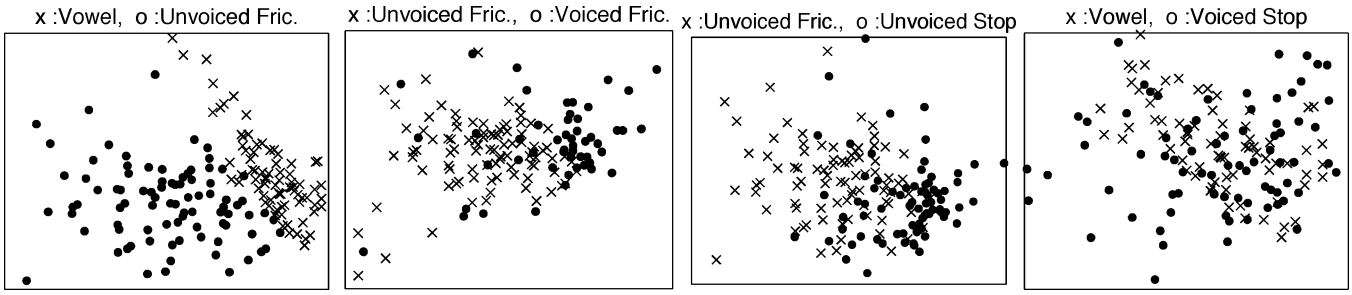

Fig. 4. One excellent, one good, one average, and one bad class separation using only LEs. The axes represent the two principal components of the LE data.

draw any conclusions about the dynamics of the system based on the LEs; they still can be useful, however for classification.

- Semivowels typically had one positive LE, another almost equal to zero and one or more negative LEs. Nasals have similar features, with a typically smaller maximal LE.

More sophisticated methods such as cleaning a signal on its reconstructed attractor or cutting off the silent part before a stop-sound could possibly be applied in order to obtain validated LEs from stop sounds and unvoiced fricatives, but using any of them would insert some bias into the recognition process unless we could apply these methods to all the phonemes without affecting the derived results for the rest. Even when none of these methods is applied, the fact that no LEs are validated may prove to be useful information since this distinguishes stop sounds and unvoiced fricatives from vowels and occasionally from voiced fricatives. One can conjecture that the absence of validated LEs for a signal is indicative of its randomness and/or non-stationarity.

Separation of the phoneme classes is possible in some cases using the first three Lyapynov Exponents of phonemes, as can be seen in Fig. 4. The separation is almost perfect for vowels/unvoiced fricatives, but is not that successful for all of the class pairs. In this figure and the classification experiments described below, in case no validated exponents can be found, they are replaced by the largest nonvalidated direct exponents.

It should be mentioned that even though in most of the previous work clean and sustained signals have been used, there is little agreement about the values of LEs of speech signals. In [13], [19] vowels are reported as having their Maximum Lyapunov Exponent (MLE) almost equal to zero, while in [14], [15], [41] their MLE was found positive. Also, in [13], [14] the MLE of fricative sounds was found positive, while in most of our experiments we did not manage to validate any of them using models that are robust to noise. Apart from the quality of the data on its own, which may strongly influence the quality of the LE estimation [10], we believe this controversy is partly due to the inherent difficulty of calculating LEs, and partly due to the large variety of embedding, modeling and LE estimation procedures that exist which may influence the quality of the results. For example, in a systematic previous examination of nonlinear dynamics in speech signals [19] where a different embedding and modeling technique was used, the computation of LEs led to no significantly positive LEs for all of the tested vowels. One could assume that this is due to the SVD embedding procedure that acts as a low-pass filter or due to the algorithm used for LE calculation, which involves various parameter tuning steps. However, according to the authors the embedding procedure does not significantly influence their results, while the algorithm used to calculate LEs has been experimentally validated in another publication [25] by the same authors. The question which signal embedding procedure is the most appropriate has concerned researchers in the field of nonlinear dynamics modeling and to the best of our knowledge there is no general consensus about which method is the "best." On an empirical basis we would argue that for speech signals which have an inherent amount of noise it is better to avoid prematurely filtering the signal, and to rely on the function approximation part for the purpose of distinguishing the true underlying dynamics of the signal from the observation or system noise. In the field of machine learning/function approximation, regularization techniques or the SVD-QR procedure are employed to avoid overfitting, thereby learning only the consistent aspects of the system dynamics.

What we want to point out by the above comparison is that calculating the LEs of a time series is not something straightforward that can be accomplished using an off-the-shelf procedure, but depends on the assumptions one makes about the data and on every single component used in the procedure one employs. Since from the very start our main focus has been on models that can tolerate noise in the signals, without resulting in spurious exponents, we can be confident in our results, at least for the data set we examined which consists of short and naturally uttered phonemes.

We should note that the estimation of a positive Lyapunov Exponent, which is characteristic of a chaotic dynamical system cannot be used as a proof for the existence of chaos. Before one can decisively conclude that the speech signal is chaotic, one should use surrogate data sets and perform more principled tests [10].

2) Classification Experiments: In order to somehow quantify the meaningfulness of the calculated LEs, we examined whether they can be used to classify a phoneme into one of n large phoneme classes like "fricatives," or "/a/-like vowels." We constructed a directed acyclic graph (DAG) multiple-class classifier that uses a voting strategy to aggregate the classification results of $n(n-1)$ binary classifiers, that separate phoneme classes. This classifier works by assigning a vote to the winning class for every comparison being made and classifying the input datum as belonging to the class that gets the most votes- in case of ties each class gets half a vote.

TABLE II
CLASSIFICATION RATES FOR VARIOUS CLASS COMBINATIONS

Classes†/Features	V. / F.	uv.F / S.	V. / uv.S.	F. / NLG.	NLG. / S.	V. / F./ S.
MFCC	96.7 ± 3.8	86.7 ± 9.1	90.6 ± 8.4	96.5 ± 3.1	88.3 ± 7.0	85.9 ± 5.2
LEs	97.6 ± 1.2	75.2 ± 12.3	84.2 ± 7.1	94.1 ± 9.8	71.2 ± 9.1	55.6 ± 5.7
MFCC+LEs	99.1 ± 0.1	87.6 ± 7.6	86.6 ± 11.0	97.2 ± 2.3	82.7 ± 7.8	86.4 ± 5.1

†V.: Vowels F.: Fricatives, S.:Stops, NLG: Nasals, Liquids, Glides v/uv: voiced/unvoiced

The classifier used to perform binary comparisons was an AdaBoost classifier [42] using a linear threshold unit as a weak classifier; AdaBoost classifiers are characterized by their simplicity and good performance, which has made them very popular in the pattern recognition community. We used 50 iterations of the boosting procedure, using as a training set 80% of the available data, chose the classifier based on a validation set equal to 10% of the data and the performance of the classifier was evaluated on the rest 10% of the data. This was repeated for 30 permutations of the data to ensure the validity of the results.

Some classification results for various combinations of classes are shown in Table II for: (a) The first three LEs, estimated using a TSK-1 model, (b) The 16 first Mel-Frequency Cepstrum coefficients (MFCCs) (c) The combination of the two. We first observe that in almost all cases, the results are significantly better than $1/n$ where n is the number of classes, which would correspond to the random choice of one class. However, there can be no distinction, e.g., between vowels, considered as a unique class and semi-vowels based on their LEs, since as mentioned previously they are very similar. Using a nonparametric statistical test (Wilcoxon Matched-Pairs Signed-Ranks) the hypothesis $H_0 : p = 1/n$ was rejected with a significance level $\alpha = .0001$ for all of the class comparisons presented here, except for the comparison vowels/semi-vowels. The MFCC results are better than those achieved using solely LEs, however by combining both we sometimes achieve an increase in the correct classification rate. In this case, however, the upward trend of the means cannot lead to clear conclusions, due to the relatively high variance in the estimates. The significance tests reject the hypothesis H_0 with a significance level α higher than .4 in all cases. Adding LEs to MFCCs for class comparisons where LEs do not have good performance on their own results in a decrease in the correct classification rate, as, e.g., for the comparison between vowels and unvoiced stops. It is worth noting finally that when no validation with the inverse exponents was used at a preprocessing stage, the results deteriorated.

Even though the results presented here are somehow limited (e.g., no speaker dependency of the results is examined), they testify first of all that the LEs estimated by our model are meaningful and correlated with the class of the phoneme they came from. Further, they offer a promising research direction, toward the incorporation of LEs in speech recognition tasks [41], [43].

C. Prediction of Speech Signals

The LPC model, which is most commonly used for speech prediction and coding, can be seen as a very special case of a global polynomial: the LPC-model equation, $s(n) = \sum_{i=1}^{M} a_i s(n-i) + a_0$ can be considered as a linear approximation of the system dynamics f, where the data have been embedded in a M-dimensional phase-space with $T_d = 1$, by default. Increasing the complexity of the LPC predictor $(M \rightarrow M+1)$ can be interpreted as embedding the data in a higher dimensional space; when M is still small this results in fewer false neighbors (see Section II) and hence more predictable dynamics, but after a certain embedding dimension, when the percentage of false neighbors no longer decreases, there is no actual contribution to the predictability of the system dynamics.

On the other hand, the approach of approximating the system dynamics on the attractor is more adaptive to the available data: we first determine the embedding dimension where all (or most) false neighbors are eliminated in combination with a more proper choice of T_d and then construct models that attempt to approximate f with increasing complexity of the predictor meaning more parameters involved in the approximation of f (rather than embedding the data in higher dimensions).

It could be therefore assumed that by reconstructing the attractor of the system one could learn all of the deterministic nature of the underlying system and thereby get a lower prediction error than the one attained by the linear model. In the experiments we performed we observed that this is true, but at the cost of a larger number of parameters: most of the predictors presented previously include many parameters, and it is therefore expected that they achieve a lower prediction error. In addition, for the reconstructed state vector $X(n) = [s(n), s(n - T_d), \ldots s(n - T_d(D_e - 1)]$ we do not use at all the samples between $s(n-1)$ and $s(n - T_d + 1)$ which are usually among the most informative about $s(n+1)$. This makes it possible to embed the system in a *low dimensional* attractor, but throws away the most useful points for the prediction of $s(n+1)$. On the contrary, the LPC model uses first of all the nearest points which are the most informative and this results in a very low MSE with a small number of parameters. It has to be added that when $s(n)$ is the time series of a vowel or a voiced fricative, a scatter plot of $\{s(n), s(n-1), s(n-2)\}$ shows that most of the points lie approximately on a plane, i.e., $s(n)$ is approximately a linear function of $s(n-1)$ and $s(n-2)$, which is indicative of the appropriateness of the LPC model for speech coding.

However, when estimating the cross validation error, where data that have not been presented at the training stage are used to estimate the performance of the model, nonlinear predictors like TSK-1 models give consistently better prediction results, while the LPC model gives large prediction errors whenever a spike is present. Therefore, our approach could be more appropriate than the linear model for tasks demanding an accurate model of the speech signal, that would not use an error signal to make up for the prediction error. TSK, RBF and SVM models allow techniques of machine learning like cross-validation and

regularization to be applied (see, e.g., work in [18], [20] for regularization and RBF networks), which allow the construction of models that are neither susceptible to over-fitting nor too simple like LPC.

Work in the direction of speech synthesis using nonlinear models [16]–[21] is based mainly on radial basis function networks and local polynomial models, which are used to predict the future state of the speech production system using the previous observations. Interesting results have been obtained and we believe that using some more sophisticated models like the TSK even better results could be achieved. Our preliminary experiments in this direction are promising and we seek to further continue research in this direction.

VI. Conclusions and Future Research

In this paper nonlinear models of the speech production system have been presented, that are constructed on the reconstructed attractor of speech signals. These models have been used for the extraction of features that can help with the characterization of chaotic systems, namely Lyapunov Exponents. Experiments with Lyapunov Exponents of various phoneme classes have shown that they can be useful features for speech sound classification, proposing a promising area of future research.

Even though we cannot conclude whether the speech production system is chaotic or not using as sole evidence the values of Lyapunov Exponents, we believe it is promising that in most of the experiments run the Lyapunov Exponents of most of the voiced phonemes we experimented with (vowels, glides and nasals) were estimated to be above zero. Even if the speech production system is not chaotic, nonlinear function approximation models can be useful for the analysis of speech signals and deserve further attention.

An interesting future research direction seems to us the introduction of a continuous adaptation procedure to the TSK models, using the on-line EM algorithm proposed in [40], so as to facilitate the modeling and analysis of continuous speech signals. This would allow their incorporation in speech coding, synthesis or recognition systems, since only small incremental changes would be necessary for new frames, allowing an efficient computation and transmission of model parameters.

Acknowledgment

The authors wish to thank the two reviewers for their comments which helped improve the quality of the paper.

References

[1] H. M. Teager and S. M. Teager, "Evidence for nonlinear sound production mechanisms in the vocal tract," in *Speech Production and Speech Modeling*, W. Hardcastle and A. Marchal, Eds. Bonas, France, July 1989, vol. 55.

[2] J. F. Kaiser, "Some observations on vocal tract operation from a fluid flow point of view," in *Vocal Fold Physiology: Biomechanics, Acoustics, and Phonatory Control*, I. R. Titze and R. C. Scherer, Eds. Denver, CO: Denver Center for Performing Arts, 1983, pp. 358–386.

[3] T. J. Thomas, "A finite element model of fluid flow in the vocal tract," *Comput. Speech Lang.*, vol. 1, pp. 131–151, 1986.

[4] R. S. McGowan, "An aeroacoustics approach to phonation," *J. Acoust. Soc. Amer.*, vol. 83, no. 2, pp. 696–704, 1988.

[5] G. Richard, D. Sinder, H. Duncan, Q. Lin, J. Flanagan, S. Levinson, M. Krane, S. Slimon, and D. Davis, "Numerical simulation of fluid flow in the vocal tract," in *Proc. Eurospeech*, 1995.

[6] A. Barney, C. Shadle, and P. Davies, "Fluid flow in a dynamical mechanical model of the vocal folds and tract: part 1 & 2," *J. Acoust. Soc. Amer.*, vol. 105, no. 1, pp. 444–466, Jan. 1999.

[7] P. Maragos, A. Dimakis, and I. Kokkinos, "Some advances in nonlinear speech modeling using modulations, fractals, and chaos," in *Proc. Int. Conf. on DSP*, Santorini, Greece, Jul. 2002.

[8] V. Pitsikalis, I. Kokkinos, and P. Maragos, "Nonlinear analysis of speech signals: Generalized dimensions and Lyapunov exponents," in *Proc. of Eurospeech*, Geneva, Switzerland, Sep. 2003.

[9] H. Abarbanel, *Analysis of Observed Chaotic Data*. New York: Springer-Verlag, 1996.

[10] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1997.

[11] T. F. Quatieri and E. M. Hofstetter, "Short-time signal representation by nonlinear difference equations," in *Proc. ICASSP-1990*, vol. 3, Albuquerque, NM, April 1990, pp. 1551–1554.

[12] H. P. Bernhard and G. Kubin, "Speech production and chaos," in *Proc. XIIth Int. Congress Phonetic Sciences*, Aug. 1991.

[13] S. Narayanan and A. Alwan, "A nonlinear dynamical systems analysis of fricative consonants," *J. Acoust. Soc. Amer.*, vol. 97, no. 4, pp. 2511–2524, 1995.

[14] A. Kumar and S. Mullick, "Nonlinear dynamical analysis of speech," *J. Acoust. Soc. Amer.*, vol. 100, no. 1, pp. 615–629, 1996.

[15] I. Tokuda, R. Tokunaga, and K. Aihara, "A simple geometrical structure underlying speech signals of the Japanese vowel /a/," *Int. J. Bifurcation Chaos*, vol. 6, no. 1, pp. 149–160, 1996.

[16] M. Birgmeier, "A fully Kalman-trained radial basis function network for nonlinear speech modeling," in *Proc. Int. Conf. Neural Networks*, Nov. 1995, pp. 259–254.

[17] G. Kubin, "Synthesis and coding of continuous speech with the nonlinear oscillator model," in *Proc. ICASSP-1996*, Atlanta, GA, 1996, pp. 267–270.

[18] E. Rank and G. Kubin, "Nonlinear synthesis of vowels in the LP residual domain with a regularized RBF network," in *Proc. of the 6th Int. Work-Conf. on Artificial and Neural Networks*, 2001.

[19] M. Banbrook, S. McLaughlin, and I. Mann, "Speech characterization and synthesis by nonlinear methods," *IEEE Trans. Speech Audio Processing*, vol. 7, no. 1, pp. 1–17, Jan. 1999.

[20] I. Mann and S. McLaughlin, "Synthesising natural-sounding vowels using a nonlinear dynamical model," *Signal Process.*, vol. 81, pp. 1743–1756, 2001.

[21] S. McLaughlin, D. Leith, and I. Mann, "Using Gaussian processes to synthesize voiced speech with natural pitch variations," in *Proc. Int. Conf. on DSP*, 2002.

[22] M. Casdagli and S. Eubank, Eds., *Nonlinear Modeling and Forecasting*, 1992.

[23] J.-P. Eckmann, K. O. Kamphorst, D. Ruelle, and S. Ciliberto, "Lyapunov exponents from time series," *Phys. Rev. A*, vol. 34, no. 6, pp. 4971–4979, Dec. 1986.

[24] A. Darbyshire and D. Broomhead, "Robust estimation of tangent maps and Lyapunov spectra," *Phys. D*, vol. 89, pp. 287–305, 1996.

[25] M. Banbrook, G. Ushaw, and S. McLaughlin, "How to extract Lyapunov exponents from short and noisy time series," *IEEE Trans. Signal Processing*, vol. 45, no. 5, pp. 1378–1382, 1997.

[26] G. Gouesbet and C. Letellier, "Global vector field reconstruction by using a multivariate polynomial l_2 approximation on nets," *Phys. Rev. E*, vol. 49, no. 6, pp. 4955–4972, 1994.

[27] G. Golub and C. V. Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins Univ. Press, 1989.

[28] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, 1985.

[29] J. Moody and C. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Comput.*, no. 1, pp. 281–294, 1989.

[30] M. Cowper, B. Mulgrew, and C. Unsworth, "Nonlinear prediction of chaotic signals using a normalized radial basis function network," *Signal Process.*, vol. 82, pp. 775–789, 2002.

[31] S. Ishii and M. Sato, "Reconstruction of chaotic dynamics based on on-line EM algorithm," *Neural Networks*, vol. 14, no. 9, pp. 1239–1256, 2001.

[32] L. Stokbro and D. K. Umberger, "Forecasting with weighted maps," in *Nonlinear Modeling and Forecasting*. Reading, MA: Addison-Wesley, 1990.

[33] M. Jordan and R. Jacobs, "Hierarchical mixtures of experts and the EM algorithm," *Neural Comput.*, vol. 6, no. 2, pp. 181–214, 1994.

[34] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc. B*, vol. 39, pp. 1–22, 1977.
[35] J. Jang, "ANFIS: Adaptive network-based fuzzy inference system," *IEEE Trans. Syst., Man, Cyben.*, vol. 23, pp. 665–685, 1993.
[36] V. Vapnik, S. Golowich, and A. Smola, "Support vector method for function approximation, regression estimation, and signal processing," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 1996, vol. 8, pp. 281–287.
[37] S. Munkherjee, E. Osuna, and F. Girosi, "Nonlinear prediction of chaotic time series using Support Vector Machines," in *Proc. IEEE Workshop Neural Networks for Signal Processing*, Sep. 1997, pp. 511–520.
[38] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: the RPROP algorithm," in *Proc. IEEE Int. Conf. on Neural Networks*, San Francisco, CA, 1993, pp. 586–591.
[39] M. Mannle, "FTSM—fast Takagi Sugeno fuzzy modeling," in *Proc. Safeprocess, IFAC*, Budapest, Hungary, 2000, pp. 587–591.
[40] M. Sato and S. Ishii, "On-line EM algorithm for the normalized Gaussian network," *Neural Comput.*, vol. 12, no. 2, pp. 407–432, 2000.
[41] A. Petry and D. Barone, "Preliminary experiments in speaker verification using time-dependent largest Lyapunov exponents," *Comput. Speech Lang.*, vol. 17, pp. 403–413, 2003.
[42] Y. Freund and R. Schapire, "Experiments with a new boosting algorithm," in *Machine Learning: Proc. 13th Int. Conf.*, 1996, pp. 148–156.
[43] F. Martinez, A. Guillamon, J. Alcaraz, and M. Alcaraz, "Detection of chaotic behavior in speech signals using the largest Lyapunov exponent," in *Proc. Int. Conf. Digital Signal Processing*, 2002, pp. 317–320.

Iasonas Kokkinos (S'04) was born in Athens, Greece, in 1980. He received the Diploma degree in electrical and computer engineering from the National Technical University of Athens (NTUA) in 2001, where he is currently pursuing the Ph.D. degree in computer vision.

During 2003, he visited with the Odyssee Group, INRIA, Sophia Antipolis, France, where he worked on biologically motivated models of image segmentation. His current research interests include object recognition, image segmentation, statistical techniques for pattern recognition, and nonlinear models for signal analysis.

Petros Maragos (S'81–M'85–SM'91–F'95) received the Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1980, and the M.Sc.E.E. and Ph.D. degrees from the Georgia Institute of Technology (Georgia Tech), Atlanta, in 1982 and 1985, respectively.

In 1985, he joined the faculty of the Division of Applied Sciences, Harvard University, Cambridge, MA, where he worked for eight years as Professor of electrical engineering, affiliated with the interdisciplinary Harvard Robotics Lab. He has also been a Consultant to several industry research groups including Xerox's research on image analysis. In 1993, he joined the faculty of the School of Electrical and Computer Engineering, Georgia Tech. Throughout 1996 to 1998, he was on academic leave as a Senior Researcher with the Institute for Language and Speech Processing, Athens. In 1998, he joined the faculty of the National Technical University of Athens where he is currently a Professor of electrical and computer engineering. His current research and teaching interests include the general areas of signal processing, systems, communications, pattern recognition, and their applications to image processing and computer vision, and computer speech processing and recognition.

Dr. Maragos received the 1987 U.S. National Science Foundation Presidential Young Investigator Award; the 1988 IEEE Signal Processing Society's Paper Award for the paper "Morphological Filters;" the 1994 IEEE Signal Processing Society's Senior Award, the 1995 IEEE Baker Award for the paper "Energy Separation in Signal Modulations with Application to Speech Analysis;" and the 1996 Pattern Recognition Society's Honorable Mention Award for the paper "Min-Max Classifiers." He was an Associate Editor for the IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, Guest Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, Editorial Board Member for the *Journal of Visual Communication and Image Representation*, currently Editorial Board member of *Signal Processing*, General Chairman of the 1992 SPIE Conference on Visual Communications and Image Processing, co-Chairman of the 1996 International Symposium on Mathematical Morphology, co-Chairman of the 2001 International Workshop on VLBR Video Coding, and member of two IEEE DSP committees.