Tree-Structured Nonlinear Signal Modeling and Prediction

Olivier J. J. Michel, *Member, IEEE*, Alfred O. Hero, III, *Fellow, IEEE*, and Anne Emmanuelle Badel, *Member, IEEE*

***Abstract*—In this paper, we develop a regression tree approach to identification and prediction of signals that evolve according to an unknown nonlinear state space model. In this approach, a tree is recursively constructed that partitions the p-dimensional state space into a collection of piecewise homogeneous regions utilizing a 2^p-ary splitting rule with an entropy-based node impurity criterion. On this partition, the joint density of the state is approximately piecewise constant, leading to a nonlinear predictor that nearly attains minimum mean square error. This process decomposition is closely related to a generalized version of the thresholded AR signal model (ART), which we call piecewise constant AR (PCAR). We illustrate the method for two cases where classical linear prediction is ineffective: a chaotic "double-scroll" signal measured at the output of a Chua-type electronic circuit and a second-order ART model. We show that the prediction errors are comparable with the nearest neighbor approach to nonlinear prediction but with greatly reduced complexity.**

***Index Terms*—Chaotic signal analysis, nonlinear and nonparametric modeling and prediction, piecewise constant AR models, recursive partitioning, regression trees.**

I. Introduction

NONLINEAR signal prediction is an interesting and challenging problem, especially in applications where the signal exhibits unstable or chaotic behavior [30], [35], [61]. A variety of approaches to modeling nonlinear dynamical systems and predicting nonlinear signals from a sequence of N time samples have been proposed [12], [64] including hidden Markov models (HMM's) [24], [25], nearest neighbor prediction [20], [21], spline interpolation [39], [62], radial basis functions [10], and neural networks [32], [33]. This paper presents a stable low-complexity tree-structured approach to nonlinear modeling and prediction of signals arising from nonlinear dynamical systems.

Tree-based regression models were first introduced as a nonparametric exploratory data analysis technique for non-additive statistical models by Sondquist and Morgan [58]. The regression-tree model represents the data in a hierarchical structure where the leaves of the tree induce a nonuniform partition of the data space over which a piecewise homogeneous statistical model can be defined. Each leaf can be labeled by a scalar or vector-valued nonlinear response variable. Once a cost-complexity metric known as a deviance criterion in the book by Breiman *et al*. on classification and regression trees (CART) [9] is specified, the tree can be recursively grown to perform particular tasks such as nonlinear regression, nonlinear prediction, and clustering [9], [13], [55], [66]. The tree-based approach has several attractive features in the context of nonlinear signal prediction. Unlike maximum likelihood approaches, no parametric model is required; however, if one is available, it can easily be incorporated into the tree structure as a constraint. Unlike approaches based on moments, since the tree-based model is based entirely on joint histograms, all computed statistics are bounded and stable, even in the case of heavy-tailed densities. Unlike most methods, e.g., nearest neighbor, maximum likelihood, kernel density estimation, and spline interpolation, since the tree is constructed from rank order statistics, its performance is invariant to monotonic nonlinear transformations of the predictor variables. Furthermore, as different branches of the tree are grown independently, the tree can easily be updated as new data becomes available.

Our tree-based prediction algorithm has been implemented in Matlab[1] using a k-d tree growing procedure that is similar, but not identical, to that of the S-plus function `tree( )`, as described by Clarke and Pregibon [13]. Important features and contributions of this work are the following:

1) The Takens [19] time delay embedding method is used to construct a discrete-time phase trajectory, i.e., a temporally evolving vector state, for the signal. This trajectory is then input to the tree-growing procedure that attempts to partition the phase space into piecewise homogeneous regions.
2) The partitioning is accomplished by adding or deleting branches (nodes) of the tree according to a maximum entropy homogenization principle. We test that the joint probability density function (j.p.d.f.) is approximately uniform within any node (parent cell) by comparing the conditional entropy of the data points in the candidate partition of the node (children cells) to the maximum achievable conditional entropy in that partition. Cross-entropy criteria for node splitting have been used in the past, e.g., the Kullback–Liebler (KL) "node impurity" measure goes back to Breiman *et al*. [9] and has been proposed as a splitting criterion for tree-structured vector quantization (TSVQ) in Perlmutter *et al*. [47]. More recently, Zhang [66] proposed an entropy criterion for

Manuscript received August 21, 1997; revised April 2, 1999. The associate editor coordinating the review of this paper and approving it for publication was Prof. Peter C. Doerschuk.

O. J. J. Michel and A. E. Badel are with the Laboratoire de Physique, École Normale Supérieure de Lyon, Lyon, France.

A. O. Hero, III is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: hero@eecs.umich.edu).

Publisher Item Identifier S 1053-587X(99)08311-7.

[1]The Matlab code is available by request.

multivariate binomial classification trees that is closer in spirit to the method in this paper. Zhang found that the use of the entropy criterion produces regression trees that are more structurally stable, i.e., they exhibit less variability as a function of N, than those produced using the standard squared prediction error criterion. For the nonlinear prediction application, which is the subject of this paper, we have observed similar advantages using the Pearson Chi-square test of region homogeneity in place of the maximum entropy criterion of Zhang.

3) Similarly to Clarke and Pregibon [13], a median-based splitting rule is used to create splits of a parent cell along orthogonal hyperplanes, which is referred to as a median perpendicular splitting tree in the book by Devroye *et al*. [17]. However, unlike previous methods that split only along the coordinate exhibiting the most spread, here, the median splitting rule is applied simultaneously to each of the p coordinates of the phase space vector producing 2^p subcells. This has the advantage of producing denser partitions per node of the tree, and, as the 2^p-ary split is balanced only in the case of uniform data, the cell probabilities in the split can be used directly for homogeneity testing of the parent cell.
4) In order to reduce the complexity of the tree, a local singular value decomposition (SVD) orthogonalization of the phase space data is performed prior to splitting each node. This procedure, which can be viewed as applying a sequence of local coordinate transformations, produces a partition of the phase space into polygons whose edges are defined by nonorthogonal hyperplanes. This is similar to the principal component splitting method that has been proposed for vector quantization of color images by Orchard and Bouman [46] and other nonorthogonal splitting rules for binary classification trees [17]. However, for phase space dimension $p>2$, our method utilizes all components of the SVD as contrasted with the principal component alone.
5) When applied to nonlinear signal prediction in phase space, the SVD-based splitting rule yields a hierarchical signal model, which we call piecewise constant AR (PCAR), which is a generalization of the nonlinear autoregressive threshold (ART) model called SETAR by Tong [61]. This thresholded AR model has been proposed for many physical signals exhibiting stochastic resonance or bistable/multistable trajectories such as ECG cardiac signals, EEG brain signals, turbulent flow, economic time series, and the output of chaotic dynamical systems (see [31] and [61] for examples). A set of coefficients of the ART model can be extracted from a matrix obtained as the product of the local SVD coordinate transformation matrices. A causal and stable model can then be obtained by Cholesky factorization of this matrix.
6) We give a simple upper bound on the difference between the mean squared error of a fixed regression tree predictor and the minimum attainable mean squared prediction error. The bound establishes that a fixed regression tree predictor attains the optimal MSE when the j.p.d.f. is piecewise constant over the generated partition. This bound can be interpreted as an asymptotic bound on the actual MSE of our regression tree under the assumption that as the training set increases, the generated partition converges a.s. to a nonrandom limiting partition. Many authors have obtained conditions for asymptotic convergence of tree-based classifiers and vector quantizers [17], [45], [43], [44]. However, as this theory requires strong conditions on the input data, e.g., independence, strong mixing, or stationarity, we do not pursue issues of asymptotic convergence in this paper.
7) It is shown by both simulations and experiments with real data that the nonlinear prediction error performance of our regression tree is comparable with that of the popular but more computationally intensive nonparametric nearest neighbor prediction method introduced by Farmer [20], [21]. A similar performance/computation advantage of our regression tree method has been established by Badel *et al*. [10] relative to predictors based on radial basis functions (RBF's).

The outline of the paper is as follows. In Section II, some background on nonlinear dynamical models and their phase space representation is given. Section III continues with background on regression tree prediction, and the basic tree growing algorithm is described. In Section IV, the local SVD orthogonalization method is described, and the equivalence of our SVD-based predictor to ART is established. Finally, in Section V, experiments and simulations are presented.

II. Problem Statement

It will be implicitly assumed that all random processes are ergodic so that ensemble averages associated with the signal can be consistently estimated from time averages over a single realization.

A. Nonlinear Modeling Context

A very general class of nonlinear signal models can be obtained by making nonlinear modifications to the celebrated linear ARMA(p,q) model

$$x(n)=\sum_{i=1}^{p} a_i x(n-i)+\sum_{j=0}^{q} b_j e(n-j) \tag{1}$$

where $e(n)$ is a white Gaussian driving noise with variance σ^2, and the coefficients $\{a_i, i=1,\cdots,p\}$ and $\{b_j, j=1,\cdots,q\}$ are constants independent of $x(n)$ or e_n. Let

$$\boldsymbol{X}_n^{(k)}=[x(n-1) \quad x(n-2) \quad \cdots \quad x(n-k)]^T$$

and

$$\boldsymbol{E}_n^{(k')}=[e(n-1) \quad e(n-2) \quad \cdots \quad e(n-k')]^T$$

be vectors constructed from k and k' past values of x and e, respectively. Nonlinear ARMA(p,q) models can be obtained by letting the coefficients (1) be functions of the ARMA state variables

$$\{a_i, i=1,\cdots,p\}=\mathcal{A}(\boldsymbol{X}_n^{(k)},\boldsymbol{E}_n^{(k')})$$
$$\{b_j, j=0,\cdots,q\}=\mathcal{B}(\boldsymbol{X}_n^{(k)},\boldsymbol{E}_n^{(k')})$$

where $\mathcal{A}$ and $\mathcal{B}$ are functions of $\mathbb{R}^{k+k'}$ into $\mathbb{R}^p$ and $\mathbb{R}^q$, respectively.

This formulation has been used by Tong [61] and others to generate a wide class of nonlinear stochastic models. For example, we easily obtain second-order Volterra models or bilinear models by choosing $\boldsymbol{B} = [1, 0, \cdots, 0]^T$, $\mathcal{B}(\boldsymbol{X}_n^{(k)}, \boldsymbol{E}_n^{(k')}) = \boldsymbol{B}^T \cdot \boldsymbol{E}_n^{(k')}$, and $\mathcal{A}$ as the endomorphism

$$\mathcal{A}(\boldsymbol{X}_n^{(k)}, \boldsymbol{E}_n^{(k')}) = A \cdot [\boldsymbol{X}_n^{(k)^T}, \boldsymbol{E}_n^{(k')^T}]^T$$

where A is a constant matrix with p rows and $(k+k')$ columns.

Similarly, by exchanging the definitions for $\mathcal{A}$ and $\mathcal{B}$ in the preceding equations, we obtain models for which the variance of the driving noise is a function of past values of x. These latter models are called heteroscedastic models and are common in econometrics and other fields (see [18] and [31]).

Moreover, we are not restricted to linear operators for $\mathcal{A}$ and $\mathcal{B}$. Piecewise constant state-dependent values for the matrix $\mathcal{A}$ ($\mathcal{B}$ being kept constant) lead to a class of nonlinear model that is known as "piecewise ARMA" and referred to as a generalized threshold autoregressive (TAR) or a TARMA model [60]. As TARMA model coefficients depend on the previous states $\boldsymbol{X}_n^{(k)}$, they belong to the general class of state-dependant models developed by Priestley [51]. TARMA models arise in areas of time series analysis including biology, oceanography, and hydrology. For more detailed discussion of nonlinear models and their range of application, see [30], [52], and [61].

In what follows, the observed data will be represented by the sampled dynamical equation

$$\begin{aligned} \boldsymbol{S}(n+1) &= F(\boldsymbol{S}(n)) + \varepsilon(n) \\ x(n+1) &= G(\boldsymbol{S}(n+1)) + \eta(n) \end{aligned} \tag{2}$$

where $\boldsymbol{S}(n)$ stands for the state vector at time n, and F and G are (in general) unknown continuous functions from $\mathbb{R}^p$ into $\mathbb{R}^p$ and $\mathbb{R}^q$, respectively. $\varepsilon(n)$ is an i.i.d. state noise, and $\eta(n)$ is an i.i.d. observation noise. For $q > 1$, the observed quantity $x(n)$ is a multichannel measurement. We focus on $q = 1$ here. Note that the well-known linear scalar AR process of order p may be represented within this framework by identifying $F(\boldsymbol{S}(n)) = A\boldsymbol{S}(n)$, where A is a $p \times p$ matrix in companion form, $G(\boldsymbol{S}(n+1)) = \boldsymbol{E}_1^T \boldsymbol{S}(n+1)$, $\boldsymbol{E}_1 = [1, 0, \cdots, 0]^T$, $\eta(n) = 0$, and $\boldsymbol{S}(n) = [x(n), \cdots, x(n-p+1)]^T$.

B. State-Space Reconstruction Method

Any process $x(n)$ obeying the pair of dynamical equations (2) is specified by its state vector $\boldsymbol{S}(n)$, which is known as the *state trajectory*, evolving over $\mathbb{R}^p$, which is known as the *state space*. The process of reconstruction of the state trajectory from real measurements is called state-space embedding. For continuous time measurements $x(t)$, the reconstructed state trajectory is

$$\boldsymbol{X}(n) = [x(n) \quad x(n-\tau_1) \quad \cdots \quad x(n-\tau_{\hat{p}-1})]^T \tag{3}$$

where by $x(n)$ we mean $x(nT_s)$, where

$T_s > 0$ sampling period;

$\hat{p}$ positive integer known as the (estimated) embedding dimension;

τ_i positive real numbers known as the embedding delays.

State-space reconstruction was first proposed by Whitney [65], who stated conditions for identifiability of the continuous time state trajectory in the absence of observation noise. These conditions were formally proved and extended by Takens for the case of nonlinear dynamical systems exhibiting chaotic behavior [11], [19], [59]. In practice, only a finite number of (generally) equispaced samples are available, and the embedding delay is set to $\tau_k = k\tau$, $\tau = mT_s$, where m is an integer value. In this finite case, the value used for τ is very important. Insufficiently large values lead to strong correlation or apparent linear dependences between the coordinates. On the other hand, overly large delays τ excessively decorrelate the components of $\boldsymbol{X}(n)$ so that the dynamical structure is lost [2, ch. 3], [22], [23], [35, ch. 9], [40].

Numerous authors have addressed the problem of finding the best embedding parameters τ and $\hat{p}$ (see, e.g., [22], [23], and [40] for detailed discussion). Selection of the dimension $\hat{p}$ requires investigation of the effective dimension of the space spanned by the estimated residuals. Overestimation of p creates state reconstructions with excessive variance, whereas underestimation creates overly smooth (biased) reconstructions. A widely used method (see [2] or [35] for a discussion on this topic) we will use for estimating p is the following: If $\hat{p} > p$ is the true state dimension, then the estimated trajectories will lie on a lower dimensional manifold in $\mathbb{R}^{\hat{p}}$. This occurrence can be detected by testing a trajectory-dependent dimensionality criterion, e.g., the behavior of the algebraic dimension of the state trajectory vectors, as $\hat{p}$ is increased [38]. We will adopt here the method of Fraser [22] for selection of τ: τ equals the time at which the first zero of the autocorrelation function occurs, i.e., $\tau = \min\{\delta > 0 \colon \hat{C}(\delta) = 0\}$, where

$$\hat{C}(\delta) = \frac{1}{N-1} \sum_{k=1}^{N} x(k)x(k+\delta).$$

III. Growing the Tree

In this section, we discuss the construction of the tree-structured predictor and give a bound on the mean squared prediction error of any fixed tree for the case that $\hat{p} = p$. As above, let $\boldsymbol{X}(n) = [x(n-1), \cdots, x(n-p)]^T$ be a state vector of dimension p. A pth-order tree-structured predictor implements a regression function $\hat{x}(n) = g(\boldsymbol{X}(n))$, which is piecewise constant as $\boldsymbol{X}(n)$ ranges over cells π_k in a partition $\{\pi_k\}$ of $\mathbb{R}^p$ [13]. The most common tree-growing procedure [9], [13], [66] for regression and classification tries to find the partition of the phase space such that the predictive density $f(x(n)|x(n-1), \cdots, x(n-p))$ is approximately constant as the predictor variables $x(n-1), \cdots, x(n-p)$ vary over any of the partition cells. As is shown below, if the tree-growing procedure does this sucessfully, the tree-based predictor $\hat{x}_p(n)$ can attain mean squared error that is virtually identical to that of the optimal predictor $E[x(n)|x(n-1), \cdots, x(n-p)]$.

A. Regression Tree as a Quantized Predictor

Let $I_{\pi_k}(\boldsymbol{X}(n))$ be the indicator of the partition cell π_k and define the vector quantizer function

$$Q(\boldsymbol{X}(n)) = \sum_{k=1}^{L} \boldsymbol{q}_k I_{\pi_k}(\boldsymbol{X}(n))$$

where $\boldsymbol{q}_k = [q_{k1}, \cdots, q_{kp}]^T$ is an arbitrary point in π_k. Typically, $\boldsymbol{q}_k$ is taken as the centroid of region π_k, but this is immaterial in the following. Since the predictor function $\hat{x}(n) = g(\boldsymbol{X}(n))$ is piecewise constant, it is obvious that $g(\boldsymbol{X}(n)) = g(Q(\boldsymbol{X}(n)))$, i.e., the tree-structured predictor can be implemented using only the quantized predictor variables $Q(\boldsymbol{X}(n))$. Therefore, given the partition $\{\pi_k\}$, the optimal tree-based predictor can be constructed from the multidimensional histogram (the partition cell probabilities) as the conditional mean of $x(n)$, given the vector $Q(\boldsymbol{X}(n))$.

B. A Bound on MSE of Tree-Structured Predictor

It follows from Theorem 1 in the Appendix that if the conditional density $f(x(n)|\boldsymbol{X}(n))$ is (Lipschitz) continuous of order α within all partition cells of the partition $\{\pi_k\}$ of $\mathbb{R}^p$, the mean squared error $E[(x(n) - E[x(n)|Q(\boldsymbol{X}(n))])^2]$ of the tree-structured predictor satisfies the bound

$$\begin{aligned} 0 \le & E[(x(n) - E[x(n)|Q(\boldsymbol{X}(n))])^2] \\ & - E[(x(n) - E[x(n)|\boldsymbol{X}])^2] \\ & \le 2 \max_i K_{\pi_i} m_x^2 E[||\boldsymbol{X} - Q^o(\boldsymbol{X})||^\alpha] \end{aligned} \tag{4}$$

where

$Q^o(\boldsymbol{X}(n))$	minimum mean squared error quantizer on $\{\pi_k\}$;
m_x	upper bound on the mean squared valued of $x(n)$ given $\boldsymbol{X}(n)$;
K_{π_k}	Lipschitz constant characterizing the modulus of continuity within π_k.

The upper bound in (4) is decreasing in the minimum αth power quantization error $E[||\boldsymbol{X}(n) - Q^o(\boldsymbol{X}(n))||^\alpha]$ associated with optimal vector quantization of the predictor variables. Bounds and asymptotic expressions exist for this quantity [29], [42], which can be used to render the upper bound (21) more explicit; however, this will not be explored here.

Note that the upper bound in (4) is decreasing in $\max_i K_{\pi_i}$ and equals zero when $f(x(n)|\boldsymbol{X}(n))$ is piecewise constant in $\boldsymbol{X}(n)$, i.e., $f(x(n)|\boldsymbol{X}(n) = \boldsymbol{x}) = \Sigma_i f(x(n)|\boldsymbol{q}_i) I_{\pi_i}(\boldsymbol{x})$, where $\boldsymbol{q}_i \in \pi_i$ are arbitrary. Thus, in the case of a piecewise uniform conditional density, the optimal predictor of $x(n)$ given quantized data $Q(\boldsymbol{X}(n))$ is identical to the optimal nonlinear predictor given unquantized data $\boldsymbol{X}(n)$, i.e., the tree-structured predictor attains the minimum possible prediction MSE. Note that for a general conditional density, both $E[||\boldsymbol{X} - Q^o(\boldsymbol{X})||^\alpha]$ and the total variations $\{K_{\pi_k}\}$ decrease as the sizes of the partition cells $\{\pi_k\}_i$ decrease. Hence, the mean square prediction error can be seen from (4) to improve monotonically as the conditional density $f(x(n)|\boldsymbol{X}(n))$ becomes well approximated by the staircase function $f(x(n)|Q(\boldsymbol{X}(n))$ over $\boldsymbol{X}(n) \in \mathbb{R}^p$. This forms the basis for tree-based nonlinear prediction, as explained in more detail below.

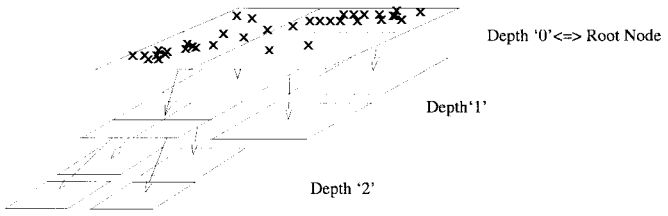


Fig. 1. Graphical depiction of the tree growing algorithm using separable 2^p-ary splitting rule. For a $\hat{p} = 2$ dimensional state space embedding, the tree is a quadtree. The root-node is split into four subcells, and the sample distribution of points is found to be nonuniform. Among the derived subsets, only the one depicted by the lower left corner square is found to be nonuniform and is split further.

C. Branch Splitting and Stopping Rules

Here, we describe the generic recursive procedure used for growing the tree from training data. Let $\hat{p}$ be an estimate of the phase space dimension p of the signal. Assume that at iteration l of the tree growing procedure, we have created a partition Π^l and consider the partition cells π_i^l, which we call the ith parent nodes at depth l. We refine the partition Π^l by recursively splitting each partition cell π_i^l into $2^{\hat{p}}$ smaller cells, which are called children nodes of the ith parent.

To control the number of nodes of the tree, we test the residuals in each partition element of Π^l against a uniform distribution. If the test for uniformity fails in a particular cell, that cell is split, and $2^{\hat{p}}$ parent nodes at depth $l+1$ are created. Otherwise, the cell is not split and is declared a terminal node. The set of terminal nodes are called the leaves of the tree. See Fig. 1 for a graphical illustration of the generic tree-growing procedure. The final tree specifies a set of leaves $\pi_1, \cdots, \pi_L$ partitioning the state-space together with the empirical histogram (cell occupancy rate) $\hat{p}_j = P(\boldsymbol{X} \in \pi_j) = N_{\pi_j}/N$, where N_{π_j} is the number of samples $\{\boldsymbol{X}(k)\}_{k=1}^N$ that fall into leaf π_j.

1) Cell Uniformity Test: Here, we discuss the selection of the goodness-of-split criterion that is used to test uniformity. As above, let $\boldsymbol{X}(k)$ denote the $\hat{p}$-dimensional vector sampled at time kT_s, where the reconstruction dimension $\hat{p}$ is fixed. Many discriminants are available for testing uniformity, including Kolmogorov–Smirnov tests [37], rank-order statistical tests [16], and scatter matrix tests [26]. Following Breiman *et al.* and Zhang [9], [66], we adopt an entropy-like criterion. However, as contrasted to previous implementations [9], [66], this criterion is implemented using simple Chi-square goodness-of-fit test of significance over the distribution of child cell probabilities.

For a partition $\{\pi_1, \cdots, \pi_{2^{\hat{p}}}\}$ of a cell Π, let N_{π_i} be the number of vectors $\boldsymbol{X}(k)$, $k = 1, \cdots, N$, found in π_i. We assume that the vectors falling into the cells are approximately i.i.d. and that N_{π_i}, $i = 1, \cdots, 2^{\hat{p}}$ are approximately multinomial distributed random variables with class probabilities $p_i = P(\boldsymbol{X}(p) \in \pi_i | \boldsymbol{X}(p) \in \Pi) = E[N_{\pi_i}]/N$. These are reasonable assumptions when the volume of cell Π is small and $x(n)$ satisfies a long range decorrelation property (weak mixing), but we do not pursue a proof of this here. The test of uniformity is implemented by using the empirical cell probabilities $\hat{p}_i = (N_{\pi_i}/N)$ to test the uniform hypothesis H_0: $p_i = 2^{-\hat{p}}$, $i = 1,$

$\cdots$, $2^{\hat{p}}$ against the composite alternative hypothesis H_1: $p_i \neq 2^{-\hat{p}}$, $i = 1, \cdots, 2^{\hat{p}}$. Define the Kullback–Liebler (KL) distance between $\hat{p}_i$ and the uniform distribution $p_i = 2^{-\hat{p}}$

$$D(\hat{p}_i, p_i) = \sum_{i=1}^{2^{\hat{p}}} p_i \log\left(\frac{\hat{p}_i}{p_i}\right) = \log 2^{\hat{p}} + \sum_{i=1}^{2^{\hat{p}}} p_i \log \hat{p}_i. \quad (5)$$

It is easy to show that the generalized likelihood ratio test of H_0 versus H_1 decides H_0 if $D(\hat{p}_i, p_i) < \eta$, where the threshold η selected to ensure that the probability of false rejection of H_0 is equal to a prescribed false alarm rate (see, e.g., [7, ch. 8]).

However, since the distribution of $D(\hat{p}_i, p_i)$ is intractable under H_0, the decision threshold cannot easily be chosen to satisfy a prespecified false alarm level. We instead propose Pearson's Chi square goodness-of-fit test statistic χ^2, which has a central Chi square distribution under H_0. In particular, it can be shown [6], [7] that Pearson's Chi square statistic is a local approximation to the KL distance statistic (5) in the sense that

$$D(\hat{p}_i, p_i) = \frac{1}{2N_\Pi}\chi^2 + o(\max_i(\hat{p}_i - p_i)^2)$$

where $\chi^2 = N_\Pi \sum_{i=1}^{2^{\hat{p}}} ((\hat{p}_i - p_i)^2/p_i)$ is distributed as a central Chi square with $2^{\hat{p}} - 1$ degrees of freedom undr H_0.

2) Separable Splitting Rule: Another component of the procedure for growing a tree is the method of splitting parent cells into children cells. The standard cell splitting rule attempts to create a pair of rectangular subcells for which all marginal probabilities are identical regardless of the underlying distribution. The median-based binary splitting method for constructing k-d trees [5], [15], [13], [17] is commonly used for this purpose. As the median is a rank-order statistic, this gives the property that the predictor is invariant to monotone transformations of the predictor variables: a property not shared by most other nonlinear predictors. Here, we present a variant of the standard median splitting rule that generates $2^{\hat{p}}$ rectangular children cells that only have equal probabilities when the data is uniform over the parent cell. A version of this $2^{\hat{p}}$-ary splitting rule that generates nonrectangular cells is discussed in Section IV.

Let $\times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$ denote the hyper-rectangle constructed from the Cartesian product of intervals $[\alpha_i, \beta_i]$, $\alpha_i < \beta_i$, e.g., $\times_{i=1}^{2}[\alpha_i, \beta_i] = [\alpha_1, \beta_1] \times [\alpha_2, \beta_2]$ is a right parallelepiped in $\mathbb{R}^2$. We start with a partition element $\Pi = \times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$. Let this partition element contain N_Π of the reconstructed state vectors $\{\boldsymbol{X}(k)\}_{k=1}^{N}$. Define the N_Π-element vector $\boldsymbol{X}_\Pi^j = [\boldsymbol{e}_j^T \boldsymbol{X}(k): \boldsymbol{X}(k) \in \Pi, k = 1, \cdots, N]$ as the projection of the inscribed reconstruction vectors onto the jth coordinate axis. That is, $\boldsymbol{X}_\Pi^j$ is the set of jth coordinates of those $\boldsymbol{X}(k)$ falling into Π, $k = 1, \cdots, N$. Denote by $\hat{T}_\Pi^j$ the sample median of the jth coordinate axis projections

$$\hat{T}_\Pi^j = \text{median}\{\boldsymbol{e}_j^T \boldsymbol{X}(k): ; \boldsymbol{X}(k) \in \Pi, k = 1, \cdots, N\}$$

where, for a scalar sequence $\{x_i\}_{i=1}^n$, the sample median is a threshold such that half fall to the left and half to the right

$$\text{median}\{x_i\} = \begin{cases} x_{(n/2)}, & n \text{ even} \\ x_{([n+1]/2)}, & n \text{ odd} \end{cases}$$

and $x_{(1)} \leq \cdots \leq x_{(n)}$ denotes the rank ordered sequence. Note that when the points $\{\boldsymbol{X}(k)\}_k$ are truly uniform over the parent cell, the medians $\{\hat{T}_\Pi^j\}_j$ will tend to be near the midpoints of the edges of the parent cell.

The standard median tree implements a binary split of parent cell Π about a hyperplane perpendicular to that coordinate axis j having the largest spread of points $\boldsymbol{X}_\Pi^j$, where the hyperplane intersects this coordinate axis at the median $\hat{T}_\Pi^j$. This produces a pair of children cells that contain an identical number of points. In contrast, we split Π into $2^{\hat{p}}$ rectangular children cells whose edges are defined by all $\hat{p}$ perpendicular hyperplanes of the form $\{\boldsymbol{X}: \boldsymbol{e}_j^T \boldsymbol{X} = \hat{T}_\Pi^j\}$, $j = 1, \cdots, \hat{p}$. This produces a tree with a denser partition than the standard median tree having the same number of nodes. Unlike the standard median splitting rule tree, these $2^{\hat{p}}$ children cells will not have identical numbers of points unless the points are truly uniform over Π. This allows the cell occupancies in the $2^{\hat{p}}$-ary split to be used directly for uniformity testing as described in the previous section.

3) Stopping Rule: The last component of the tree-growing procedure is a stopping rule to avoid overfitting. As above, define $\boldsymbol{X}_\Pi^j = \{\boldsymbol{e}_j^T \boldsymbol{X}(k): \boldsymbol{X}(k) \in \Pi, k = 1, \cdots, N\}$ as the jth coordinates of the vectors $\boldsymbol{X}(k)$ falling into the hyper-rectangle $\Pi = \times_{i=1}^{\hat{p}}[\alpha_i, \beta_i]$. Thus, each of the elements of $\boldsymbol{X}_\Pi^j$ lies in the interval $[\alpha_j, \beta_j]$. Under the assumption that these elements are i.i.d. with continuous marginal probability density function $f_{x^j|\Pi}$, the sample median $\hat{T}_\Pi^j$ is an asymptotically unbiased and consistent estimator of the theoretical median T_Π^j, which is the half mass point of the marginal cumulative distribution function. Conditioned on N_Π, the sample median has an asymptotic normal distribution [41]

$$\hat{T}_\Pi^j \sim \mathcal{N}\left(T_\Pi^j, \frac{1}{4N_\Pi[f_{x^j|\Pi}(T_\Pi^j)]^2}\right). \quad (6)$$

The stopping rule is constructed under the assumption that $f_{x^j|\Pi}$ is a uniform density $f_{x^j|\Pi}(x) = 1/(\beta_j - \alpha_j)$ over $x \in [\alpha_i, \beta_i]$. Under this assumption, $T_\Pi^j = (\beta_i + \alpha_i)/2$ is the midpoint, and the sample medians $\hat{T}_\Pi^j$, $j = 1, \cdots, \hat{p}$ are statistically independent. A natural stopping criterion is to require that the number N_Π of data points within Π be sufficiently large so that the Gaussian approximation to the density of $\hat{T}_\Pi^j$ has negligible mass outside of the interval $[\alpha_i, \beta_i]$. When this is the case, it can be expected that $\hat{T}_\Pi$ will be a reliable estimate of the interval midpoint. More concretely, we will require that N_Π satisfies

$$P(|\hat{T}_\Pi^j - T_\Pi^j| \leq (\beta_j - \alpha_j)/2, j = 1, \cdots, \hat{p}) \geq 1 - \epsilon \quad (7)$$

where $\epsilon \in [0, 1]$ is a suitable (small) prespecified constant.

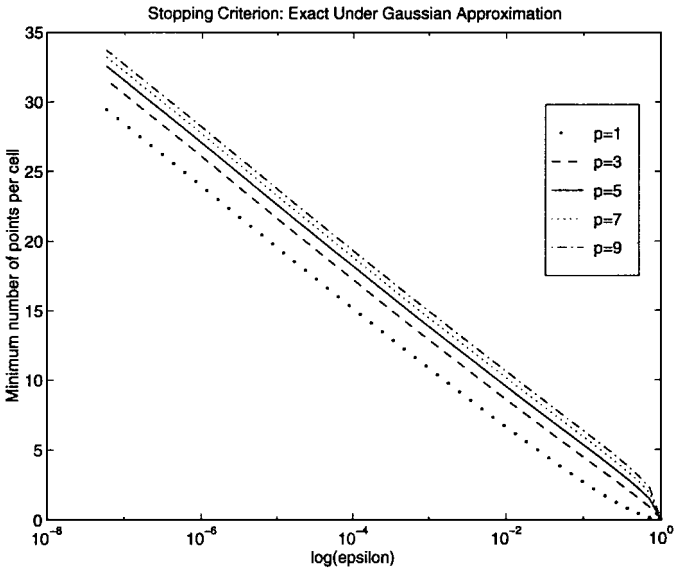


Fig. 2. Family of curves describing cell subdivision stopping rule in terms of minimum number of points N_Π falling into a rectangular cell π and the probability criterion $\epsilon \in [0,1]$. The vertical axis is the minimum number of points that will be assigned to a subdivided cell, and the horizontal axis is the log of ϵ.

Since the $\hat{T}^j_\Pi$ are independent, (7) is equivalent to

$$1 - \prod_{j=1}^{\hat{p}} P(|\hat{T}^j_\Pi - T^j_\Pi| \le (\beta_j - \alpha_j)/2) \le \epsilon$$

which, under the Gaussian approximation (6), gives

$$1 - P(|Z| \le \sqrt{N_\Pi})^{\hat{p}} \le \epsilon$$

where Z is a standard normal random variable (zero mean and unit variance). Thus, we obtain the following stopping criterion: Continue subdividing the cell Π as long as

$$N_\Pi \ge 2[\mathrm{erf}^{-1}([1-\epsilon]^{1/\hat{p}})]^2 \tag{8}$$

where $\mathrm{erf}(x) = (2/\sqrt{\pi}) \int_0^x e^{-t^2}\, dt$. Use of the asymptotic representation $1 - \mathrm{erf}(x) = \mathrm{erfc}(x) = e^{-x^2}/(x\sqrt{\pi}) + o(1/x)$ [3, 26.2.12] gives the log-linear small ϵ version of (8)

$$N_\Pi \ge 2 \ln\left(\frac{2\hat{p}}{\epsilon}\right). \tag{9}$$

The right-hand side of (8) is plotted as a function of ϵ for several values of $\hat{p}$ in Fig. 2. Note that as predicted by the asymptotic (small ϵ) bound (9), the curves are very close to log linear in ϵ. As a concrete example, the criterion $\epsilon = 0.01$ (99% of Gaussian probability mass is inside π) gives for $\hat{p} = 2$: $N_\Pi = 8$, and for $\hat{p} = 4$: $N_\Pi = 10$ as the minimum number of data points N_Π for which a cell will be further subdivided. These numbers are of the same order as those obtained from the volume estimation criterion used by Badel *et al.* [6].

D. Computational Cost

The steps outlined in the preceding subsections may be summarized by the following tree growing algorithm:

1) **input** sampled time series and embedding parameters, $(\hat{p}, \tau)$, Pearson's χ^2 test threshold
2) **initialize** $\Pi^0 =$ set of all state vectors
3) **while** nonempty nonterminal leaves exist, at current depth l
4) **for** each cell Π^l
5) **if** Π^l contains $N_{\Pi^l} \ge \hat{p}^2 4^{\hat{p}-1}$ vectors
6) compute the splitting thresholds $T^j_{\Pi^l}$, $j = 1, \cdots, \hat{p}$
7) estimate empirical probabilities at depth $l+1$ for the children of Π^l
8) Compute Pearson's χ^2 statistic from the $2^{\hat{p}}$ probabilites
9) **if** χ^2 is less than threshold
10) Π^l is stored as a terminal leaf
11) **else**
12) $\{T^j_{\Pi^l}, j = 1, \cdots, \hat{p}\}$ are stored
13) $\{\Pi^{l+1}_k, k = 1, \cdots, 2^{\hat{p}}\}$ are stored
14) **endif**
15) **else** Π^l is stored as an 'empty' terminal leaf
16) **endif**
17) **endfor**
18) $l = l + 1$;
19) **goto** line 3.

The computational cost associated with this tree estimation algorithm is signal dependent. For example, in the case of a state space containing N realizations of a p-dimensional white noise, the trivial partition $\Pi_0 = \mathbb{R}^p$ will generally pass the uniformity test, and the algorithm will stop at the root node. In this case, only a very few computations are needed. In the following, we give an estimate of the worst-case cost occurring when the terminal nodes all occur at the same depth.

At depth l of the tree, under the assumption that all obtained cells were stored as nonempty, nonterminal leaves, the tree has $2^{\hat{p}l}$ cells. The average number of $\hat{p}$-dimensional vectors in each leaf is

$$\langle N_{\Pi^l} \rangle = \frac{N_{\Pi^0}}{2^{\hat{p}l}}.$$

The most computation consuming step in the algorithm is the splitting threshold determination procedure that requires rank ordering each of the coordinates of the inscribed state vectors. Using an optimized method (e.g., the heap-sort algorithm [50]) leads to a cost proportional to

$$C_l \simeq 2^{\hat{p}l} \langle N_{\Pi^l} \rangle \log_2 \langle N_{\Pi^l} \rangle = N_{\Pi^0} \hat{p} \log_2 \frac{N_{\Pi^0}}{2^{\hat{p}l}}.$$

By adding the computational costs obtained for each depth in the range $l = 0, \cdots, l_{\max} - 1$, we obtain the expression of the total cost

$$C_{\mathrm{Tot}} \simeq N_{\Pi^0} \hat{p} l_{\max} \log_2 \frac{N_{\Pi^0}}{2^{\hat{p}(l_{\max}-1)/2}}.$$

Note that the expression of C_{Tot} corresponds to the worst case where no cells pass the χ^2 uniformity test until the minimal

cell residency stopping criterion is reached. This computational cost is well below that of the well-known nearest neighbor one-step prediction method: $C_{NN} \simeq \hat{p}(N_{\Pi_0}^2/2) + N_{\Pi_0} \log N_{\Pi_0}$.

IV. Complexity Reduction via SVD Orthogonalization

The number of leaves in the final tree, i.e., the number of cells in the partition of state space, is a reasonable measure of model complexity. However, without additional preprocessing of the data, the separable splitting rule described in the previous section can produce trees of greatly varying complexity for state space trajectories which are identical up to a rotation in $\mathbb{R}^p$. This is an undesirable feature since a simple transformation of coordinates in the state space, such as translation, scale, and rotation, does not change the intrinsic complexity of the process, e.g., as measured by process entropy or Lyapunov exponent.

As a particularly simple example, consider the case where the state trajectory evolves about line segment in two dimensions $x(n) = ax(n-1) + \epsilon(n)$, where $\epsilon(n)$ is a white noise with variance σ^2. Under the separable 2^p-ary partitioning rule when $a = +1$ or $a = -1$, a very complex tree will result. This is because the Chi-square splitting criteria will lead to a tree with cell sizes on the order of magnitude of σ. This is troublesome, as a simple rotation of the axis coordinates by an angle of $\pi/4$ will lead the partitioning algorithm to stop at the root node. Here, we perform a local recursive orthogonalization of the state vector prior to node splitting in order to produce trees with fewer leaves. This produces a new orthogonalized sequence of node variables $\boldsymbol{Z}_{\pi_j^l}$ that are used in place of $\boldsymbol{X}_{\pi_j^l}$ to perform separable splitting and goodness-of-split tests discussed in the previous section. The local recursive orthogonalization described below differs from a similar principal component orthogonalization for binary partitioning, first described by Orchard and Bouman [46], in that all the components of the SVD are utilized for the $2^{\hat{p}}$-ary partition used in this paper.

A. Local Recursive Orthogonalization

We recursively define a set of orthogonalized node variables as follows. Let $\boldsymbol{X}$ be a $\hat{p} \times N$ matrix of samples $\boldsymbol{X}(k)$, $k = 1, \cdots, N$ of the $\hat{p}$-dimensional state trajectory. Let the covariance of $\boldsymbol{X}(k)$ be denoted $\Lambda_{\boldsymbol{X}}$, and let it have the SVD (eigendecomposition) $\Lambda_{\boldsymbol{X}} = M_{\boldsymbol{X}}^T \operatorname{diag}(\lambda_{\boldsymbol{X}_{(k)}}) M_{\boldsymbol{X}}$. Define the root node $\pi^0 = \mathbb{R}^{\hat{p}}$. Next, define the orthogonalized set of vectors $\boldsymbol{Z}_{\pi^0}$

$$\boldsymbol{Z}_{\pi^0} = M_{\boldsymbol{X}}(\boldsymbol{X} - E[\boldsymbol{X}]).$$

The matrix $\boldsymbol{Z}_{\pi^0}$ is now used in place of $\boldsymbol{X}$ to determine the split of the root node into children $\pi_1^1, \cdots, \pi_{2^{\hat{p}}}^1$ according to the same separable splitting and stopping criteria as before. In practice, the empirical mean $\hat{\boldsymbol{X}} = (1/N)\boldsymbol{X}\boldsymbol{1}$ and empirical covariance $(\boldsymbol{X} - \hat{\boldsymbol{X}})(\boldsymbol{X} - \hat{\boldsymbol{X}})^T/(N-1)$ are used in place of $E[\boldsymbol{X}]$ and $\Lambda_{\boldsymbol{X}}$.

Now, assume a split occurs at the root node and define $\boldsymbol{Z}_{\pi_j^1}^0$ as the matrix of columns of $\boldsymbol{Z}_{\pi^0}$ that lie inside π_j^1. The (empirical) mean and covariance matrix $\Lambda_{\boldsymbol{Z}_{\pi_j^1}^0}$ of $\boldsymbol{Z}_{\pi_j^1}^0$ are computed. Next, the unitary matrix $M_{\boldsymbol{Z}_{\pi_j^1}^0}$ of the eigenvectors of $\Lambda_{\boldsymbol{Z}_{\pi_j^1}^0}$ is extracted via SVD. This unitary matrix is applied to $\boldsymbol{Z}_{\pi_j^1}^0$ to produce an equivalent but uncorrelated set of vectors $\boldsymbol{Z}_{\pi_j^1}$

$$\boldsymbol{Z}_{\pi_j^1} = M_{\boldsymbol{Z}_{\pi_j^1}^0}(\boldsymbol{Z}_{\pi_j^1}^0 - E[\boldsymbol{Z}_{\pi_j^1}^0]) + E[\boldsymbol{Z}_{\pi_j^1}^0]\boldsymbol{1}_{\pi_j^1}^T$$

where $\boldsymbol{1}_{\pi_j^1}^T$ stands for the transpose of the vector containing $N_{\pi_j^1}$ ones. Application of this local orthogonalization procedure over all $2^{\hat{p}}$ hyper-rectangles $\pi_1^1, \cdots, \pi_{2^{\hat{p}}}^1$ produces a set of local coordinate rotations that results in changing the shape of the hyper-rectangles into hyper-parallelepipeds. When this process is repeated, these hyper-parallelepipeds are further subdivided, producing, at termination of the algorithm, a partition of the state space into general polytopes π_j^1.

The general recursion from depth l to depth $l+1$ can be written as

$$\boldsymbol{Z}_{\pi_j^{l+1}} = M_{\boldsymbol{Z}_{\pi_j^{l+1}}^l} \boldsymbol{Z}_{\pi_j^{l+1}}^l + \boldsymbol{C}^l \boldsymbol{1}_{\pi_j^{l+1}}^T \tag{10}$$

where $\boldsymbol{C}^l = E[\boldsymbol{Z}_{\pi_j^{l+1}}^l] - M_{\boldsymbol{Z}_{\pi_j^{l+1}}^l} E[\boldsymbol{Z}_{\pi_j^{l+1}}^l]$.

B. Relation to Piecewise Constant AR (PCAR) Models

Once the tree growing procedure terminates, the partitions π_j^l can be mapped back to the original state space by a sequence of backward recursions that backprojects the π_j^{l+1} node variables $\boldsymbol{Z}_{\pi_j^{l+1}}$ into the parent cell π^l via the relation

$$\boldsymbol{Z}_{\pi_j^{l+1}}^l = M_{\boldsymbol{Z}_{\pi_j^{l+1}}^l}^T (\boldsymbol{Z}_{\pi_j^{l+1}} - \boldsymbol{C}_{\pi^l}). \tag{11}$$

Iteration of (11) over l yields an equation for backprojection of $\boldsymbol{Z}_{\pi_j^{l+1}}$ to the root node. By induction on l, the forward recursion (10) gives the relation

$$\boldsymbol{Z}_{\pi^l} = \mathcal{M}_{\pi^l} \boldsymbol{X}_{\pi^l} - \mathcal{C}_{\pi^l} \boldsymbol{1}_{\pi^l}^T \tag{12}$$

where $\boldsymbol{X}_{\pi^l}$ denotes the subset of columns of $\boldsymbol{X}$ that are mapped to terminal node π^l at depth l via the sequence of bijective maps (10), and $\mathcal{M}_{\pi^l}$ and $\mathcal{C}_{\pi^l}$ are matrices formed from the telescoping series

$$\mathcal{M}_{\pi^l} = \prod_{i=0}^{l} M_{\boldsymbol{Z}_{\pi^l}^i} \tag{13}$$

$$\mathcal{C}_{\pi^l} = \sum_{i=0}^{l} \left[\prod_{j=i}^{l} M_{\boldsymbol{Z}_{\pi^l}^j} \right] \boldsymbol{C}_{\pi^i} \tag{14}$$

where $M_{\boldsymbol{Z}_{\pi^l}^0}$ is defined as the p-dimensional identity matrix.

For any parent node π^l, the covariance matrix of the rotated data $\boldsymbol{Z}_{\pi^l}$ is diagonal, which means that the components of $\boldsymbol{Z}^{\pi^l}$ are separable (in the mean squared sense) but not necessarily uniform. On this rotated data, the Chi-square test for uniformity can easily be implemented on a coordinate-by-coordinate basis. When the tree-growing procedure terminates, we will have found a set of partition cells $\pi_1^{l_1}, \cdots, \pi_L^{l_L}$ such that each $\pi^l = \pi_j^l$ contains points $\boldsymbol{Z}_{\pi^l}$ that are (approximately) uniformly distributed over π^l. Thus, (12) gives

an autoregressive AR$(p-1)$ model whose coefficients are piecewise constant over regions of state space $\boldsymbol{X}_{\pi^l}$.

This can be made more transparent by writing the ith component of relation (12) as

$$x(n) = -\sum_{j=1}^{p-1} a_{\pi^l}(i,j)x(n-j) + w_{\pi^l}(n)$$
$$\boldsymbol{X}(n) \in \pi^l \tag{15}$$

where $a_{\pi^l}(i,j) = m_{\pi^l}(i,j+1)/m_{\pi^l}(i,1)$, $m_{\pi^l}(i,j)$ denotes the i,j element of $\mathcal{M}_{\pi^l}$, and $w_{\pi^l}(n) = (\boldsymbol{Z}_{\pi^l}(n) + \mathcal{C}_{\pi^l}\mathbf{1}_{\pi^l}^T)\boldsymbol{e}_1^i$ is a white noise.

Note that the coefficients for the PCAR representation (15) may not be stable. There is an alternative approach to orthogonalizing the node variables which uses Gramm–Schmidt recursions and guarantees that all PCAR coefficients are stable. This method is equivalent to constructing the Schur complement by adding one coordinate to each vector in the node, amounting to recursively synthesizing a local stable AR$(p-1)$ model over $p = 1, 2, 3, \cdots$. This is tantamount to performing Cholesky (LDU) factorization of the local covariance matrices $\Lambda_{\boldsymbol{Z}^l_{\pi^{l+1}_j}}$ [56], as contrasted with the SVD factorization described above. In the sequel, the former method will lead to what will be called a Schur-tree, whereas the latter will lead to a tree called the SVD-tree.

The PCAR model (12) is a generalization of the AR-threshold (ART) model called SETAR in Tong [61]. Similarly to the PCAR model (12), SETAR is an AR model whose coefficients are piecewise constant over regions of state space; however, unlike the PCAR model, these regions are restricted to half planes. In particular, a two-level single coordinate pth-order SETAR model is

$$x(n) = \begin{cases} a_{10} + a_{11}x(n-1) + \cdots + a_{1d}x(n-p) + \sigma_1\epsilon(n) \\ \quad \text{if } x(n-d) \le T_0 \\ a_{20} + a_{21}x(n-1) + \cdots + a_{2d}x(n-p) + \sigma_2\epsilon(n) \\ \quad \text{if } x(n-d) > T_0 \end{cases} \tag{16}$$

where $d \in \{1, \cdots, p\}$. As far as we know, filtering, prediction, and identification of SETAR models have only been studied for the case where the switching of the AR coefficients depends on a single coordinate $x(n-d)$ and where the switching threshold T_0 is known. The PCAR generalization of SETAR models allows transition thresholds to be applied to linear combinations of past values. As will be illustrated below, the orthogonalized version of the tree based partitioning algorithm is well adapted to filtering, prediction and identification over these models.

V. Examples and Applications

In this section, the tree-structured predictors are applied to various real and simulated data examples.

A. Illustrative Examples

To illustrate the parsimony of the local recursive orthogonalization method, we first consider a rather artificial random process that follows a piecewise linear trajectory through state space (see Fig. 3). A trajectory made of three linear segments in a two-dimensional (2-D) state space was simulated. The segments have slopes 1.25, -0.25, and -2.5, respectively.

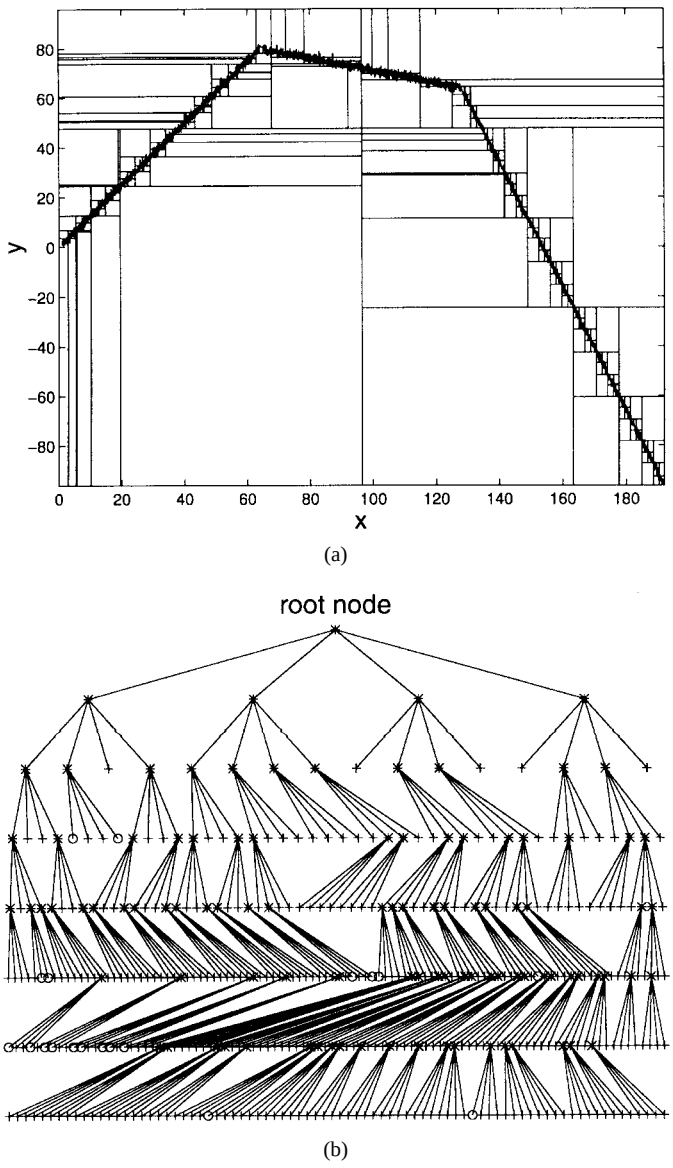


Fig. 3. Tree-structured predictor for separable splitting rule applied to a piecewise linear phase space trajectory in two dimensions. (a) Simulated state space trajectory in two dimensions with superimposed rectangular partition produced by recursive tree (RT) growing algorithm. (b) Representation of the quadtree associated with the state-space trajectory depicted in (a).

Each segment contains 128 realization of the 2-D state vector. White Gaussian i.i.d. noise of variance $\sigma^2 = 5$ was added to the trajectory. We first applied the recursive tree (RT) method in $p = 2$ state dimensions without SVD orthogonalization. Both the rectangular partition of the state space [Fig. 3(a)] and the tree partitioning algorithm [Fig. 3(b)] exhibit high complexity. The number of terminal leaves of the resulting quadtree is driven exclusively by the variance of the additive noise. We next grew a quadtree using the local recursive SVD orthogonalization procedure, which will be called SVD-tree here, described in Section IV. The orthogonalization procedure re-expresses the state vectors in their local eigenbases at each splitting iteration and, as seen from

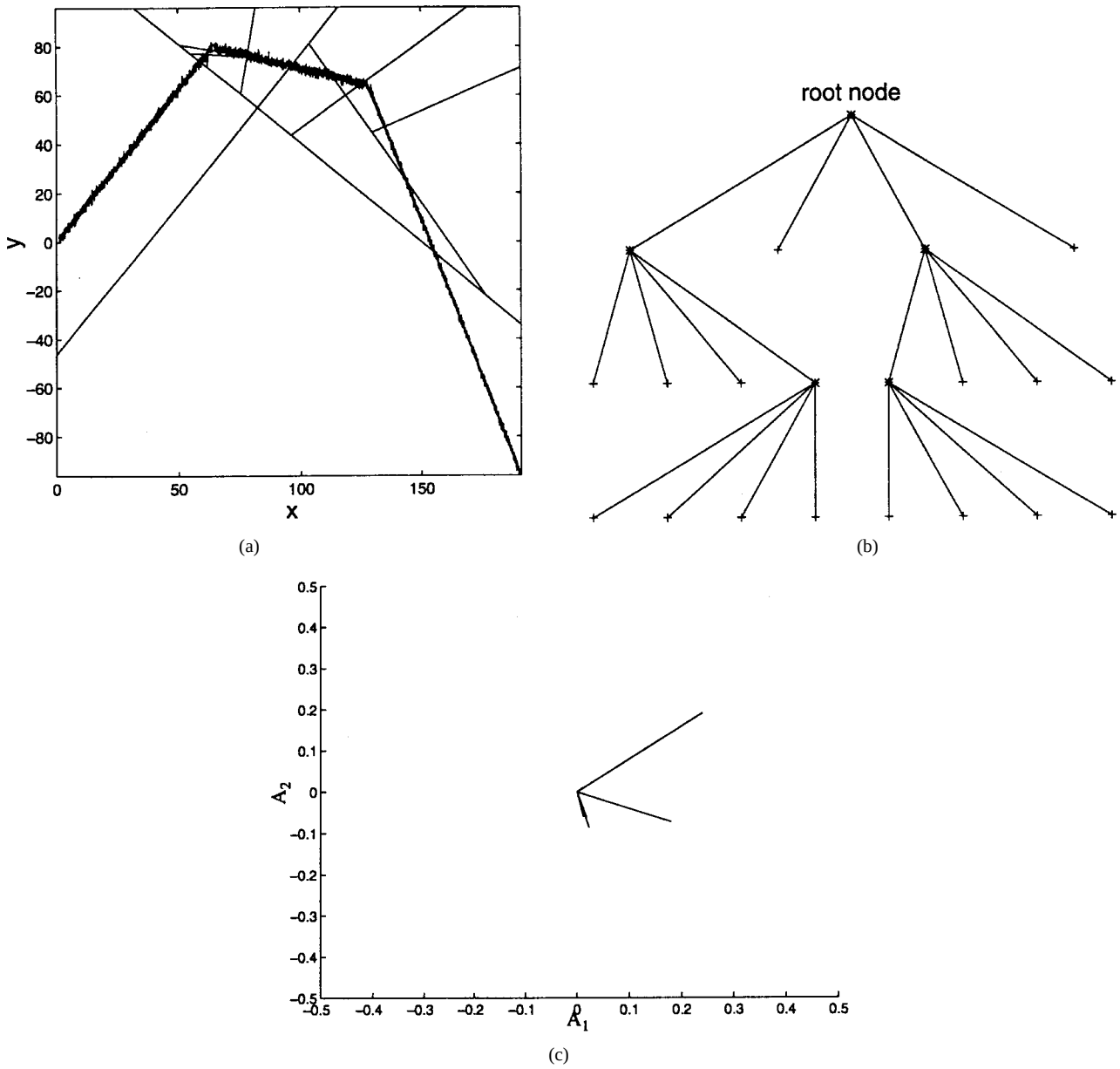


Fig. 4. (a) Same simulated state-space trajectory as in Fig. 3 but with recursive SVD-tree partitioning. (b) Representation of the SVD-tree associated with the state-space trajectory depicted in (a). (c) The pairs of estimated (normalized) AR coefficients governing the dynamics in each cell are plotted with lengths proportional to the occupancy rate (number of points) of the cell.

Fig. 4, produces a tree partitioning of lower complexity with many fewer leaves. As explained in Section IV, applying the recursive SVD orthogonalization on a cell π_j^l synthesizes the local AR(1) model [recall (15)]

$$x(n) = -a_{\pi_j^l} x(n-1) + w_{\pi_j^l}(n) \qquad [x(n), x(n-1)] \in \pi_j^l.$$

We denote by $\boldsymbol{A}_j^l = [1, a_{\pi_j^l}]/\sqrt{1+a_{\pi_j^l}^2}$ the unit-length vector of the AR model synthesized in the cell π_j^l. Fig. 4(c) plots the set of unit-length vectors for all cells π_j^l resulting from the SVD-tree partition.

The length of each segment is plotted proportionally to the number of points falling into the corresponding cell. Note that this graphical representation clearly reveals the existence of three distinct linear segments governing the state trajectories.

B. Chua Circuit Experiments

We ran experiments on a physical chaotic voltage waveform, measured at the output of a "double-scroll" Chua electronic circuit (see [40], [49], and [63]).

The nonlinear differential equations governing the Chua circuit are

$$\begin{aligned} \frac{dx}{dt} &= \alpha(y-\gamma) \\ \frac{dy}{dt} &= x - y + z \\ \frac{dz}{dt} &= -\beta y. \end{aligned} \tag{17}$$

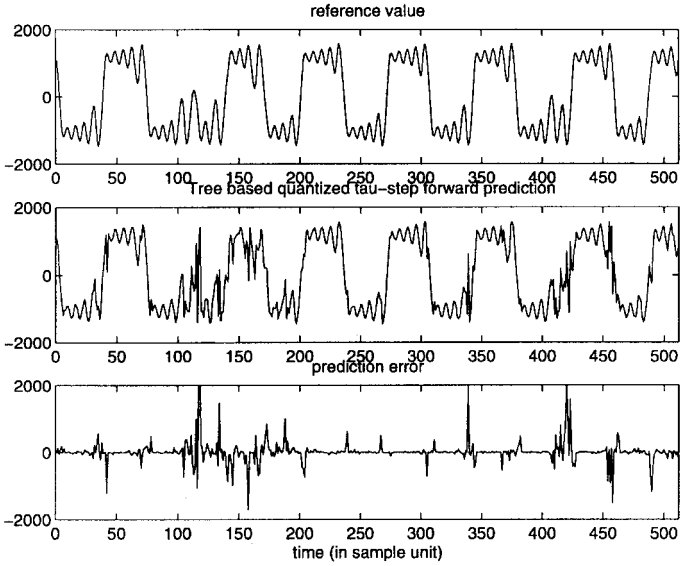
We built the circuit from "off the shelf" components chosen to get the following set of parameter values $\alpha = 9$, $\beta = \frac{100}{7}$, $m_0 = -\frac{1}{7}$, and $m_1 = \frac{2}{7}$. The voltage signal at the output of the electronic circuit was digitized. The sampling frequency was 14.4 kHz. We chose an embedding dimension $\hat{p} = 4$ to generate the state trajectory $\boldsymbol{X}(k) = [x(k), x(k-\tau), x(x-2\tau), x(k-3\tau)]^T$. We used a stopping threshold of $N_\pi > 16$ data points, which corresponds to $\epsilon \simeq 3.10^{-3}$ (for $p = 4$) via (8). The reconstruction delay τ was chosen in such a way as to minimize the mutual information between the coordinates (see [23] and [19]): In this case, $\tau = 4$ sampling periods. A training set of $N_{\Pi_0} = 8192$ points was used to grow the Schur-tree and obtain the empirical histogram $\{N_i/N\}$ on the leaves $\{\pi_i\}$ of the tree. A nonlinear predictor of $x(n)$ given $x(n-1), \cdots, x(n-\hat{p}+1)$ was implemented by approximating the conditional mean $\hat{x}(n) = E[x(n)|x(n-1), \cdots, x(n-\hat{p}+1)]$ using the tree-induced vector quantizer function $Q(\ )$ and the empirical histogram. Specifically, with $Q(x(n), \cdots, x(n-\hat{p}+1)) = \Sigma_i \boldsymbol{q}_i I_{\pi_i}(x(n), \cdots, x(n-\hat{p}+1))$

$$\hat{x}(n) = \frac{\sum_{j=1}^{\hat{p}} \xi_{j1} \hat{P}(\xi_{j1}, q_{n-1}, \cdots, q_{n-\hat{p}+1})}{\sum_{j=1}^{\hat{p}} \hat{P}(\xi_{j1}, q_{n-1}, \cdots, q_{n-\hat{p}+1})} \tag{18}$$

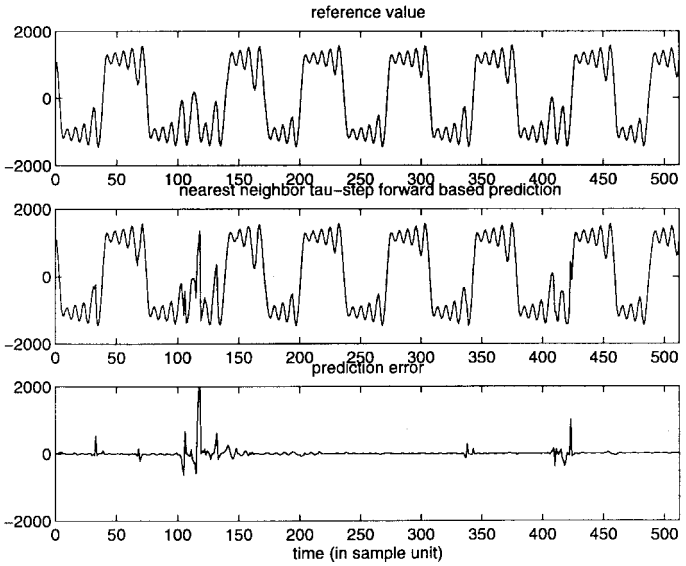
where $\{\xi_i\}$ are centroids of the partition cells $\{\pi_i\}$ at the leaves of the tree, ξ_{j1} denotes the first element of the vector ξ_j, q_{n-1}, $\cdots$, $q_{n-\hat{p}+1}$ are the second through $\hat{p}$th elements of the vector $Q(x(n), \cdots, x(n-\hat{p}+1))$, and $\hat{P}(\boldsymbol{q}) = (1/N)\Sigma_i N_i I_{\pi_i}(\boldsymbol{q})$ is the empirical histogram indexed by $\boldsymbol{q}$.

Fig. 5(a) and (b) show time segments of actual measured and predicted output Chua circuit voltages using the Schur-tree predictor and the popular but costlier nearest neighbor (NN) prediction method, respectively. The NN prediction method is briefly summarized below.

The NN prediction method consists of finding in a learning sequence $L = \boldsymbol{X}(n)$, $n = 1, \cdots, N$ the point $\boldsymbol{X}(j)$, $1 \leq j \leq N$ in the state space that is the closest (in some metric) to the current observation $\boldsymbol{X}(t)$ and defining the predictor as $\boldsymbol{X}(\hat{t}+1) = \boldsymbol{X}(j+1)$. As is shown in Devroye *et al.* [17], under certain technical conditions, the mean squared prediction error of the NN predictor decreases to zero in N. The NN predictor was implemented in a manner identical to the one proposed by Farmer [21]. While more sophisticated implementations of NN predictors are available, see, e.g., [1], [20], and [54], they require higher implementation complexity than Farmer's implementation for only a small improvement in prediction error performance. We performed benchmarks in Matlab 4.2c on a Sun Ultra-1 workstation for 512 one-step predictions of the SETAR model described above. The CPU run times were 33.2 s for SVD-tree versus 115.3 s for the NN prediction algorithm, respectively, with comparable prediction error performance.



(a)



(b)

Fig. 5. One step forward predictor for the sampled output of the Chua electronic circuit. (a) SVD-Tree algorithm. (b) the nearest neighbor algorithm.

C. SETAR Time Series Simulations

Fig. 6 presents results for the simulated SETAR model

$$x(k) = \begin{cases} 1.71x_{k-1} - 0.81x_{k-2} + 0.356 + \varepsilon_k, & x_{k-1} > 0 \\ -0.562x_{k-2} - 3.91 + \varepsilon_k, & x_{k-1} \leq 0. \end{cases}$$

The time series $\{x(k)\}$ was embedded in a three-dimensional (3-D) reconstructed state space ($\hat{p} = 3$) with unit delay τ. The 8-ary Schur-tree was grown according to the methods described in Section IV. Fig. 6(a) shows time segments of the actual and predicted SETAR time series and the associated prediction error. Fig. 6(b) gives a graphical depiction of the 8-ary tree. Fig. 6(c) shows the the estimates of the AR vectors governing the SETAR model in each cell obtained from the recursive local orthogonalization. Note that these estimated AR vectors cluster in two directions that

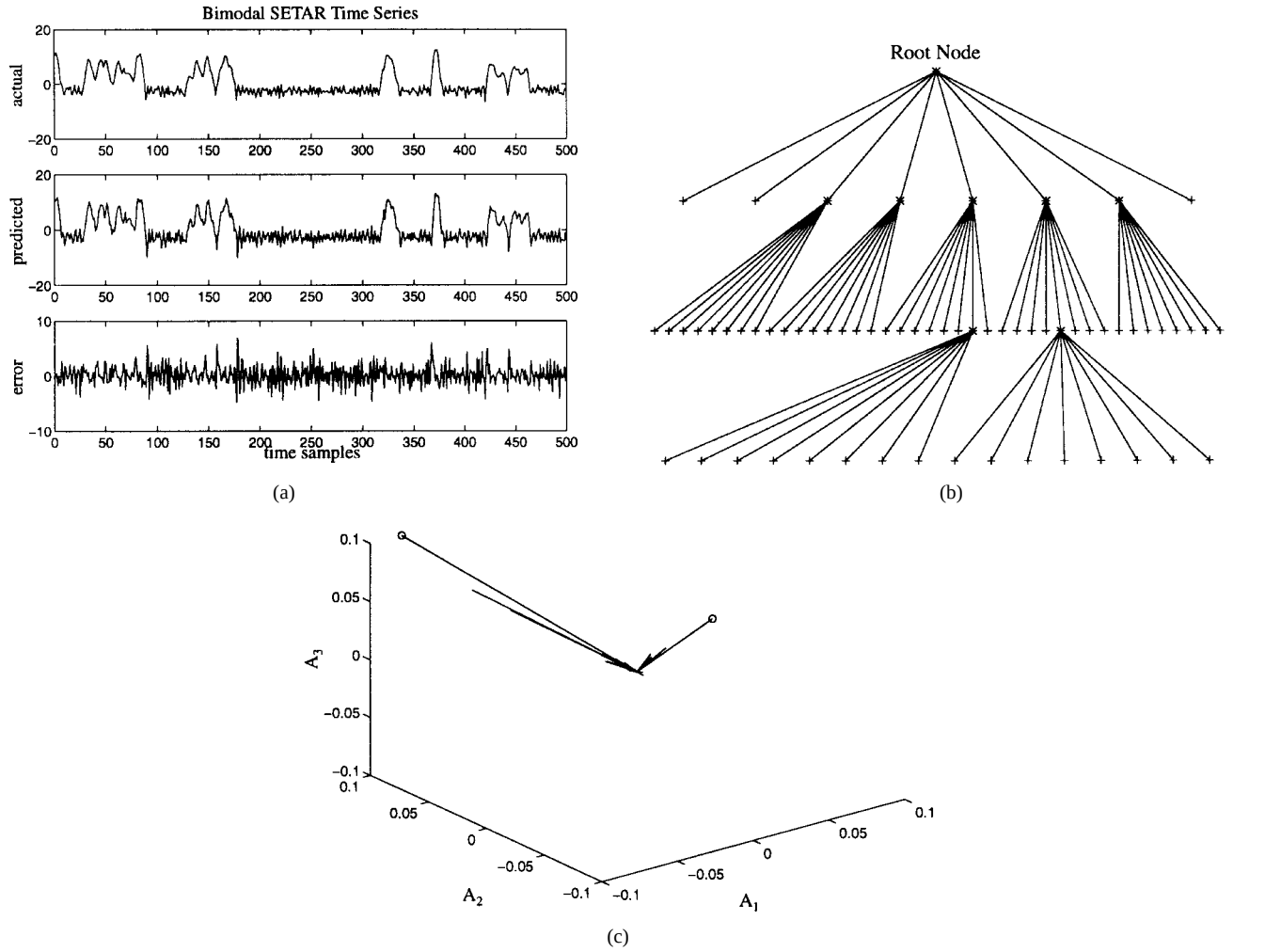


Fig. 6. SETAR time series from (5.3). (a) One step forward predictor trajectory and prediction errors obtained from Schur–Tree algorithm. (b) Eight-ary tree constructed from a 3-D state phase space using Schur–Tree algorithm. (c) Unit-length AR direction vectors.

closely correspond to the two AR(2) regimes of the actual SETAR model (5.3).

D. Rössler Simulations

The discrete-time Rössler system generates a chaotic measurement $x(t)$ generated by the nonlinear set of differential equations

$$\begin{aligned}\frac{dx}{dt} &= -y - z \\ \frac{dy}{dt} &= -x + ay \\ \frac{dz}{dt} &= b + xz - cz \end{aligned} \tag{19}$$

where x, y, and z are components of the 3-D state vector of the Rössler system.

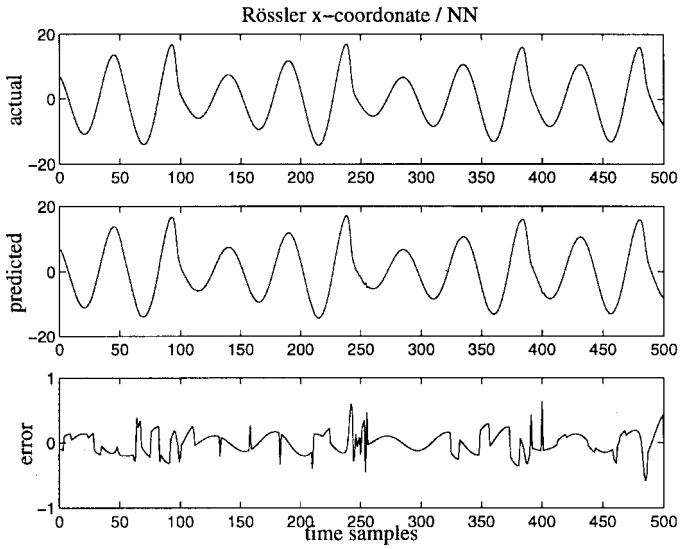
We simulated (19) using the following set of parameter values: $a = 0,15$, $b = 0,2$, and $c = 10$. The set of nonlinear coupled ordinary differential equations were numerically integrated, using an order 3 Runge–Kutta approach. The recorded time series correspond to the first coordinate $(x(t))$ of the system sampled at a period $h = 0.4s$.[2]

The reconstruction dimension was varied from 2–5, but the reconstruction delay is maintained to a constant value $\tau = 4h$. The prediction error variance is estimated from N predicted values by

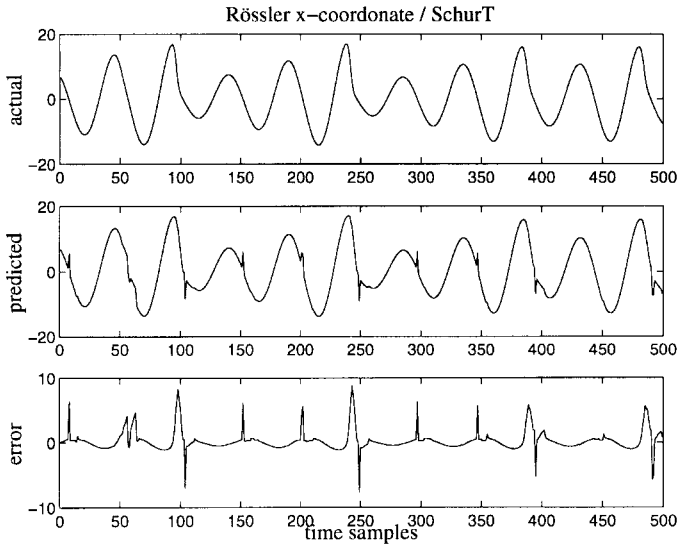
$$V = \frac{\sum_{i=1}^{N}(e_i - \overline{e}_i)^2}{\sum_{i=1}^{N}(x_i - \overline{x}_i)^2}.$$

The Schur-tree was grown from phase space time series of duration $N = 500$, and the training set consisted of 8192 phase space state vectors. Fig. 7 shows the one-step forward prediction and errors for NN and Schur-tree methods applied to the Rössler time series. Note that both NN and Schur-tree predictors have similar trajectories, although the more complex

[2]To simulate this chaotic system by numerically integrating this set of ordinary differential equations, h must be set to a much smaller value than the sampling step of the recorded time series in order to avoid numerical instabilities. The integration was performed with a time increment $h' = h/64$.

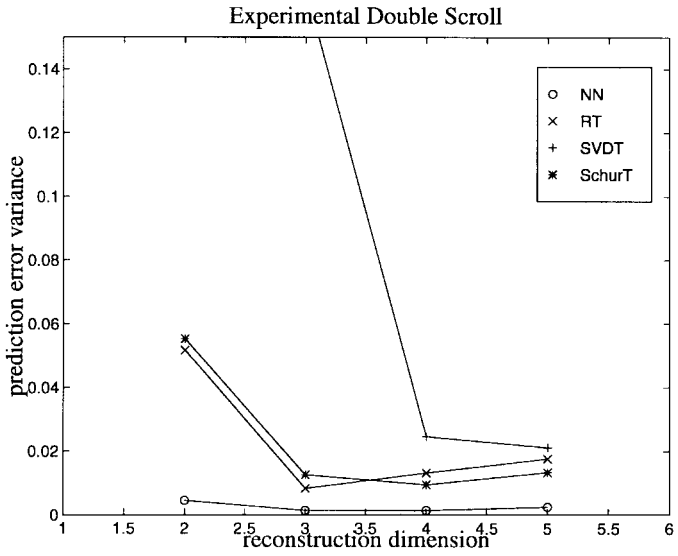


(a)

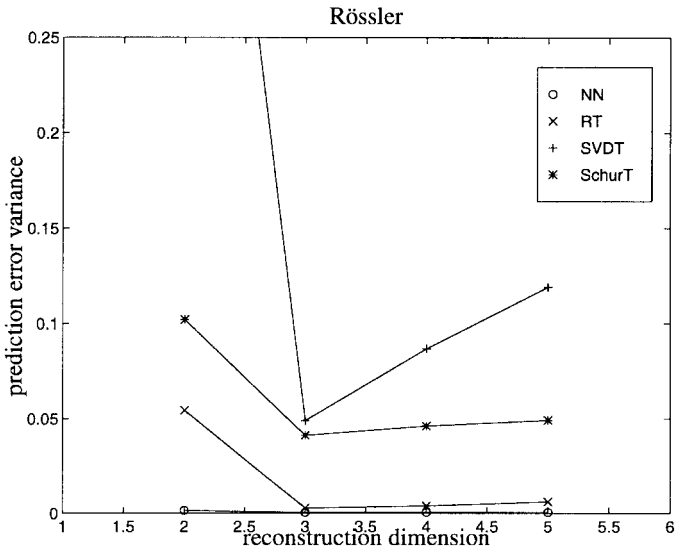


(b)

Fig. 7. Simulated time series, one step forward predicted values, and prediction errors for the first coordinate of the Rössler system ($\hat{p} = 3$) using (a) the NN algorithm and (b) the Schur–tree algorithm.



(a)



(b)

Fig. 8. Normalized prediction error variance as a function of the reconstruction dimension for (a) the voltage output of a Chua electronic circuit and (b) the simulated Rössler time series.

NN implementation achieves somewhat smaller prediction error. The spikes observed in the Schur-tree prediction residuals are due to transitions between the local models in phase space.

E. Algorithm Comparisons

A comparison of the performance of the four different one-step forward prediction methods discussed in this paper is illustrated in Fig. 8 for the Chua circuit measurements and the Rössler simulation. The four methods studied are

- the tree-structured predictor of Section III-C (RT);
- the SVD-tree discussed in Section IV;
- the Schur-tree discussed in Section IV;
- the nearest neighbor (NN) algorithm.

Note the relative performance advantage of the recursive Schur-tree as compared with the SVD-tree. We believe that this is due to the instability of the AR model obtained from SVD-tree; the Schur-tree is guaranteed to give a stable model. In all the cases, the NN algorithm slightly outperforms the tree-based methods, but the improvement is obtained at a significant increase in computational burden.

VI. Conclusions

We have presented a low-complexity algorithm based on recursive state space partitioning for performing near-optimal nonlinear prediction and identification of nonlinear signals. We have also derived local SVD and Schur decomposition versions that are naturally suited to piecewise constant AR models (SETAR). These algorithms were numerically illustrated for simulated SETAR measurements, simulated chaotic measurements, and voltage measurements obtained from a Chua electronic circuit.

The tree based prediction approach presented here is related to the classification and regression tree (CART) technique [9] and adaptive tree-structured vector quantization (TSVQ) [14]. The main difference is our use of a locally defined recursive SVD orthogonalization and its intrinsic applicability to piecewise linear generalizations of thresholded AR (SETAR) models [61]. Our tree structure with SVD orthogonalization is also related to (unitary) transform coding [27], where the difference is that the orthogonalization is applied locally and recursively to each splitting node. Future work will include detection of the local linearized dynamics and regularization for smoothing out model discontinuity between partition cells. A related issue for future study is how to deal with larger values of the imbedding dimension $\hat{p}$. The $2^{\hat{p}}$-ary splitting rule proposed here produces subcells of equal volume but gives a model with complexity, i.e., the number of free parameters, exponential in $\hat{p}$. Therefore, to avoid the need for unreasonably large amounts of training data, $\hat{p}$ must be held as small as possible without sacrificing quantization error performance. A reasonable alternative would be to use the standard binary splitting rule for growing the model: restricting the $2^{\hat{p}}$ splitting rule to implementation of the subcell uniformity tests.

APPENDIX

Let $\boldsymbol{X}$ and Y be real vector ($\mathbb{R}^m$) and scalar valued random variables, respectively. Let the joint distribution of $\boldsymbol{X}, Y$ have the Lebesgue density $f_{X,Y}(\boldsymbol{x}, y)$. Define the marginals $f_X(\boldsymbol{x})$ and $f_Y(y)$ and, for any $\boldsymbol{x}$ satisfying $f(\boldsymbol{x})>0$, the conditional density $f_{Y|X}(y|\boldsymbol{x})$. Given a set $\mathcal{A}$, the conditional density function $f_{Y|X}(y|\boldsymbol{x})$ is said to be Lipschitz continuous of order $\alpha>0$ almost everywhere in $\boldsymbol{x} \in \mathcal{A}$ (in the Hellinger metric) if there exists a finite constant $K_{\mathcal{A}}$, called a Lipschitz constant, such that for any $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{A}$ for which $f_X(\boldsymbol{x}_1), f_X(\boldsymbol{x}_2) > 0$

$$\int |f_{Y|X}^{1/2}(\boldsymbol{y}|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(\boldsymbol{y}|\boldsymbol{x}_2)|^2 \, dy \le K_{\mathcal{A}} ||\boldsymbol{x}_1 - \boldsymbol{x}_2||^{\alpha}. \quad (20)$$

Lipschitz continuity of the above form is a common explicit smoothness condition assumed for probability measures and densities [34], [36]. Lipschitz continuity implies pointwise continuity of $f_{Y|X}(y|\boldsymbol{x})$ for almost all y [34].

For an arbitrary vector $\boldsymbol{x} \in \mathbb{R}^m$ and a discrete set of vectors $\mathcal{Q} = \{\boldsymbol{q}_1, \boldsymbol{q}_2, \cdots\}$ in $\mathbb{R}^m$, let Q, Q: $\mathbb{R}^m \to \mathcal{Q}$, be a vector function (a vector quantizer) operating on $\boldsymbol{x}$. The set of quantization cells $\{\pi_1, \pi_2, \cdots\}$ are defined as the inverse images $\{Q^{-1}(\boldsymbol{q}_1), Q^{-1}(\boldsymbol{q}_2), \cdots\}$ of elements of $\mathcal{Q}$. The following theorem provides a bound on the increase in the minimum mean square prediction error due to quantization of the predictor variables $\boldsymbol{x}$.

Theorem 1: Let $\{\pi_i\}_i$ be a partition of $\mathbb{R}^m$. Assume that for each i the density $f_{Y|X}(y|\boldsymbol{x})$ is Lipschitz continuous of order $\alpha>0$ almost everywhere in $\boldsymbol{x} \in \pi_i$, and let K_{π_i} be the associated Lipschitz constant. Assume also that $E[Y^2|\boldsymbol{X}] \le m_Y^2 < \infty$ (a.s.). Then

$$\begin{aligned} 0 &\le E[(Y - E[Y|Q(\boldsymbol{X})])^2] - E[(Y - E[Y|\boldsymbol{X}])^2] \\ &\le 2 \max_i K_{\pi_i} m_Y^2 E[||\boldsymbol{X} - Q^o(\boldsymbol{X})||^{\alpha}] \end{aligned} \quad (21)$$

where $Q^o(\boldsymbol{x}) = \Sigma_i \boldsymbol{x}i_i I_{\pi_i}(\boldsymbol{x})$ and $\boldsymbol{x}i_i \in \mathbb{R}^m$ are the quantization vectors defined in Lemma 2.

The upper bound in (21) is decreasing in $\max_i K_{\pi_i}$ and equals zero when $f(y|\boldsymbol{x})$ is piecewise constant in $\boldsymbol{x}$, i.e., $f(y|\boldsymbol{x}) = \Sigma_i f(y|\boldsymbol{x}i_i) I_{\pi_i}(\boldsymbol{x})$, where $\boldsymbol{x}i_i \in \pi_i$ are arbitrary. Thus, in this case, use of quantized predictor variables do not degrade optimal prediction MSE. In addition, note that the upper bound in (21) is decreasing in the mean square quantization error associated with quantizing the predictor variables $E[||\boldsymbol{X} - Q(\boldsymbol{X})||^2]$. Bounds and asymptotic expressions exist for this quantity [29], [42] which can be used to make the bound (21) more explicit.

The following lemmas will be useful in the proof of Theorem 1.

Lemma 1: Define the optimal predictor $\hat{\mu}_{Y|X}(\boldsymbol{X}) = E[Y|\boldsymbol{X}]$ based on the predictor variables $\boldsymbol{X}$. Assume that for some subset $\mathcal{A}$ of $\mathbb{R}^m$, the density $f_{Y|X}(y|\boldsymbol{x})$ is Lipschitz continuous of order α almost everywhere in $\boldsymbol{x} \in \mathcal{A}$ and that $E[Y^2|\boldsymbol{X}] \le m_Y^2 < \infty$ (a.s.). Then, $\hat{\mu}_{Y|X}(\boldsymbol{x})$ is pointwise continuous almost everywhere over $\boldsymbol{x} \in \mathcal{A}$.

Proof of Lemma 1: First, observe that for any two functions f_1 and f_2, we have by the triangle inequality

$$\begin{aligned} |f_1 - f_2| &= |f_1^{1/2}(f_1^{1/2} - f_2^{1/2}) + f_2^{1/2}(f_1^{1/2} - f_2^{1/2})| \\ &\le |f_1^{1/2}|\,|f_1^{1/2} - f_2^{1/2}| + |f_2^{1/2}|\,|f_1^{1/2} - f_2^{1/2}|. \end{aligned} \quad (22)$$

Therefore, by definition of the conditional mean, for arbitrary $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{A}$

$$\begin{aligned} &|\hat{\mu}_{Y|X}(\boldsymbol{x}_1) - \hat{\mu}_{Y|X}(\boldsymbol{x}_2)| \\ &\quad \le \int dy|y|\,|f_{Y|X}(y|\boldsymbol{x}_1) - f_{Y|X}(y|\boldsymbol{x}_2)| \\ &\quad \le \int dy|y| f_{Y|X}^{1/2}(y|\boldsymbol{x}_1)|f_{Y|X}^{1/2}(y|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)| \\ &\qquad + \int dy|y| f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)|f_{Y|X}^{1/2}(y|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)|. \end{aligned} \quad (23)$$

Applying the Cauchy–Schwartz inequality to the two integrals in the expression at bottom of (23)

$$\begin{aligned} &\left(\int dy|y| f_{Y|X}^{1/2}(y|\boldsymbol{x}_1)|f_{Y|X}^{1/2}(y|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)|\right)^2 \\ &\quad \le \int dy|y|^2 f_{Y|X}(y|\boldsymbol{x}_1) \\ &\qquad \cdot \int dy|f_{Y|X}^{1/2}(y|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)|^2 \end{aligned}$$

and similarly for the second integral. Hence, Lipschitz continuity of $f_{Y|X}(y|\boldsymbol{x})$ over $\boldsymbol{x} \in \mathcal{A}$ gives the bound

$$\begin{aligned} &|\hat{\mu}_{Y|X}(\boldsymbol{x}_1) - \hat{\mu}_{Y|X}(\boldsymbol{x}_2)|^2 \\ &\quad \le 2 \max_{\boldsymbol{x}} E[y^2|\boldsymbol{x}] \int dy|f_{Y|X}^{1/2}(y|\boldsymbol{x}_1) - f_{Y|X}^{1/2}(y|\boldsymbol{x}_2)|^2 \\ &\quad \le 2 m_Y^2 K_{\mathcal{A}} ||\boldsymbol{x}_1 - \boldsymbol{x}_2||^{\alpha} \end{aligned} \quad (24)$$

where $K_{\mathcal{A}} < \infty$ is the associated Lipschitz constant. This establishes the lemma. □

Lemma 2: Define the optimal predictor $\hat{\mu}_{Y|Q}(\boldsymbol{x}) = E[Y|Q(\boldsymbol{X})]$ based on quantized predictor variables $Q(\boldsymbol{X})$. Assume that $f_{Y|X}(y|\boldsymbol{x})$ is Lipschitz continuous of order α almost everywhere in $\boldsymbol{x} \in \pi_i$ and that $E[Y^2|\boldsymbol{X}] \le m_Y^2 < \infty$ (a.s.). Then, for any quantization cell $\pi_i \subset \mathbb{R}^m$, there exists

a point $\boldsymbol{x}i_i \in \pi_i$ such that

$$\hat{\mu}_{Y|Q}(\boldsymbol{x}) = \hat{\mu}_{Y|X}(\boldsymbol{x}i_i), \quad \forall \boldsymbol{x} \in \pi_i.$$

Furthermore, the point $\boldsymbol{x}i_i$ satisfies the equation

$$\int_{\pi_i} d\boldsymbol{x}\hat{\mu}_{Y|X}(\boldsymbol{x})f(\boldsymbol{x}) = \hat{\mu}_{Y|X}(\boldsymbol{x}i_i)P_X(\pi_i)$$

where $P_X(\pi_i) = \int_{\pi_i} f(\boldsymbol{x})\,d\boldsymbol{x}$.

Proof of Lemma 2: By definition of conditional expectation, $\hat{\mu}_{Y|Q}(\boldsymbol{x}) = \int dyyf_{Y|Q}(y|Q(\boldsymbol{x}))$, where

$$f_{Y|Q}(y|Q(\boldsymbol{x})) = \int_{\pi_i} d\boldsymbol{x} f_{Y|X}(y|\boldsymbol{x})f(\boldsymbol{x})/P_X(\pi_i), \qquad \boldsymbol{x} \in \pi_i$$

is the conditional density of Y given $Q(\boldsymbol{X}) = Q(\boldsymbol{x})$. Invoking Fubini's theorem [53] to permute the order of integration, we obtain the Lebesgque–Steiltjes integral representation

$$\begin{aligned}\hat{\mu}_{Y|Q}(\boldsymbol{x}) &= \frac{1}{P_X(\pi_i)} \int_{\pi_i} d\boldsymbol{x} f(\boldsymbol{x}) \int dyyf_{Y|X}(y|\boldsymbol{x}) \\ &= \frac{1}{P_X(\pi_i)} \int_{\pi_i} dP(\boldsymbol{x})\hat{\mu}_{Y|X}(\boldsymbol{x}) \qquad \boldsymbol{x} \in \pi_i\end{aligned}$$

where $dP(x) = f(x)\,dx$. By Lemma 1, $\hat{\mu}_{Y|X}(\boldsymbol{x})$ is continuous, and therefore, by the mean value theorem for Lebesgue–Steiltjes integrals [53], there exists a point $\xi_i \in \pi_i$ such that

$$\frac{1}{P_X(\pi_i)} \int_{\pi_i} dP(\boldsymbol{x})\hat{\mu}_{Y|X}(\boldsymbol{x}) = \hat{\mu}_{Y|X}(\boldsymbol{x}i_i), \qquad \boldsymbol{x} \in \pi_i.$$

This establishes the Lemma. □

Proof of Theorem 1: Define

$$\Delta^2 \stackrel{\text{def}}{=} E[(Y - E[Y|Q(\boldsymbol{X})])^2] - E[(Y - E[Y|\boldsymbol{X}])^2].$$

That $\Delta^2 > 0$ follows directly from the fact that the conditional mean estimator $E[Y|\boldsymbol{X}]$ minimizes mean square prediction error. Next, we deal with the right-hand side of (21). It is easily verified by iterated expectation that $E[(Y - E[Y|Q(\boldsymbol{X})])E[Y|Q(\boldsymbol{X})]] = 0$, and $E[(Y - E[Y|\boldsymbol{X}])E[Y|\boldsymbol{X}]] = 0$ (orthogonality principle of nonlinear estimation). Therefore

$$\begin{aligned}\Delta^2 &= E[(Y - E[Y|Q(\boldsymbol{X})])Y] - E[(Y - E[Y|\boldsymbol{X}])Y] \\ &= E[(E[Y|\boldsymbol{X}] - E[Y|Q(\boldsymbol{X})])Y] \\ &= E[(E[Y|\boldsymbol{X}] - E[Y|Q(\boldsymbol{X})])E[Y|\boldsymbol{X}]].\end{aligned}$$

Thus, by Fubini [8], we have the integral representation

$$\begin{aligned}\Delta^2 &= \int d\boldsymbol{x}[\hat{\mu}_{Y|X}(\boldsymbol{x}) - \hat{\mu}_{Y|Q}(\boldsymbol{x})]\hat{\mu}_{Y|X}(\boldsymbol{x})f(\boldsymbol{x}) \\ &= \sum_i \int_{\pi_i} d\boldsymbol{x}[\hat{\mu}_{Y|X}(\boldsymbol{x}) - \hat{\mu}_{Y|Q}(\boldsymbol{x})]\hat{\mu}_{Y|X}(\boldsymbol{x})f(\boldsymbol{x}) \qquad (25)\end{aligned}$$

where the $\hat{\mu}$ quantities are defined as in Lemmas 1 and 2. Invoking the latter lemma, there exists a point $\boldsymbol{x}i_i \in \pi_i$ such that $\hat{\mu}_{Y|Q}(\boldsymbol{x}) = \hat{\mu}_{Y|X}(\boldsymbol{x}i_i)$, $\boldsymbol{x} \in \pi_i$, and $\int_{\pi_i} d\boldsymbol{x}[\hat{\mu}_{Y|X}(\boldsymbol{x}) - \hat{\mu}_{Y|X}(\boldsymbol{x}i_i)]f(\boldsymbol{x}) = 0$. Therefore, from (25)

$$\Delta^2 = \sum_i \int_{\pi_i} d\boldsymbol{x}|\hat{\mu}_{Y|X}(\boldsymbol{x}) - \hat{\mu}_{Y|X}(\boldsymbol{x}i_i)|^2 f(\boldsymbol{x}).$$

Application of the bound (24) on $|\hat{\mu}_{Y|X}(\boldsymbol{x}_1) - \hat{\mu}_{Y|X}(\boldsymbol{x}_2)|^2$ obtained in the course of proving Lemma 1 yields

$$\begin{aligned}\Delta^2 &\le 2m_Y^2 \cdot \max_i K_{\pi_i} \sum_i \int_{\pi_i} d\boldsymbol{x}||\boldsymbol{x} - \boldsymbol{x}i_i||^\alpha f(\boldsymbol{x}) \\ &= 2m_Y^2 \cdot \max_i K_{\pi_i} E[||\boldsymbol{X} - Q^o(\boldsymbol{X})||^\alpha]\end{aligned}$$

where $Q^o(\boldsymbol{x}) = \Sigma_i \boldsymbol{x}i_i I_{\pi_i}(\boldsymbol{x})$. □

REFERENCES

[1] H. Abarbanel, R. Brown, J. Sidorowitch, and L. Tsimring, "The analysis of observed chaotic data in physical systems," *Rev. Mod. Phys.*, vol. 65, no. 4, pp. 1331–1391, 1990.

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

[3] M. Abromowitz and I. Stegun, *Handbook of Mathematical Functions*. New York: Dover, 1977.

[4] M. Basseville, "Distances measures for signal processing and pattern recognition," *Signal Process.*, vol. 18, pp. 349–369.

[5] J. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. Assoc. Comput. Mach.*, vol. 18, pp. 509–517, 1975.

[6] A. E. Badel, O. Michel, and A. O. Hero, "Arbres de régression: Modélization non paramétrique et analyze des Séries Temporelles," *Rev. Traitement Signal*, vol. 14, no. 2, pp. 117–133, June 1997.

[7] P. J. Bickel and K. A. Doksum, *Mathematical Statistics: Basic Ideas and Selected Topics*. San Francisco, CA: Holden-Day, 1977.

[8] P. Billingsley, *Probability and Measure*. New York: Wiley, 1979.

[9] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "*Classification and Regression Trees*. New York: Wadsworth, 1984.

[10] A. E. Badel, L. Mercier, D. Guegan, and O. Michel, "Comparison of several methods to predict chaotic time series," in *Proc. IEEE-ICASSP*, Munich, Germany, Apr. 1997, vol. V, pp. 3793–3796.

[11] M. Casdagli, T. Sauer, and J. A Yorke, "Embedology," *J. Stat. Phys.*, vol. 65, pp. 579–616, 1991.

[12] M. Casdagli and S. Eubank, Eds., "Nonlinear modeling and forecasting," in *Proc. Inst. Studies Sci. Complexity*, Santa Fe, NM, 1991, vol. 12.

[13] L. Clarke and D. Pregibon, "Tree-based models," in *Statistical Models in S*, J. Chambers and T. Hastie, Eds. New York: Wadsworth, 1992, pp. 377–419.

[14] P. A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structures source coding and modeling," *IEEE Trans. Inform. Theory*, vol. 35, pp. 299–315, Apr. 1989.

[15] W. Cleveland, E. Grosse, and W. Shyu, "Local regression models," in *Statistical Models in S*, J. Chambers and T. Hastie, Eds. New York: Wadsworth, 1992, pp. 309–376.

[16] H. A. David, *Order Statistics*. New York: Wiley, 1981.

[17] L. Devroye, L. Györfi, and G. Lugosi, "A probabilistic theory of pattern recognition," *Appl. Math. Series*, vol. 31, 1996.

[18] R. F. Engle, "Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation," *Econometrica*, vol. 50, pp. 987–1007, 1982.

[19] J. P. Eckmann and D. Ruelle, "Ergodic theory of chaos and strange attractors," *Rev. Mod. Phys.*, vol. 57, no. 3, pp. 617–656, 1985.

[20] J. D. Farmer and J. Sidorowitch, "Predicting chaotic time series," *Phys. Rev. Lett.*, vol. 59, p. 845, 1987

[21] ——, "Exploiting chaos to predict the future and reduce noise," in *Evolution, Learning, and Cognition*, Y. C. Lee, Ed. Singapore: World Scientific, 1988, pp. 277–330.

[22] A. M. Fraser, "Information and entropy in strange attractors," *IEEE Trans. Inform. Theory*, vol. 35, pp. 245–262, Apr. 1989.

[23] A. M. Fraser and H. L. Swinney, "Independent coordinates for strange attractors from mutual information," *Phys. Rev. A*, vol. 33, no. 2, pp. 1134–1140, 1981.

[24] A. M. Fraser, "Modeling nonlinear time series," in *Proc. ICASSP*, San Francisco, CA, 1992, vol. 5, pp. V313–V317.

[25] A. M. Fraser and A. Dimitriadis, "Forecasting probability densisty by using hidden Markov models with mixed states," *Time Series Prediction*, A. S. Weigend and N. A. Gershenfeld, Eds., vol. XV, pp. 265–282.

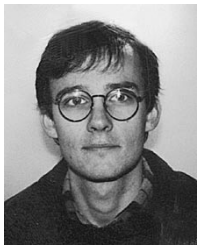
[26] K. Fukunaga, *Statistical Pattern Recognition*, 2nd ed. San Diego, CA: Academic, 1990.

[27] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Boston, MA: Kluwer, 1992.

[28] P. Grassberger and I. Procaccia, "Measuring the strangeness of strange attractors," *Phys. D*, vol. 9, pp. 189–208, 1983.

[29] R. M. Gray, *Source Coding Theory*. Norwell, MA: Kluwer, 1990.

[30] D. Guegan, "On the identification and prediction of nonlinear models," in *Proc. Workshop New Directions Time Series Anal*, 1992.
[31] ——, "Séries chronologiques non linéaires à temps discret," *Stat. Math. Probab.*, 1994.
[32] S. Haykin and J. Principe "Using neural networks to dynamically model chaotic events such as sea clutter; Making sense of a complex world," *IEEE Signal Processing Mag.*, p. 81, May 1998.
[33] D. R. Hush and B. G. Horne "Progress in supervised neural networks," *IEEE Signal Processing Mag.*, p. 38, Jan. 1993.
[34] I. A. Ibragimov and R. Z. Has'minskii, *Statistical Estimation: Asymptotic Theory*. New York: Springer-Verlag, 1981.
[35] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1997.
[36] L. LeCam, *Asymptotic Methods in Statistical Decision Theory*. New York: Springer-Verlag, 1986.
[37] E. L. Lehmann, *Testing Statistical Hypotheses*. New York: Wiley, 1959.
[38] W. Liebert, K. Pawelzik, and H. G. Schuster, "Optimal embedding of chaotic attractors from topological considerations," *Europhys. Lett.*, vol. 14, no. 6, pp. 521–526, 1991.
[39] A. Mead *et al.*, "Prediction of chaotic time series using CNLS-net-example: The mackey-Glass equation," in *Nonlinear Modeling and Forecasting*, M. Casdagli and S. Eubank, Eds. Reading, MA: Addison-Wesley, 1992, vol. 12, pp. 39–72.
[40] O. Michel and P. Flandrin, "Higher order statistics for chaotic signal processing," *Contr. Dyn. Syst.*, vol. 75, pp. 105–153, 1996.
[41] A. M. Mood, F. A. Graybill, and D. C. Boes, *Introduction to the Theory of Statistics*, 3rd ed. New York: McGraw-Hill, 1974.
[42] D. N. Neuhoff, "On the asymptotic distribution of the errors in vector quantization," *IEEE Trans. Inform. Theory*, vol. 42, pp. 461–468, Mar. 1996.
[43] A. Nobel, "Vanishing distortion and shrinking cells," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1303–1305, Aug. 1996.
[44] ——, "Recursive partitioning to reduce distortion," *IEEE Trans. Inform. Theory*, vol. 43, pp. 1122–1133, Aug. 1997.
[45] A. Nobel and R. Olshen, "Termination and continuity of greedy growing for tree-structured vector quantizers," *IEEE Trans. Inform. Theory*, vol. 42, pp. 191–205, Feb. 1996.
[46] M. Orchard and C. Bouman, "Color quantization of images," *IEEE Trans. Signal Processing*, vol. 39, pp. 2677–2690, Dec. 1991.
[47] K. Perlmutter, S. Perlmutter, R. Gray, R. Olshen, and K. Oehler, "Bayes risk weighted vector quantization with posterior estimation for image compression and classification," *IEEE Trans. Image Processing*, vol. 5, pp. 347–360, Feb. 1996.
[48] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 2nd ed. New York: McGraw-Hill, 1984.
[49] T. S. Parker and L. O. Chua, *Practical Numerical Algorithms for Chaotic Systems*. New York: Springer-Verlag, 1989.
[50] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. Cambridge, U.K.: Cambridge Univ. Press, 1989.
[51] M. B. Priestley, "State dependant models: A general approach to nonlinear time series analysis," *J. Time Series Anal.*, vol. 1, pp. 47–71, 1980.
[52] ——, *Non Linear and Non Stationary Time Series Analysis*. San Diego, CA: Academic, 1988.
[53] F. Riesz and B. S. Nagy, *Functional Analysis*. New York: Ungar, 1955.
[54] T. Sauer, "A noise reduction method for signal from nonlinear systems," *Phys. D*, vol. 58, pp. 193–201, 1992.
[55] M. R. Segal, "Tree structured methods for longitudinal data," *J. Amer. Stat. Assoc.*, vol. 87, pp. 407–418, May 1992.
[56] L. L. Scharf, *Statistical Signal Processing, Detection, Estimation and Time Series Analysis*. Reading, MA: Addison-Wesley, 1990.
[57] R. Shaw, "Strange attractors, chaotic behavior, and information flow," *Zeitschrift NaturForschung*, vol. 36A, no. 1, pp. 80–112, 1981.
[58] J. N. Sonquist and J. N. Morgan, "The detection of interaction effects," Monograph 35, Survey Res. Cent., Inst. Soc. Res., Univ. Michigan, Ann Arbor, 1964.
[59] F. Takens, "Detecting strange attractors in turbulence," *Lecture Notes Math.*, vol. 898, pp. 366–381, 1981.
[60] H. Tong "Threhold models in nonlinear time series analysis," *Lecture Notes Stat.*, vol. 21, 1983.
[61] ——, *Nonlinear Time Series: A Dynamical system Approach*. New York: Oxford Univ. Press, 1990.
[62] G. Wahba, "Multivariate function and operator estimation, based on smoothing splines and reproducing kernels," in *Nonlinear Modeling and Forecasting*, M. Casdagli and S. Eubank, Eds. Reading, MA: Addison Wesley, 1992, vol. 12, pp. 95–112.
[63] T. P. Weldon, "An inductorless double-scroll chaotic circuit," *Amer. J. Phys.*, vol. 58, no. 10, pp. 936–941, 1990.
[64] A. S. Weigend and N. A. Gershenfeld, Eds., "Time series prediction: Forecasting the future and understanding the past," *Proc. Inst. Studies Sci. Complexity*, Santa Fe, NM, 1992, vol. 15.
[65] H. Whitney, "Differentiable manifolds," *Ann. Math.*, vol. 37, no. 3, pp. 645–680, 1936.
[66] H. Zhang, "Classification trees for multiple binary responses," *J. Amer. Stat. Assoc.*, vol. 93, no. 441, pp. 180–193, Mar. 1998.



Olivier J. J. Michel (S'84–M'85) was born in Mont Saint Martin, France, in 1963. He received the Agrégation de Physique degree from the Department of Applied Physics, Ecole Normale Supérieure de Cachan, Cachan, France, in 1986. He received the Ph.D degree from University Paris-XI, Orsay, in signal processing in 1991.

In 1991, he joined the Physics Department, École Normale Supérieure, Lyon, France, as an assistant professor. His research interest include nonstationnary spectral analysis, array processing, nonlinear time series problems, information theory, and dynamical systems studies, in close relationship with physical experiments in the field of chaos and hydrodynamical turbulence.

Alfred O. Hero, III (S'79–M'84–SM'96–F'97) was born in Boston, MA, in 1955. He received the B.S. (summa cum laude) from Boston University in 1980 and the Ph.D. from Princeton University, Princeton, NJ, in 1984, both in electrical engineering.

Since 1984, he has been with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, where he is currently Professor and Director of the Communications and Signal Processing Laboratory. He has held positions of Visiting Scientist at Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, from 1987 to 1989; Visiting Professor at l'École Nationale de Techniques Avancees (ENSTA), Paris, France, in 1991; and William Clay Ford Fellow at the Ford Motor Company, Dearborn, MI, in 1993. He has served as a consultant for U.S. government agencies and private industry. His present research interests are in the areas of detection and estimation theory, statistical signal and image processing, statistical pattern recognition, signal processing for communications, channel equalization and interference mitigation, spatio-temporal sonar and radar processing, and biomedical signal and image analysis.

Dr. Hero is a member of Tau Beta Pi, the American Statistical Association, the New York Academy of Science, and Commission C of the International Union of Radio Science (URSI). He held the G.V.N. Lothrop Fellowship in Engineering at Princeton University. In 1995, he received a Research Excellence Award from the College of Engineering at the University of Michigan. In 1999, he received a Best Paper Award from the IEEE Signal Processing Society. He was Associate Editor for the IEEE TRANSACTIONS ON INFORMATION THEORY from 1994 to 1997; Chair of the IEEE SPS Statistical Signal and Array Processing Technical Committee from 1996 to 1998; and Treasurer of the IEEE SPS Conference Board from 1997 to 2000. He was co-chair for the 1999 IEEE Information Theory Workshop and the 1999 IEEE Workshop on Higher Order Statistics. He served as Publicity Chair for the 1986 IEEE International Symposium on Information Theory and was General Chair of the 1995 IEEE International Conference on Acoustics, Speech, and Signal Processing. He received the 1999 Meritorious Service Award from the IEEE Signal Processing Society.

Anne Emmanuelle Badel (S'92–M'93) was born in Lyon, France, in 1971. She received the Agrégation de Physique degree in 1994 from the the Physics Department, École Normale Supérieure (ENS), Lyon, France. She received the Ph.D. degree in physics from ENS in 1998.

Her research interests are in nonlinear time series analysis.