Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT®

Signal Processing 85 (2005) 1859–1868

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Short communication

Symbolic time series analysis for anomaly detection: A comparative evaluation☆

Shin C. Chin, Asok Ray*, Venkatesh Rajagopalan

The Pennsylvania State University, University Park, PA 16802, USA

Received 24 June 2004; received in revised form 21 December 2004

Abstract

Recent literature has reported a novel method for anomaly detection in complex dynamical systems, which relies on symbolic time series analysis and is built upon the principles of *automata theory* and *pattern recognition*. This paper compares the performance of this symbolic-dynamics-based method with that of other existing pattern recognition techniques from the perspectives of early detection of small anomalies. Time series data of observed process variables on the fast time-scale of dynamical systems are analyzed at slow time-scale epochs of (possible) anomalies. The results are derived from experiments on a nonlinear electronic system with a slowly varying dissipation parameter.

Keywords: Fault detection; Symbolic dynamics; Pattern recognition; Complex systems

1. Introduction

Anomaly in a dynamical system is defined as a deviation from its nominal behavior and can be associated with parametric or non-parametric changes that may gradually evolve in the system. Along this line, Ray [1] reported a novel concept of anomaly detection in complex dynamical systems via symbolic time series analysis. The concept is built upon a fixed-structure, fixed-order Markov chain and is called the *D-Markov machine* in the sequel. The *D*-Markov machine makes use of symbolic time series analysis [2] as a means to capture the coarse-grained dynamical behavior in terms of symbol sequences [3]. Deviation from the nominal behavior is represented by a change in the pattern of symbol sequences, which is detectable at an early stage.

The objective of this paper, which is an extension of the previous paper [1], is to assess the performance and efficacy of the *D*-Markov machine method for anomaly detection [4] by comparison with other existing pattern recognition techniques, such as principal component analysis,

☆This work has been supported in part by Army Research Office (ARO) under Grant No. DAAD19-01-1-0646; and NASA Glenn Research Center under Grant No. NNC04GA49G.

*Corresponding author. Tel./fax: +1 814 865 6377.
E-mail address: axr2@psu.edu (A. Ray).

0165-1684/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2005.03.014

mutilayer perceptron neural networks, and radial basis function neural networks [5,6]. Time series data, generated from laboratory experiments on a nonlinear electronic system, are analyzed to this effect.

2. Existing pattern recognition techniques

This section briefly describes three major classes of pattern recognition techniques, which are based on time series data, for comparison with the *D-Markov machine* [1]:

- Syntactic or structural matching
- Statistical pattern recognition
- Neural networks

The above classes of pattern recognition techniques may not be mutually non-overlapping. It is possible that the same pattern recognition method can be interpreted to belong to more than one class.

2.1. Syntactic methods

The syntactic approach adopts a hierarchical perspective, where a pattern is viewed to be composed of simple subpatterns that are themselves built from yet simpler subpatterns. The underlying concept of pattern matching in the syntactic approach is similar to the comparison of the probability vectors under nominal and anomalous cases in the proposed *D*-Markov machine [1]. One of the major shortcomings of conventional syntactic pattern recognition [5] is utilization of noisy patterns for detection of the primitives and making the associated inference of the grammar from the time series data. In contrast, the *D*-Markov machine is significantly robust to measurement noise because it extracts the averaged features of signal dynamics from repeated transitions among the finitely many states.

2.2. Statistical methods

In the statistical approach, each pattern is represented in terms of d features or measurements and is viewed as a point in a d-dimensional space. The goal is to choose those features that allow pattern vectors belonging to different categories to occupy compact and disjoint regions in the d-dimensional feature space.

Feature extraction methods in statistical pattern recognition determine an appropriate subspace of dimension $q \in \mathbb{N}$, where $\mathbb{N}$ is the set of positive integers, using either linear or nonlinear methods in the original feature space of dimension n ($q \leqslant n$). The best known linear feature extractor relies on the principal component analysis (PCA) or Karhunen–Loève expansion [5,7]. The eigenvectors of the ($n \times n$ positive semi-definite) covariance matrix of the time series data, corresponding to the q largest eigenvalues, form the n-dimensional patterns.

If the time response of an appropriate process variable $y(t)$ is sampled to generate a time series sequence y_k, then data samples of large enough length ($\ell = dn$) can be used to capture the dynamical characteristics of the observed process. The length ℓ of time series data is partitioned into d subsections, each being of length $n = \ell/d$. The resulting ($d \times n$) data matrix is processed to generate the ($n \times n$) covariance matrix that is positive-definite or positive-semidefinite real-symmetric. The next step is to compute the orthonormal eigenvectors $v^1, v^2, \ldots, v^n$ and the corresponding eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$ that are arranged in decreasing orders of magnitude.

Dimensionality of the model, formulated from time series data, is reduced for feature extraction. The eigenvectors associated with the first (i.e., largest) q eigenvalues are chosen as the feature vectors such that

$$\frac{\sum_{i=q+1}^{n} \lambda_i}{\sum_{i=1}^{n} \lambda_i} < \eta, \tag{1}$$

where the threshold $\eta \ll 1$ is a positive real close to 0. The resulting pattern is the matrix, consisting of the feature vectors as columns,

$$\widetilde{M} = \left(\sqrt{\frac{\lambda_1}{\sum_{k=1}^{n} \lambda_k}} v_1 \quad \ldots \quad \sqrt{\frac{\lambda_q}{\sum_{k=1}^{n} \lambda_k}} v_q \right). \tag{2}$$

The above steps are executed for time series data under the nominal (stationary) condition to obtain

$\widetilde{M}_{\text{nom}}$. Then, these steps are repeated at subsequent observations at slow-time epochs, $\{t_1, t_2, \ldots\}$, as the (possible) anomaly progresses using the same values of parameters, ℓ, d, n and q, used under the nominal condition, to obtain the respective pattern matrices $\widetilde{M}_1, \widetilde{M}_2, \ldots$. The anomaly measures at slow-time epochs $\{t_1, t_2, \ldots\}$ are obtained as

$$\mathscr{M}_k \equiv d(\widetilde{M}_k, \widetilde{M}_{\text{nom}}), \tag{3}$$

where the $d(\bullet, \bullet)$ is an appropriately defined distance function.

It should be noted that different metrics may be used as anomaly measures as stated in [1]. One may choose the metric that yields the most satisfactory result for the specified purpose.

2.3. Neural network methods

The most commonly used family of feedforward neural networks for pattern classification tasks include the following [5]:

- Multilayer perceptron neural networks (MLPNN)
- Radial basis Function neural networks (RBFNN)

2.3.1. MLPNN for anomaly detection

The simplest implementation of back-propagation learning in MLPNN updates the network weights and biases in the direction in which the performance function decreases most rapidly. The *mean-square error* criterion is adopted in the recursive algorithm to update the weight matrix $\mathbf{w}^k(n)$ at the kth layer of the network in the nth iteration [8] as follows:

$$\mathbf{w}^k(n+1) = \mathbf{w}^k(n) - \alpha^k \mathbf{g}^k(n), \tag{4}$$

where $\mathbf{g}^k(n)$ is the gradient of the (averaged) functional of the output error vectors (i.e., the difference between the target vector and the output vector) with respect to $\mathbf{w}^k(n)$; and α^k is the learning rate parameter at the kth layer of the network.

Different layers of a given MLPNN may contain different numbers of neurons. Time series signals enter into the input layer nodes, progress forward through the hidden layers, and finally emerge from the output layer. Each node i at a given layer k receives a signal from all nodes j in its preceding layer $(k-1)$ through a synapsis of weight w_{ij}^k and the process is carried onto the nodes in the following layer $(k+1)$. The weighted sum of signals x_j^{k-1} from all nodes j of the layer $(k-1)$ together with a bias w_{i0}^k produces the excitation z_i^k that, in turn, is passed through a nonlinear *activation function* f to generate the output x_i^k from the node i at the layer k. That is,

$$z_i^k = \sum_j w_{ij}^k x_j^{k-1} + w_{i0}^k, \tag{5}$$

$$x_i^k = f^k(z_i^k). \tag{6}$$

Various choices for the activation function f^k are possible; the hyperbolic tangent function, $f^k(x) = \tanh(x)\ \forall k$, has been adopted in this paper.

For anomaly detection, the MLPNN is trained by setting a set of N input vectors, each of dimension ℓ, and a specified target output vector τ of dimension q, which is set to be the zero vector. This implies that the input layer has ℓ neurons and the output layer has q neurons. If the time series data is obtained from an ergodic process, then a data set of length $N\ell$ can be segmented into N vectors of length ℓ to construct the input pattern matrix $\mathscr{P} \in \mathbb{R}^{\ell \times N}$ that is obtained from the N input vectors as

$$\mathscr{P} \equiv [p^1\ p^2\ \cdots\ p^N], \tag{7}$$

where $p^k \equiv [y_{(k-1)\ell+1}\ y_{(k-1)\ell+2}\ \cdots\ y_{k\ell}]^{\mathrm{T}}$; and each y_k is a sample from the ensemble of the time series data. The corresponding output matrix $\mathscr{O}$ is the output of the trained MLPNN (under the nominal condition) under the input pattern $\mathscr{P}$.

$$\mathscr{O} \equiv [o^1\ o^2\ \cdots\ o^N], \tag{8}$$

where $o^i \in \mathbb{R}^q$ is the output of the trained MLPNN under the input $p^k \in \mathbb{R}^\ell$. The performance vector $v \in \mathbb{R}^q$ is obtained as the average of the N outputs.

$$v \equiv \frac{1}{N} \sum_{k=1}^{N} o^k. \tag{9}$$

The time series data under the nominal condition generates the input pattern matrix $\mathscr{P}_{\text{nom}}$. Each column of $\mathscr{P}_{\text{nom}}$ is used to train the MLPNN with respect to the given target output vector τ that is set to be the zero vector. The output of the trained MLPNN is $\mathscr{O}_{\text{nom}}$ with $\mathscr{P}_{\text{nom}}$ as the input; and the resulting performance vector is v_{nom}. Subsequently, input pattern matrices $\{\mathscr{P}_1, \mathscr{P}_2, \ldots\}$ are obtained at slow-time epochs $\{t_1, t_2, \ldots\}$ and corresponding output matrices of the trained MLPNN are $\{\mathscr{O}_1, \mathscr{O}_2, \ldots\}$, which yield the respective performance vectors $\{v_1, v_2, \ldots\}$. The anomaly measures at slow-time epochs $\{t_1, t_2, \ldots\}$ are obtained as

$$\mathscr{M}_k \equiv d(v_k, v_{\text{nom}}), \tag{10}$$

where the $d(\bullet, \bullet)$ is an appropriately defined distance function.

2.3.2. RBF for anomaly detection

The radial basis function [6] in RBFNN is introduced as

$$f(y) = \exp\left(-\frac{\sum_k |y_k - \mu|^\alpha}{N\theta_\alpha}\right), \tag{11}$$

where the exponent parameter $\alpha \in (0, \infty)$; and μ and θ_α are the center and αth central moment of the data set, respectively. For $\alpha = 2, f(\bullet)$ becomes Gaussian, which is the typical radial basis function used in the neural network literature. To perform anomaly detection, the first task is to obtain the sampled time series data when the dynamical system is in the nominal condition and then the mean μ and the central moment θ_α are calculated as

$$\mu = \frac{1}{N}\sum_{k=1}^{N} y_k \quad \text{and} \quad \theta_\alpha = \frac{1}{N}\sum_{k=1}^{N} |y_k - \mu|^\alpha. \tag{12}$$

The distance between any vector y and the center μ is obtained as $d(y, \mu) \equiv (\sum_n |y(n) - \mu|^\alpha)^{1/\alpha}$. Following Eq. (11), the radial basis function at the nominal condition is: $f_{\text{nom}} = f(y)$. Under all conditions including anomalous ones, the parameters μ and θ are kept fixed. However, at slow time epochs $\{t_1, t_2, \ldots\}$, the radial basis functions $\{f_1, f_2, \ldots\}$ are evaluated from the data sets under the (possibly anomalous) conditions. The anomaly measure at the epoch t_k in the slow time scale is obtained as a distance from the nominal condition and is given by

$$\mathscr{M}_k = d(f_{\text{nom}}, f_k), \tag{13}$$

where the $d(\bullet, \bullet)$ is an appropriately defined distance function.

3. Symbolic time series analysis

The concept of *symbolic time series analysis* is built upon phase-space partitioning for encoding nonlinear system dynamics from observed time series data, followed by construction of a finite-state machine model from a symbol sequence under the nominal condition. These issues have been described in detail in the previous paper [1].

This paper has adopted two alternative partitioning approaches for construction of symbol sequences from time series data. The first one is the symbolic false nearest neighbors (SFNN) approach [9] that optimizes the generating partition by avoiding topological degeneracies. The criterion is that short sequences of consecutive symbols ought to localize the corresponding state space point as closely as possible. This is achieved by forming a particular geometrical embedding of the symbolic sequence under the candidate partition and minimizing the apparent errors in localizing state space points. In a good partition, nearby points in the embedding remain close when mapped back into the state space. In contrast, bad partitions induce topological degeneracies where symbolic words map back to globally distinct regions of state space. The *nearest neighbor* to each point in the embedding is found in terms of Euclidean distance of symbolic neighbors and better partitions yield a smaller proportion of *symbolic false nearest neighbors*. For convenience of implementation, the partitions are parameterized with a relatively small number of free parameters. This is accomplished by defining the partitions with respect to a set of radial-basis “influence” functions. The statistic for symbolic false nearest neighbors is minimized over the free parameters using “differential evolution”, which is a genetic algorithm suitable for continuous parameter spaces [9].

The second approach [1] used in this paper is called the wavelet space (WS) method that is built upon time-frequency analysis of the time series data to generate the symbol sequence. A proper choice of scales and the mother wavelet is important in this approach. The large scales represent averaging effects, while the small scales show the details. The power spectrum of the time-localized signal is first analyzed to approximately identify the locally dominant frequencies in the signal. This is followed by generation of wavelet coefficients at the set of scales corresponding to these frequencies. The objective is to extract relevant information in the particular frequency spectrum. The mother wavelet needs to be selected based on the dynamical behavior of the specific application. (Note that the choice of mother wavelet is an open research issue in wavelet literature.)

The graphs of wavelet coefficients versus scale at selected time shifts are stacked starting with the smallest value of scale and ending with its largest value and then back from the largest value to the smallest value of the scale at the next instant of time shift. Then, the wavelet space is partitioned into segments of coefficients on the ordinate separated by horizontal lines. The number of segments in a partition is equal to the size of the alphabet and each segment is associated with a symbol in the alphabet.

3.1. The D-Markov machine

The finite-state machine is constructed as a Dth order Markov chain, called the D-Markov machine [1], for identifying patterns based on time series analysis of the observed data. The core concept of the D-Markov machine is succinctly presented below.

Let the symbolic representation of a discrete-time, discrete-valued stochastic process be denoted by: $\mathbb{S} \equiv \cdots S_{-2}S_{-1}S_0S_1S_2\cdots$. At any instant t, this sequence of random variables can be split into a sequence $\overleftarrow{S}_t$ of the past and a sequence $\overrightarrow{S}_t$ of the future. Assuming conditional stationarity of the symbolic process $\mathbb{S}$ (i.e., $P[\overrightarrow{S}_t|\overleftarrow{S}_t = \overleftarrow{s}]$ being independent of t), the subscript t can be dropped to denote the past and future sequences as $\overleftarrow{S}$ and $\overrightarrow{S}$, respectively. A symbol string, made of the first L symbols of $\overrightarrow{S}$, is denoted by $\overrightarrow{S}^L$. Similarly, a symbol string, made of the last L symbols of $\overleftarrow{S}$, is denoted by $\overleftarrow{S}^L$.

For $D \in \mathbb{N}$, the set of positive integers, a stochastic symbolic stationary process is called Dth order Markov process if the probability of the next symbol depends only on the previous D symbols, i.e. the following condition holds:

$$P(s_i|s_{i-1}s_{i-2}\cdots) = P(s_i|s_{i-1}\cdots s_{i-D}). \tag{14}$$

Alternatively, symbol strings $\overleftarrow{S}, \overleftarrow{S}' \in \overleftarrow{\mathbf{S}}$ become indistinguishable whenever the respective sub-strings $\overleftarrow{S}^D$ and $\overleftarrow{S}'^D$, made of the most recent D symbols, are identical. Thus, a set $\{\overleftarrow{S}^L : L \geqslant D\}$ of symbol strings can be partitioned into a maximum of $|\mathscr{A}|^D$ equivalence classes [1], where $\mathscr{A}$ is the symbol alphabet. Each symbol string in $\{\overleftarrow{S}^L : L \geqslant D\}$, derived from a stationary process, belongs to exactly one of the $|\mathscr{A}|^D$ equivalence classes. Given $D \in \mathbb{N}$ and a symbol string $\overleftarrow{s}$ with $|\overleftarrow{s}| = D$, the *effective* state $q(D, \overleftarrow{s})$ is the equivalence class of symbol strings as defined below:

$$q(D, \overleftarrow{s}) = \{\overleftarrow{S} \in \overleftarrow{\mathbf{S}} : \overleftarrow{S}^D = \overleftarrow{s}\} \tag{15}$$

and the set $\mathbf{Q}(D)$ of *effective* states of the symbolic process is the collection of all such equivalence classes. That is,

$$\mathbf{Q}(D) = \{q(D, \overleftarrow{s}) : \overleftarrow{s} \in \overleftarrow{\mathbf{S}}^D\} \tag{16}$$

and hence $|\mathbf{Q}(D)| = |\mathscr{A}|^D$. A random variable for a state in the above set $\mathbf{Q}$ of states is denoted by $\mathscr{Q}$ and the jth state as q_j. The probability of transitions from state q_j to state q_k is defined as

$$\pi_{jk} = P(s \in \overrightarrow{S}^1 \mid q_j \in \mathbf{Q},\ (s, q_j) \to q_k);$$
$$\sum_k \pi_{jk} = 1. \tag{17}$$

Given an initial state and the next symbol from the original process, only certain successor states are accessible. This is represented as the allowed state transitions resulting from a single symbol. Note that $\pi_{ij} = 0$ if $s_2 s_3 \cdots s_D \neq s'_1 \cdots s'_{D-1}$ whenever $q_i \equiv s_1 s_2 \cdots s_D$ and $q_j \equiv s'_1 s'_2 \cdots s'_D$. Thus, for a D-Markov machine, the stochastic matrix $\Pi \equiv [\pi_{ij}]$ becomes a branded matrix with at most $|\mathscr{A}|^{D+1}$ non-zero entries. The left eigenvector $\mathbf{p}$ corresponding to the unit eigenvalue of Π is the state probability vector under the (fast time scale) stationary condition of the dynamical system. Since Π is an irreducible stochastic matrix under a stationary condition, there exists a unique unit eigenvalue by the Perron–Frobenius theorem [10].

The construction of a D-Markov machine is fairly straightforward. Given $D \in \mathbb{N}$, the states are as defined in Eqs. (15) and (16). On a given symbol sequence $\mathscr{S}$, a window of length $(D+1)$ is slided by keeping a count of occurrences of sequences $s_{i_1} \cdots s_{i_D} s_{i_{D+1}}$ and $s_{i_1} \cdots s_{i_D}$ which are respectively, denoted by $N(s_{i_1} \cdots s_{i_D} s_{i_{D+1}})$ and $N(s_{i_1} \cdots s_{i_D})$. Note that if $N(s_{i_1} \cdots s_{i_D}) = 0$, then the state $q \equiv s_{i_1} \cdots s_{i_D} \in \mathbf{Q}$ has zero probability of occurrence. For $N(s_{i_1} \cdots s_{i_D}) \neq 0$, the transitions probabilities are then obtained by these frequency counts as follows:

$$\pi_{jk} \equiv P[q_k | q_j] = \frac{P[q_k, q_j]}{P[q_j]} = \frac{P(s_{i_1} \cdots s_{i_D} s)}{P(s_{i_1} \cdots s_{i_D})}$$

$$\Rightarrow \pi_{jk} \approx \frac{N(s_{i_1} \cdots s_{i_D} s)}{N(s_{i_1} \cdots s_{i_D})}, \tag{18}$$

where the corresponding states are denoted by: $q_j \equiv s_{i_1} s_{i_2} \cdots s_{i_D}$ and $q_k \equiv s_{i_2} \cdots s_{i_D} s$.

The time series data under the nominal condition generates the state transition matrix Π_{nom} that, in turn, is used to obtain the stationary probability vector $\mathbf{p}_{\text{nom}}$. Subsequently, probability vectors $\{\mathbf{p}_1, \mathbf{p}_2, \ldots\}$ are obtained at slow-time epochs $\{t_1, t_2, \ldots\}$ based on the respective time series data. The anomaly measures at slow-time epochs $\{t_1, t_2, \ldots\}$ are obtained as

$$\mathscr{M}_k \equiv d(\mathbf{p}_k, \mathbf{p}_{\text{nom}}), \tag{19}$$

where the $d(\bullet, \bullet)$ is an appropriately defined distance function.

4. Application to a nonlinear electronic system

This section uses the application example of an electronic system, presented in [1], to make a comparative assessment of different pattern recognition techniques for anomaly detection. Simulation results were used to plan the experiments and interpret the experimental observations. The nonlinear electronic circuit implements a second-order non-autonomous, forced Duffing equation, represented as

$$\frac{\mathrm{d}^2 y(t)}{\mathrm{d}t^2} + \beta(t_s) \frac{\mathrm{d}y(t)}{\mathrm{d}t} + y(t) + y^3(t) = A \cos \omega t. \tag{20}$$

The dissipation parameter $\beta(t_s)$, realized in form of a resistance in the circuit, varies with the slow time t_s and is treated as a constant in the fast time scale at which the dynamical system is excited. Although the system dynamics is represented by a low-order differential equation, it exhibits chaotic behavior that is sufficiently complex from thermodynamic perspectives [3] and is adequate for illustration of the anomaly detection concept. The goal here is to detect, at an early stage, changes in $\beta(t_s)$ which are associated with the anomaly.

Setting the stimulus with amplitude $A = 22.0$ and $\omega = 5.0$ rad/s, the stationary behavior of the system response for this input stimulus is obtained for several values of β in the range of 0.10–0.35. Changes in the stationary behavior take place starting from $\beta \approx 0.15$ with significant changes occurring in the narrow range of $0.27 < \beta < 0.29$. The four plates in Fig. 1 exhibit four phase-plane plots for the values of the parameter β at 0.10 (nominal condition), 0.27, 0.28, and 0.29, respectively. This observation reveals that the stimulus at the excitation frequency of $\omega = 5.0$ rad/s can be effectively used for detection of small changes in β in the range of $0.15 < \beta < 0.35$.

The following anomaly detection approaches are investigated by using the same set of time series data generated from the above experiment with the nominal condition being at $\beta(t_s) = 0.10$:

- Principal component analysis (PCA)
- Multilayer perceptron neural network (MLPNN)

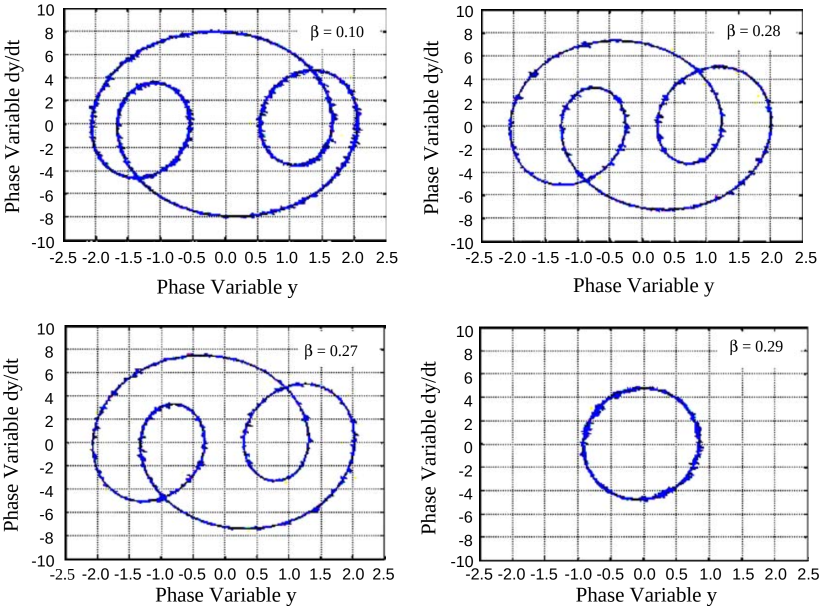

Fig. 1. Phase plots for electronic circuit experiment.

- Radial basis function neural network (RBFNN)
- *D*-Markov machine with symbolic false nearest neighbors (SFNN) partitioning
- *D*-Markov machine with wavelet space (WS) partitioning

The next four paragraphs describe how anomaly measures are calculated based on the above five techniques of pattern recognition. The fourth paragraph addresses both SFNN and WS methods of partitioning in the *D*-Markov method.

Following the principal component analysis (PCA) procedure described in Section 2.2, a block of sampled time series data, having length $\ell = 2700$, is divided into $d = 270$ segments, each of which is of length $n = 10$; these segments are arranged to form a 270×10 data matrix. The resulting 10×10 (symmetric positive-definite) covariance matrix of the data matrix yields a monotonically decreasing set of eigenvalues, $\lambda_1 \ldots \lambda_{10}$, and the associated orthonormal eigenvectors $v^1, \ldots, v^{10}$. At the nominal condition $\beta = 0.10$, the first two eigenvalues are dominant (i.e., $q = 2$) for a threshold of $\eta = 5.0 \times 10^{-5}$ such that

$$\frac{\sum_{i=3}^{10} \lambda_i}{\sum_{i=1}^{10} \lambda_i} \approx 3.0 \times 10^{-5} < \eta.$$

The matrix $\widetilde{M}_{\mathrm{nom}}$ in Eq. (2) is calculated from the data set at $\beta_{\mathrm{nom}} = 0.10$. Similarly, the matrices $\widetilde{M}_1, \ldots, \widetilde{M}_{13}$ are obtained corresponding to $\beta_1 = 0.15, \ldots, \beta_{13} = 0.35$, respectively. The anomaly measures at different values of β are determined according to Eq. (10) relative to nominal matrix $\widetilde{M}_{\mathrm{nom}}$ with the induced Euclidean norm as the metric.

Following the multilayer perceptron neural network (MLPNN) procedure, described in Section 2.3.1, the resulting pattern matrix $\mathscr{P}_{\mathrm{nom}}$ is made of $N = 200$ columns. Each column, having a length $\ell = 30$, is generated from the time series data at $\beta_{\mathrm{nom}} = 0.10$ to train the MLPNN that is chosen to have a input layer (with 30 neurons), 4 hidden layers (with 50 neurons in layer 1, 40 neurons in layer 2, 30 neurons in layer 3, and 40 neurons in layer 4), and the output layer (with 10 neurons): this structure of the MLPNN yields very good convergence for the data sets under consideration. The target corresponding to each input pattern vector is chosen to be 10×1 zero vector. The MLPNN is trained with the nominal data set at $\beta_{\text{nom}} = 0.10$; the gradient descent back-propagation algorithm has been used for network training with an allowable performance mean-square error of 1.0×10^{-5}. The input pattern matrices, $\mathscr{P}_1, \ldots, \mathscr{P}_{13}$, each of dimension (30×200), are then generated from the anomalous data sets at $\beta_1 = 0.15, \ldots, \beta_{13} = 0.35$, respectively, to excite the trained network. The resulting output matrices of the trained MLPNN are $\mathscr{O}_1, \ldots, \mathscr{O}_{13}$, which yield the respective performance vectors, $v_1, \ldots, v_{13}$. Anomaly measures at different values of β are determined according to Eq. (10).

Following the radial basis function neural network (RBFNN) procedure, described in Section 2.3.2, the length of the sampled time series data is chosen to be $N = 2701$. In contrast to the standard RBFNN, where the exponent α is usually chosen to be 2, it was set to $\alpha = 0.1$ for improved anomaly measure sensitivity. An estimate of the parameters, μ and θ_α, are obtained according to Eq. (12) based on the data under the nominal condition, which yields the requisite radial basis function f_{nom} following Eq. (11). The anomaly measures at different values of β are determined according to Eq. (13) with the induced Euclidean norm as the metric.

Based on the time series data the nominal condition at $\beta_{\text{nom}} = 0.10$, the first step in the D-Markov machine method is to find a partition for symbol sequence generation. The partitioning methods, SFNN and WS, described in Section 3, have been investigated for efficacy of anomaly detection. (The mother wavelet *db*1 has been selected for WS partitioning in this application.) For the given stimulus of this experiment, partitioning of the phase space/wavelet space must remain invariant at all epochs of the slow time scale. The value of $D = 1$ was used for construction of the D-Markov machine for all values of β. Following the procedure, described in Section 3.1, the state machines are constructed and the connection matrix $\Pi \equiv [\pi_{jk}]$ and the state probability vector $\mathbf{p}$ for each set of time series data. The state machines were constructed with the symbol alphabet $\mathscr{A} = \{0, 1, 2, \ldots, 7\}$. The anomaly measures at different values of β are determined according to Eq. (19) with the angle between the vectors as the metric [1].

4.1. Comparison of anomaly detection methods

Fig. 2 shows a comparison of the anomaly measure $\mathscr{M}$ for the five cases, described in previous sections, where the same set of time series data has been used in each case. Each anomaly measure profile in Fig. 2 is normalized to unity with respect to its own peak value. The efficacy of a specific anomaly detection technique is largely determined by its capability for accurate detection of small anomalies as early as possible. From this perspective, the five normalized curves are examined to determine how early a specific method is capable of detecting anomalies. In each case, the reference point of nominal condition is represented by the parameter $\beta = 0.10$. All five plots show gradual increase in the anomaly measure $\mathscr{M}$ for β in the approximate range of 0.10–0.25, followed by an abrupt increase in the anomaly measure in the vicinity of $\beta \approx 0.29$ when a (possible) bifurcation takes place. As β increases further, the anomaly curves remain fairly constant; this is analogous to a phase transition in the thermodynamic sense.

A comparison of the normalized curves in Fig. 2 shows that the D-Markov method, with SFNN

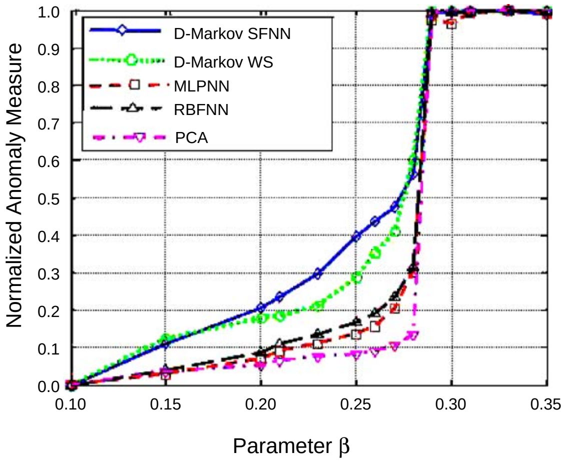

Fig. 2. Performance comparison of anomaly detection techniques.

partitioning and WS partitioning, detects the presence of anomalies much earlier than the PCA, MLPNN and RBFNN techniques. This is evident from the larger slopes of the anomaly measure curves at smaller values of β allows anomaly detection long before the occurrence of actual bifurcation. For WS partitioning, the change in curvature of the anomaly curve in the vicinity of $\beta \approx 0.23$ is an early warning of the forthcoming bifurcation. While all five methods of anomaly detection successfully detects the anomaly in the Duffing system as β increases to the critical value of ≈ 0.29, the performance of the *D*-Markov method with both SFNN and WS partitioning, and specifically WS partitioning, is clearly superior to that of the remaining three pattern recognition techniques from the perspectives of early detection of anomalies.

5. Summary and conclusions

This paper elaborates a novel concept of anomaly detection in complex systems, called the *D*-Markov method [1], that relies on symbolic time series analysis of process variable(s) and is built upon the principles of symbolic dynamics, automata theory, and pattern recognition. The dynamical systems under consideration are assumed to exhibit nonlinear behavior on two time scales, where anomalies may occur on a slow time scale that is several orders of magnitude larger than the fast time scale of the system dynamics. It is also assumed that the dynamical systems in the absence of external stimuli are stationary at the fast time scale and that any non-stationary behavior is observable only on the slow time scale.

The concept of the *D*-Markov method for small change detection in dynamical systems is elucidated on a nonlinear electronic system representing the forced Duffing equation with a slowly varying dissipation parameter. Although the system dynamics in this experiment is represented by a low-order differential equation, it exhibits chaotic behavior that is sufficiently complex for illustration of the anomaly detection concept [3]. The time series data of quasi-stationary phase trajectories are collected to create the respective symbolic dynamics (i.e., strings of symbols) and the probability vector of the finite state automaton is considered as a representation of the phase trajectory's stationary behavior. The distance between any two such vectors under the same stimulus is the measure of anomaly that the system has been subjected to.

The *D*-Markov method is compared with existing techniques of pattern recognition for early detection of anomalies in a nonlinear dynamical system that exhibits complex phenomena such as bifurcation and period doubling; the selected techniques for this comparative evaluation belong to the classes of statistical pattern recognition and neural networks. The statistical pattern recognition method involves principal component analysis (PCA) and the two neural network methods use multilayer perceptron (MLP) and radial basis function (RBF) techniques. The comparison is based on the same sets of time series data generated from the electronic system apparatus. Following conclusions are made on early detection capability.

- The performance of the *D*-Markov methods with SFNN and WS partitioning is significantly superior to that of the remaining three methods.
- The performance of the MLP and RBF neural network methods is better than that of the PCA method.

A major advantage of working with symbols is that the efficiency of numerical computation is significantly enhanced relative to what can be achieved by direct analysis of the original time series data [11]. This is of paramount importance to real-time applications in mobile platforms, where both computational speed and memory requirements of instrumentation and control computers are usually limited. Furthermore, analysis of symbolic data is, on the average, very robust to measurement noise. Often symbolization can be accomplished directly in the instrumentation software to yield inexpensive and relatively simple devices.

The major conclusion based on this limited experimental investigation is that symbolic dynamics along with the stimulus-response methodology and having a vector representation of slowly varying dynamical behavior is effective for early detection of small anomalies.

Further theoretical and experimental research is recommended in the following areas for an application of this method to anomaly detection in operating plants:

- Symbol sequence generation from time series data using the WS partitioning method;
- Development of numerically efficient methods for SFNN partitioning;
- Evaluation of the *D*-Markov machine algorithm relative to other algorithms [1] for automata construction;
- Robustness assessment under noise contamination of the time series data;
- Implementation of anomaly detection techniques for real-time control to mitigate potential failures and extend remaining life without any significant loss of performance.

References

[1] A. Ray, Symbolic dynamic analysis of complex systems for anomaly detection, Signal Process. 84 (7) (2004) 1115–1130.

[2] D. Lind, M. Marcus, A Introduction to Symbolic Dynamics and Coding, Cambridge University Press, Cambridge, 1995.

[3] C. Beck, F. Schlögl, Thermodynamics of Chaotic Systems: An Introduction, Cambridge University Press, Cambridge, NY, 1993.

[4] M. Markou, S. Singh, Novelty detection: a review—parts 1 and 2, Signal Process. 83(12) (2003) 2481–2521.

[5] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley, New York, 2001.

[6] C.M. Bishop, Neural Networks for Pattern Recognition, Oxford University Press Inc., New York, 1995.

[7] D.P. Fukunaga, Statistical Pattern Recognition, second ed., Academic Press, Boston, 1990.

[8] S. Haykin, Neural Networks: A Comprehensive Foundation, Prentice-Hall, Upper Saddle River, NJ, 1999.

[9] M.B. Kennel, M. Buhl, Estimating good discrete partitions form observed data: symbolic false nearest neighbors, Phys. Rev. E 91 (8) (2003) 084102.

[10] R.B. Bapat, T.E.S. Raghavan, Nonnegative Matrices and Applications, Cambridge University Press, Cambridge, 1997.

[11] C.S. Daw, C.E.A. Finney, E.R. Tracy, A review of symbolic analysis of experimental data, Review of Scientific Instruments 74 (2) (2003) 915–930.