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

Signal Processing 84 (2004) 1115–1130

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

Symbolic dynamic analysis of complex systems for anomaly detection☆

Asok Ray*

Mechanical Engineering Department, 137, Reber Building, The Pennsylvania State University, University Park, PA 16802, USA

Received 2 October 2003

Abstract

This paper presents a novel concept of anomaly detection in complex dynamical systems using tools of *Symbolic Dynamics*, *Finite State Automata*, and *Pattern Recognition*, where time-series data of the observed variables on the fast time-scale are analyzed at slow time-scale epochs for early detection of (possible) anomalies. The concept of anomaly detection in dynamical systems is elucidated based on experimental data that have been generated from an active electronic circuit with a slowly varying dissipation parameter.
© 2004 Elsevier B.V. All rights reserved.

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. Early detection of anomalies in complex dynamical systems is essential not only for prevention of cascading catastrophic failures, but also for enhancement of performance and availability [16]. For anomaly detection, it might be necessary to rely on time-series data generated from sensors and other sources of information [1], because accurate and computationally tractable modelling of complex system dynamics is often infeasible solely based on the fundamental principles of physics.

This paper formulates and validates, by laboratory experimentation, a novel concept for detection of slowly evolving anomalies in complex dynamical systems. Often such dynamical systems are either self-excited or can be stimulated with a priori known exogenous inputs to recognize (possible) anomaly patterns from the observed stationary response. Early detection of an anomaly (i.e., small parametric or non-parametric changes) has motivated formulation and validation of the proposed *Symbolic Dynamic* approach to pattern recognition, which is based on the following assumptions:

- The system behavior is stationary at the fast time scale of the process dynamics;

☆ 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.
* Tel.: +1-814-865-6377; fax: +1-814-863-4848.
E-mail address: axr2@psu.edu (A. Ray).

0165-1684/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2004.03.011

- An observable non-stationary behavior of the dynamical system can be associated with anomaly(ies) evolving at a slow time scale.

The theme of anomaly detection, formulated in this paper, is built upon the concepts of *Symbolic Dynamics* [14,15] *Finite State Automata* [12], and *Pattern Recognition* [9] as a means to qualitatively describe the (fast-time-scale) dynamical behavior in terms of symbol sequences [2,4]. Appropriate phase-space partitioning of the dynamical system yields an alphabet to obtain symbol sequences from time-series data [1,8,13]. Then, tools of computational mechanics [7] are used to identify statistical patterns in these symbolic sequences through construction of a (probabilistic) finite-state machine from each symbol sequence. Transition probability matrices of the finite-state machines, obtained from the symbol sequences, capture the pattern of the system behavior by information compression. For anomaly detection, it suffices that a detectable change in the pattern represents a deviation of the nominal behavior from an anomalous one. The state probability vectors, which are derived from the respective state transition matrices under the nominal and an anomalous condition, yield a vector measure of the anomaly, which provides more information than a scalar measure such as the complexity measure [20].

In contrast to the ε-machine [7,20] that has an a priori unknown structure and yields optimal pattern discovery in the sense of mutual information [5,11], the state machine adopted in this paper has an a priori known structure that can be freely chosen. Although the proposed approach is suboptimal, it provides a common state machine structure where physical significance of each state is invariant under changes in the statistical patterns of symbol sequences. This feature allows unambiguous detection of possible anomalies from symbol sequences at different (slow-time) epochs. The proposed approach is apparently computationally faster than the ε-machine [20], because of significantly fewer number of floating point arithmetic operations. These are the motivating factors for introducing this new anomaly detection concept that is based on a fixed-structure fixed-order Markov chain, called the D-Markov machine in the sequel.

The anomaly detection problem is separated into two parts [21]: (i) *forward problem of Pattern Discovery* to identify variations in the anomalous behavior patterns, compared to those of the nominal behavior; and (ii) *inverse problem of Pattern Recognition* to infer parametric or non-parametric changes based on the learnt patterns and observed stationary response. The inverse problem could be ill-posed or have no unique solution. That is, it may not always be possible to identify a unique anomaly pattern based on the observed behavior of the dynamical system. Nevertheless, the feasible range of parameter variation estimates can be narrowed down from the intersection of the information generated from inverse images of the responses under several stimuli.

It is envisioned that complex dynamical systems will acquire the ability of *self-diagnostics* through usage of the proposed anomaly detection technique that is analogous to the diagnostic procedure employed in medical practice in the following sense. Similar to the notion of injecting *medication* or *inoculation* on a nominally healthy patient, a dynamical system would be excited with known stimuli (chosen in the forward problem) in the idle cycles for self diagnosis and health monitoring. The inferred information on health status can then be used for the purpose of self-healing control or life-extending control [25]. This paper focuses on the forward problem and demonstrates the efficacy of anomaly detection based on experimental data generated from an active electronic circuit with a slowly varying dissipation parameter.

The paper is organized in seven sections and two appendices. Section 2 briefly introduces the notion of nonlinear time-series analysis. Section 3 provides a brief overview of symbolic dynamics and encoding of time series data. Section 4 presents two ensemble approaches for statistical pattern representation. It also presents information extraction based on the ε-machine [7] and the D-Markov machine, as well as their comparison from different perspectives. Section 5 presents the notion of anomaly measure to quantify the changing patterns of anomalous behavior of the dynamical system form the information-theoretic perspectives, followed by an outline of the anomaly detection procedure. Section 6 presents experimental results on a nonlinear active electronic circuit to demonstrate efficacy of the proposed anomaly detection technique. Section 7 summarizes and concludes the paper with recommendations for future research. Appendix A explains the physical significance of information-theoretic

quantities used in the Section 4.1 and Section 5. Appendix B introduces the concept of shift spaces, which is used to delineate the differences between the ε-machine [7] and the D-Markov machine in Section 4.4.

2. Nonlinear time-series analysis

This section presents nonlinear time-series analysis (NTSA) that is needed to extract relevant physical information on the dynamical system from the observed data. NTSA techniques are usually executed in the following steps [1]:

1. *Signal separation*: The (deterministic) time-dependent signal $\{y(n) : n \in \mathbb{N}\}$, where $\mathbb{N}$ is the set of positive integers, is separated from noise, using time-frequency and other types of analysis.

2. *Phase space reconstruction*: Based on the Takens Embedding theorem [22], time lagged or delayed variables are used to construct the state vector $\mathbf{x}(n)$ in a phase space of dimension d_E (which is diffeomorphically equivalent to the attractor of the original dynamical system) as follows:

$$\mathbf{x}(n) = [y(n), y(n+T), \ldots, y(n+(d_E-1)T)], \quad (1)$$

where the time lag T is determined using *mutual information*; and one of the ways to determine d_E is the *false nearest neighbors test* [1].

3. *Signal classification*: Signal classification and system identification in nonlinear chaotic systems require a set of invariants for each subsystem of interest followed by comparison of observations with those in the library of invariants. The invariants are properties of the attractor and could be independent of any particular trajectory. These invariants can be divided into two classes: *fractal dimensions* and *Lyapunov exponents*. Fractal dimensions characterize geometrical complexity of dynamics (e.g., spatial distribution of points along a system orbit); and Lyapunov exponents describe the dynamical complexity (e.g., stretching and folding of an orbit in the phase space) [18].

4. *Modeling and prediction*: This step involves determination of the parameters of the assumed model of the dynamics, which is consistent with the invariant classifiers (e.g., Lyapunov exponents, and fractal dimensions).

The first three steps show how chaotic systems may be separated from stochastic ones and, at the same time, provide estimates of the degrees of freedom and the complexity of the underlying dynamical system. Based on this information, Step 4 formulates a state-space model that can be used for prediction of anomalies and incipient failures. The functional form often used in this step, includes orthogonal polynomials and radial basis functions. This paper has adopted an alternative class of discrete models inspired from *Automata Theory*, which is built upon the principles of *Symbolic Dynamics* as described in the following section.

3. Symbolic dynamics and encoding

This section introduces the concept of *Symbolic Dynamics* and its usage for encoding nonlinear system dynamics from observed time-series data. Let a continuously varying physical process be modelled as a finite-dimensional dynamical system in the setting of an initial value problem:

$$\frac{\mathrm{d}\mathbf{x}(t)}{\mathrm{d}t} = f(\mathbf{x}(t), \theta(t_s)); \quad \mathbf{x}(0) = \mathbf{x}_0, \quad (2)$$

where $t \in [0, \infty)$ denotes the (fast-scale) time; $\mathbf{x} \in \mathbb{R}^n$ is the state vector in the phase space; and $\boldsymbol{\theta} \in \mathbb{R}^\ell$ is the (possibly anomalous) parameter vector varying in (slow-scale) time t_s. Sole usage of the model in Eq. (2) may not always be feasible due to unknown parametric and non-parametric uncertainties and noise. A convenient way of learning the dynamical behavior is to rely on the additional information provided by (sensor-based) time-series data [1,4].

A tool for behavior description of nonlinear dynamical systems is based on the concept of formal languages for transitions from smooth dynamics to a discrete symbolic description [2]. The phase space of the dynamical system in Eq. (2) is partitioned into a finite number of cells, so as to obtain a coordinate grid of the space. A compact (i.e., closed and bounded) region $\Omega \in \mathbb{R}^n$, within which the (stationary) motion under the specific exogenous stimulus is circumscribed, is identified. Encoding of Ω is accomplished by introducing a partition $\mathbb{B} \equiv \{B_0, \ldots, B_{m-1}\}$ consisting of m mutually exclusive (i.e., $B_j \cap B_k = \emptyset \ \forall j \neq k$), and exhaustive (i.e., $\bigcup_{j=0}^{m-1} B_j = \Omega$) cells. The dynamical system describes an orbit by the time-series data as:

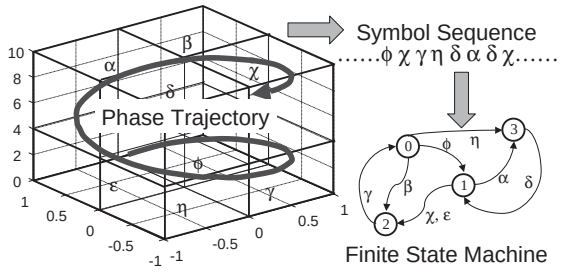

Fig. 1. Continuous dynamics to symbolic dynamics.

$\mathbb{O} \equiv \{x_0, x_1 \cdots, x_k \cdots\}, x_i \in \Omega$, which passes through or touches the cells of the partition $\mathbb{B}$.

Let us denote the cell visited by the trajectory at a time instant as a random variable S that takes a symbol value $s \in \mathscr{A}$. The set $\mathscr{A}$ of m distinct symbols that label the partition elements is called the *symbol alphabet*. Each initial state $x_0 \in \Omega$ generates a sequence of symbols defined by a mapping from the phase space into the symbol space as:

$$x_0 \longrightarrow s_{i0} s_{i1} s_{i2} \cdots s_{ik} \cdots . \tag{3}$$

The mapping in Eq. (3) is called *Symbolic Dynamics* as it attributes a legal (i.e., physically admissible) symbol sequence to the system dynamics starting from an initial state. (Note: A symbol alphabet $\mathscr{A}$ is called a generating partition of the phase space Ω if every legal symbol sequence uniquely determines a specific initial condition x_0, i.e., every symbolic orbit uniquely identifies one continuous space orbit.) Fig. 1 pictorially elucidates the concepts of partitioning a finite region of the phase space and mapping from the partitioned space into the symbol alphabet. This represents a spatial and temporal discretization of the system dynamics defined by the trajectories. Fig. 1 also shows conversion of the symbol sequence into a finite-state machine as explained in later sections.

Symbolic dynamics can be viewed as coarse graining of the phase space, which is subjected to (possible) loss of information resulting from granular imprecision of partitioning boxes, measurement noise and errors, and sensitivity to initial conditions. However, the essential robust features (e.g., periodicity and chaotic behavior of an orbit) are expected to be preserved in the symbol sequences through an appropriate partitioning of the phase space [2]. Although the theory of phase-space partitioning is well developed for one-dimensional mappings, very few results are known for two- and higher-dimensional systems [4].

4. Pattern identification

Given the intricacy of phase trajectories in complex dynamical systems, the challenge is to identify their *patterns* in an appropriate category by using one of the following two alternative approaches:

- The single-item approach, which relies on Kolmogorov Chiatin (KC) complexity, also known as algorithmic complexity [5], for exact pattern regeneration;
- The ensemble approach, which regards the pattern as one of many possible experimental outcomes, for estimated pattern regeneration.

While the single-item approach is common in coding theory and computer science, the ensemble approach has been adopted in this paper due to its physical and statistical relevance. As some of the legal symbol sequences may occur more frequently than others, a probability is attributed to each observed sequence. The collection of all legal symbol sequences $S_{-M} \cdots S_{-2} S_{-1} S_0 S_1 \cdots S_N$, $N, M = 0, 1, 2 \ldots$, defines a stochastic process that is a symbolic probabilistic description of the continuous system dynamics.

Let us symbolically denote a discrete-time, discrete-valued stochastic process as

$$\mathbb{S} \equiv \cdots, S_{-2} S_{-1} S_0 S_1 S_2 \cdots, \tag{4}$$

where each random variable S_i takes exactly one value in the (finite) alphabet $\mathscr{A}$ of m symbols (see Section 3). The symbolic stochastic process $\mathbb{S}$ is dependent on the specific partitioning of the phase space and is non-Markovian, in general. Even if a partitioning that makes the stochastic process a Markov chain exists, identification of such a partitioning is not always feasible because the individual cells may have fractal boundaries instead of being simple geometrical objects. In essence, there is a trade-off between selecting a simple partitioning leading to a complicated stochastic process, and a complicated partitioning leading to a simple stochastic process. Recent literature has reported a comprehensive numerical procedure for construction phase-space partitions from

the time-series data [13]. Having defined a partition of the phase space, the time-series data is converted to a symbol sequence that, in turn, is used for construction of a finite-state machine using the tools of Computational Mechanics [7] as illustrated in Fig. 1.

This paper considers two alternative techniques of finite-state machine construction from a given symbol sequence $\mathscr{S}$: (i) the ε-machine formulation [20]; and (ii) a new concept based on Dth order Markov chains, called the D-Markov machine, for identifying patterns based on time series analysis of the observed data. Both techniques rely on information-theoretic principles (see Appendix A) and are based on computational mechanics [7].

4.1. The ε-machine

Like statistical mechanics [10,4], computational mechanics is concerned with dynamical systems consisting of many partially correlated components. Whereas Statistical Mechanics deals with the local space–time behavior and interactions of the system elements, computational mechanics relies on the joint probability distribution of the phase-space trajectories of a dynamical system. The ε-machine construction [7,20] makes use of the joint probability distribution to infer the information processing being performed by the dynamical system. This is developed using the statistical mechanics of orbit ensembles, rather than focusing on the computational complexity of individual orbits.

Let the symbolic representation of a discrete-time, discrete-valued stochastic process be denoted by: $\mathbb{S} \equiv \cdots S_{-2}S_{-1}S_0S_1S_2 \cdots$ as defined earlier in Section 4. 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., $P[\overleftarrow{S}_t | \overrightarrow{S}_t = \overrightarrow{s}]$ 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 $\overleftarrow{S}^L$.

Prediction of the future $\overrightarrow{S}$ necessitates determination of its probability conditioned on the past $\overleftarrow{S}$, which requires existence of a function ε mapping histories $\overleftarrow{s}$ to predictions $P(\overrightarrow{S} | \overleftarrow{s})$. In essence, a prediction imposes a partition on the set $\overleftarrow{\mathbf{S}}$ of all histories. The cells of this partition contain histories for which the same prediction is made and are called the *effective states* of the process under the given predictor. The set of effective states is denoted by $\mathbf{R}$; a random variable for an effective state is denoted by $\mathscr{R}$ and its realization by ρ.

The objective of ε-machine construction is to find a predictor that is an optimal partition of the set $\overleftarrow{\mathbf{S}}$ of histories, which requires invoking two criteria in the theory of Computational Mechanics [6]:

1. *Optimal Prediction*: For any partition of histories or effective states $\mathscr{R}$, the conditional entropy $H[\overrightarrow{S}^L | \mathscr{R}] \geqslant H[\overrightarrow{S}^L | \overleftarrow{S}]$, $\forall L \in \mathbb{N}$, $\forall \overleftarrow{S} \in \overleftarrow{\mathbf{S}}$, is equivalent to remembering the whole past. Effective states $\mathscr{R}$ are called *prescient* if the equality is attained $\forall L \in \mathbb{N}$. Therefore, optimal prediction needs the effective states to be prescient.

2. *Principle of Occam Razor*: The prescient states with the least complexity are selected, where complexity is defined as the measured Shannon information of the effective states:

$$H[\mathscr{R}] = -\sum_{\rho \in \mathbf{R}} P(\mathscr{R} = \rho) \log P(\mathscr{R} = \rho). \tag{5}$$

Eq. (5) measures the amount of past information needed for future prediction and is known as *Statistical Complexity* denoted by $C_\mu(\mathscr{R})$ (see Appendix A).

For each symbolic process $\mathbb{S}$, there is a unique set of prescient states known as *causal states* that minimize the statistical complexity $C_\mu(\mathscr{R})$.

Definition 4.1 (Shalizi et al. [20]). Let $\mathbb{S}$ be a (conditionally) stationary symbolic process and $\overleftarrow{\mathbf{S}}$ be the set of histories. Let a mapping $\varepsilon : \overleftarrow{\mathbf{S}} \to \Upsilon(\overrightarrow{\mathbf{S}})$ from the set $\overleftarrow{\mathbf{S}}$ of histories into a collection $\Upsilon(\overrightarrow{\mathbf{S}})$ of measurable subsets of $\overleftarrow{\mathbf{S}}$ be defined as:

$$\forall \Gamma \in \Upsilon(\overrightarrow{\mathbf{S}}), \quad \varepsilon(\overleftarrow{s}) \equiv \{\overleftarrow{s}' \in \overleftarrow{\boldsymbol{S}} \text{ such that}$$

$$P(\overrightarrow{S} \in \Gamma | \overleftarrow{S} = \overleftarrow{s}) = P(\overrightarrow{S} \in \Gamma | \overleftarrow{S} = \overleftarrow{s}')\}. \tag{6}$$

Then, the members of the range of the function ε are called the causal states of the symbolic process $\mathbb{S}$. The ith causal state is denoted by q_i and the set of all causal states by $\mathbf{Q} \subseteq \Upsilon(\overrightarrow{\mathbf{S}})$. The random variable

corresponding to a causal state is denoted by $\mathscr{Q}$ and its realization by q.

Given an initial causal state and the next symbol from the symbolic process, only successor causal states are possible. This is represented by the legal transitions among the causal states, and the probabilities of these transitions. Specifically, the probability of transition from state q_i to state q_j on a single symbol s is expressed as:

$$T_{ij}^{(s)} = P(\overrightarrow{S}^1 = s, \mathscr{Q}' = q_j \mid \mathscr{Q} = q_i) \ \forall q_i, q_j \in \mathbf{Q}, \quad (7)$$

$$\sum_{s\in\mathscr{A}} \sum_{q_j\in\mathbf{Q}} T_{ij}^{(s)} = 1. \quad (8)$$

The combination of causal states and transitions is called the ε-*machine* (also known as *the causal state model* [20]) of a given symbolic process. Thus, the ε-machine represents the way in which the symbolic process stores and transforms information. It also provides a description of the pattern or regularities in the process, in the sense that the pattern is an algebraic structure determined by the causal states and their transitions. The set of labelled transition probabilities can be used to obtain a stochastic matrix [3] given by: $\mathscr{T} = \sum_{s\in\mathscr{A}} \mathscr{T}^s$ where the square matrix $\mathscr{T}^s$ is defined as: $\mathscr{T}^s = [T_{ij}^s] \ \forall s \in \mathscr{A}$. Denoting $\mathbf{p}$ as the left eigenvector of $\mathscr{T}$, corresponding to the eigenvalue $\lambda = 1$, the probability of being in a particular causal state can be obtained by normalizing $\|\mathbf{p}\|_{\ell_1} = 1$. A procedure for construction of the ε-machine is outlined below.

The original ε-machine construction algorithm is the subtree-merging algorithm as introduced in [7,6]. The default assumption of this technique was employed by Surana et al. [21] for anomaly detection. This approach has several shortcomings, such as lack of a systematic procedure for choosing the algorithm parameters, may return non-deterministic causal states, and also suffers from slow convergence rates. Recently, Shalizi et al. [20] have developed a code known as Causal State Splitting Reconstruction (CSSR) that is based on state splitting instead of state merging as was done in the earlier algorithm of subtree-merging [7]. The CSSR algorithm starts with a simple model for the symbolic process and elaborates the model components only when statistically justified. Initially, the algorithm assumes the process to be independent and identically distributed (iid) that can be represented by a single causal state and hence zero statistical complexity and high entropy rate. At this stage, CSSR uses statistical tests to determine when it must add states to the model, which increases the estimated complexity, while lowering the entropy rate h_μ (see Appendix A). A key and distinguishing feature of the CSSR code is that it maintains homogeneity of the causal states and deterministic state-to-state transitions as the model grows. Complexity of the CSSR algorithm is: $O(m^{L_{\max}}) + O(m^{2L_{\max}+1}) + O(N)$, where m is the size of the alphabet $\mathscr{A}$; N is the data size and $L_{\max}$ is the length of the longest history to be considered. Details are given in [20].

4.2. The suboptimal D-markov machine

This section presents a new alternative approach for representing the pattern in a symbolic process, which is motivated from the perspective of anomaly detection. The core assumption here is that the symbolic process can be represented to a desired level of accuracy as a Dth order Markov chain, by appropriately choosing $D \in \mathbb{N}$.

Definition 4.2. A stochastic symbolic stationary process $\mathbb{S} \equiv \cdots S_{-2}S_{-1}S_0S_1S_2\cdots$ is called Dth order Markov process if the probability of the next symbol depends only on the previous (at most) D symbols, i.e. the following condition holds:

$$P(S_i|S_{i-1}S_{i-2}\cdots S_{i-D}\cdots) = P(S_i|S_{i-1}\cdots S_{i-D}) \quad (9)$$

Alternatively, symbol strings $\overleftarrow{S}, \overleftarrow{S'} \in \overleftarrow{\mathbf{S}}$ become indistinguishable whenever the respective substrings $\overleftarrow{S}^D$ and $\overleftarrow{S'}^D$, made of the most recent D symbols, are identical.

Definition (4.2) can be interpreted as follows: $\forall \ \overleftarrow{S}, \ \overleftarrow{S'} \in \overleftarrow{\mathbf{S}}$ such that $|\overleftarrow{S}| \geqslant D$ and $|\overleftarrow{S'}| \geqslant D$, ($\overleftarrow{S'} \in \varepsilon(\overleftarrow{S})$ and $\overleftarrow{S} \in \varepsilon(\overleftarrow{S'})$) iff $\overleftarrow{S}^D = \overleftarrow{S'}^D$. Thus, a set $\{\overleftarrow{S}^L : L \geqslant D\}$ of symbol stings can be partitioned into a maximum of $|\mathscr{A}|^D$ equivalence classes where $\mathscr{A}$ is the symbol alphabet, under the equivalence relation defined in Eq. (6). Each symbol string in $\{\overleftarrow{S}^L : L \geqslant D\}$ either belongs to one of the $|\mathscr{A}|^D$ equivalence classes or has a distinct equivalence class. All such symbol strings belonging to the distinct equivalence class

form transient states, and would not be of concern to anomaly detection for a (fast-time-scale) stationary condition under (slowly changing) anomalies. 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:

$$q(D, \overleftarrow{s}) = \{\overleftarrow{S} \in \overleftarrow{\mathbf{S}}: \overleftarrow{S}^D = \overleftarrow{s}\} \tag{10}$$

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

$$\mathbf{Q}(D) = \{q(D, \overleftarrow{s}): \overleftarrow{s} \in \overleftarrow{\mathbf{S}}^D\} \tag{11}$$

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:

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

$$\sum_k \pi_{jk} = 1; \tag{12}$$

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 $|\mathscr{A}|^{D+1}$ nonzero entries.

The construction of a D-Markov machine is fairly straightforward. Given $D \in \mathbb{N}$, the states are as defined in Eqs. (10) and (11). 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:

$$\pi_{jk} = \frac{P(s_{i_1} \cdots s_{i_D} s)}{P(s_{i_1} \cdots s_{i_D})} \approx \frac{N(s_{i_1} \cdots s_{i_D} s)}{N(s_{i_1} \cdots s_{i_D})}, \tag{13}$$

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

As an example, Fig. 2 shows the finite-state machine and the associated state transition matrix for a D-Markov machine, where the alphabet $\mathscr{A} = \{0, 1\}$,

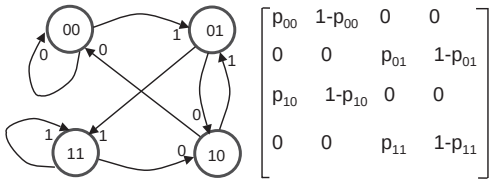

Fig. 2. State machine with $D = 2$, and $|\mathscr{A}| = 2$.

i.e., alphabet size $|\mathscr{A}| = 2$; and the states are chosen as words of length $D = 2$ from a symbol sequence $\mathscr{S}$. Consequently, the total number of states is $|\mathscr{A}|^D = 4$, which is the number of permutations of the alphabet symbols within a word of length D; and the set of states $\mathbf{Q} = \{00, 01, 10, 11\}$. The state transition matrix on the right half of Fig. 2 denotes the probability $\pi_{ij} = p_{ij}$ of occurrence of the symbol $0 \in \mathscr{A}$ at the state $q \equiv ij$, where $i, j \in \mathscr{A}$. The states are joined by edges labelled by a symbol in the alphabet. The state machine moves from one state to another upon occurrence of an event as a new symbol in the symbol sequence is received and the resulting transition matrix has at most $|\mathscr{A}|^{D+1} = 8$ non-zero entries. The machine language is complete in the sense that there are different outgoing edges marked by different symbols; however, it is possible that some of these arcs may have zero probability.

4.3. Statistical mechanical concept of D-Markov machine

This section outlines an analogy between the structural features of the D-Markov machine and those of spin models in Statistical Mechanics. The main idea is derived from the doctoral dissertation of Feldman [10] who has demonstrated how measures of patterns from Information Theory and Computational Mechanics are captured in the construction of ε-Machines. In general, the effects of an anomaly are reflected in the respective state transition matrices. Thus, the structure of the finite-state machine is fixed for a given alphabet size $|\mathscr{A}|$ and window length D. Furthermore, the number of edges is also finite because of the finite alphabet size. The elements of the state transition matrix (that is a stochastic matrix [3]) are identified from the symbol sequence.

For $|\mathscr{A}| = 2$ and $D = 2$, the finite-state machine construction is (to some extent) analogous to the one-dimensional Ising model of spin-1/2 systems with nearest neighbor interactions, where the z-component of each spin takes on one of the two possible values $s = +1$ or $s = -1$ [10,19]. For $|\mathscr{A}| \geqslant 3$, the machine would be analogous to one-dimensional Potts model, where each spin is directed in the z-direction with $|\mathscr{A}|$ different discrete values s_k: $k \in 1, 2, \ldots, |\mathscr{A}|$; for a $j/2$-spin model, the alphabet size $|\mathscr{A}| = j + 1$ [4]. For $D \geqslant 2$, the spin interactions extend up to the $(D-1)$th neighbor.

4.4. Comparison of D-Markov machine and ε-machine

An ε-machine seeks to find the patterns in the time series data in the form of a finite-state machine, whose states are chosen for optimal prediction of the symbolic process; and a finite-state automation can be used as a pattern for prediction [20]. An alternative notion of the pattern is one which can be used to compress the given observation. The first notion of the pattern subsumes the second, because the capability of optimal prediction necessarily leads to the compression as seen in the construction of states by lumping histories together. However, the converse is not true in general. For the purpose of anomaly detection, the second notion of pattern is sufficient because the goal is to represent and detect the deviation of an anomalous behavior from the nominal behavior. This has been the motivating factor for proposing an alternative technique, based on the fixed structure D-Markov machine. It is possible to detect the evolving anomaly, if any, as a change in the probability distribution over the states.

Another distinction between the D-Markov machine and ε-machine can be seen in terms of *finite-type shifts* and *sofic shifts* [15] (see Appendix B). Basic distinction between finite-type shifts and sofic shifts can be characterized in terms of the *memory*: while a finite-type shift has finite-*length* memory, a sofic shift uses finite *amount* of memory in representing the patterns. Hence, finite-type shifts are strictly proper subsets of sofic shifts. While, any finite-type shift has a representation as a graph, sofic shifts can be represented as a *labelled graph*. As a result, the finite-type shift can be considered as an "extreme version" of a D-Markov chain (for an appropriate D) and sofic shifts as an "extreme version" of a Hidden Markov process [24], respectively. The shifts have been referred to as "extreme" in the sense that they specify only a set of allowed sequences of symbols (i.e., symbol sequences that are actually possible, but not the probabilities of these sequences). Note that a Hidden Markov model consists of an internal D-order Markov process that is observed only by a function of its internal-state sequence. This is analogous to sofic shifts that are obtained by a labelling function on the edge of a graph, which otherwise denotes a finite-type shift. Thus, in these terms, an ε-machine infers the Hidden Markov Model (sofic shift) for the observed process. In contrast, the D-Markov Model proposed in this paper infers a (finite-type shift) approximation of the (sofic shift) ε-machine.

5. Anomaly measure and detection

The machines described in Sections 4.1 and 4.2 recognize patterns in the behavior of a dynamical system that undergoes anomalous behavior. In order to quantify changes in the patterns that are representations of evolving anomalies, we induce an *anomaly measure* on these machines, denoted by $\mathscr{M}$. The anomaly measure $\mathscr{M}$ can be constructed based on the following information-theoretic quantities: entropy rate, excess entropy, and complexity measure of a symbol string $\mathscr{S}$ (see Appendix A).

- The entropy rate $h_\mu(\mathscr{S})$ quantifies the intrinsic randomness in the observed dynamical process.
- The excess entropy $\mathbf{E}(\mathscr{S})$ quantifies the memory in the observed process.
- The statistical complexity $\mathscr{C}_\mu(\mathscr{S})$ of the state machine captures the average memory requirements for modelling the complex behavior of a process.

Given two symbol strings $\mathscr{S}$ and $\mathscr{S}_0$, it is possible to obtain a measure of anomaly by adopting any one of the following three alternatives:

$$\mathscr{M}(\mathscr{S}, \mathscr{S}_0)) = \begin{cases} |h_\mu(\mathscr{S}) - h_\mu(\mathscr{S}_0)|, \text{ or} \\ |\mathbf{E}(\mathscr{S}) - \mathbf{E}(\mathscr{S}_0)|, \text{ or} \\ |C_\mu(\mathscr{S}) - C_\mu(\mathscr{S}_0)|. \end{cases}$$

Note that each of the anomaly measures, defined above, is a *pseudo metric* [17]. For example, let us consider two periodic processes with unequal periods, represented by $\mathscr{S}$ and $\mathscr{S}_0$. For both processes, $h_\mu = 0$, so that $\mathscr{M}(\mathscr{S}, \mathscr{S}_0) = 0$ for the first of the above three options, even if $\mathscr{S} \neq \mathscr{S}_0$.

The above measures are obtained through scalar-valued functions defined on a state machine and do not exploit the rich algebraic structure represented in the state machine. For example, the connection matrix $\mathscr{T}$ associated with the ε-machine (see Section 4.1), can be treated as a vector representation of any possible anomalies in the dynamical system. The induced 2-*norm* of the difference between the $\mathscr{T}$-matrices for the two state machines can then be used as a measure of anomaly, i.e., $\mathscr{M}(\mathscr{S}, \mathscr{S}_0) = \|\mathscr{T} - \mathscr{T}_0\|_2$. Such a measure, used in [21], was found to be effective. However, there is some subtlety in using this measure on ε-machines, because ε-machines do not guarantee that the machines formulated from the symbol sequences $\mathscr{S}$ and $\mathscr{S}_0$ have the same number of states; and these states do not necessarily have similar physical significance. In general, $\mathscr{T}$ and $\mathscr{T}_0$ may have different dimensions and different physical significance. However, by encoding the causal states, $\mathscr{T}$ could be embedded in a larger matrix, and an induced norm of the difference between $\mathscr{T}$ matrices for these two machines can be defined. Alternatively, a (vector) measure of anomaly can be derived directly from the stochastic matrix $\mathscr{T}$ as the left eigenvector $\mathbf{p}$ corresponding to the unit eigenvalue of $\mathscr{T}$, which is the state probability vector under a stationary condition.

This paper has adopted the *D*-Markov machine approach, described in the Section 4.2 to build the state machines. Since *D*-Markov machines have a fixed state structure, the state probability vector $\mathbf{p}$ associated with the state machine have been used for a vector representation of anomalies, leading to the anomaly measure $\mathscr{M}(\mathscr{S}, \mathscr{S}_0))$ as a distance function between the respective probability vectors $\mathbf{p}$ and $\mathbf{p}_0$ (that are of identical dimensions), or any other appropriate functional.

5.1. Anomaly detection procedure

Having discussed various tools and techniques, this section outlines the steps of the *forward problem* and the *inverse problem* described in Section 1. Following are the steps for the forward problem:

(F1) Selection of an appropriate set of input stimuli.
(F2) Signal–noise separation, time interval selection, and phase-space construction.
(F3) Choice of a phase space partitioning to generate symbol alphabet and symbol sequences.
(F4) State Machine construction using generated symbol sequence(s) and determining the connection matrix.
(F5) Selection of an appropriate metric for the anomaly measure $\mathscr{M}$.
(F6) Formulation and calibration of a (possibly non-parametric) relation between the computed anomaly measure and known physical anomaly under which the time-series data were collected at different (slow-time) epochs.

Following are the steps for the inverse problem:

(I1) Excitation with known input stimuli selected in the forward problem.
(I2) Generation of the stationary behavior as time-series data for each input stimulus at different (slow-time) epochs.
(I3) Embedding the time-series data in the phase space determined for the corresponding input stimuli in Step F2 of the forward problem.
(I4) Generation of the symbol sequence using the same phase-space partition as in Step F3 of the forward problem.
(I5) State Machine construction using the symbol sequence and determining the anomaly measure.
(I6) Detection and identification of an anomaly, if any, based on the computed anomaly measure and the relation derived in Step F6 of the forward problem.

6. Application to an active electronic circuit

This section illustrates an application of the *D*-Markov machine concept for anomaly detection on an experimental apparatus that consists of an active electronic circuit. The apparatus implements a second-order non-autonomous, forced Duffing equation in real time [23]. The governing equation with a cubic nonlinearity in one of the state variables is

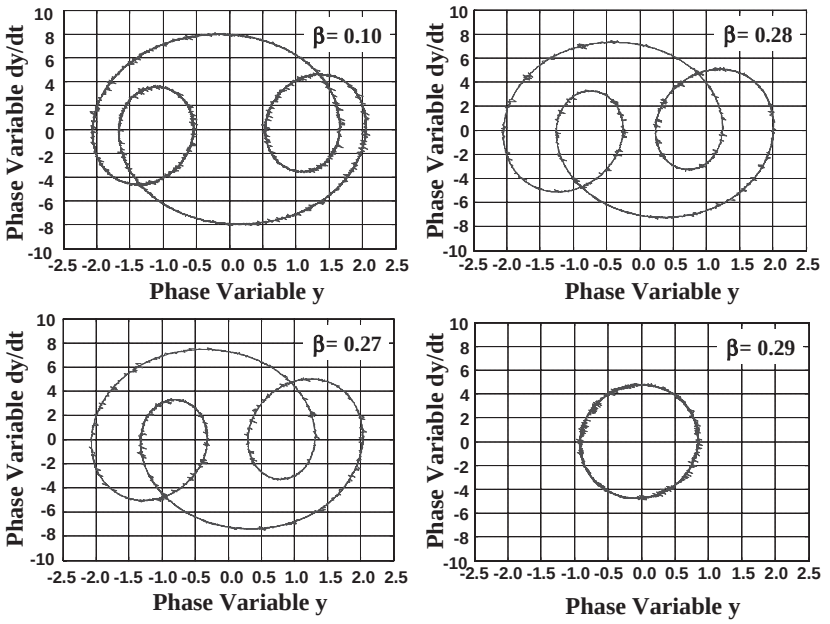

Fig. 3. Phase plots for electronic circuit experiment.

given below:

$$\frac{\mathrm{d}^2x(t)}{\mathrm{d}t^2} + \beta(t_s)\frac{\mathrm{d}x(t)}{\mathrm{d}t} + x(t) + x^3(t) = A\cos\omega t. \quad (14)$$

The dissipation parameter $\beta(t_s)$, realized in the form of a resistance in the circuit, is made to vary in the slow time scale t_s and is treated as a constant in the fast time scale t at which the dynamical system is excited. The goal is to detect, at an early stage, changes in $\beta(t_s)$ that are associated with the anomaly.

In the forward problem, the first task is the selection of appropriate input stimuli. For illustration purposes, we have used the stimulus with amplitude $A = 22.0$ and frequency $\omega = 5.0$ in this paper. Changes in the stationary behavior of the electronic circuit take place starting from $\beta \approx 0.10$ with significant changes occurring in the narrow range of $0.28 < \beta < 0.29$. 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.

The four plates in Fig. 3 exhibit four phase plots for the values of the parameter β at 0.10, 0.27, 0.28, and 0.29, respectively, relating the phase variable of electrical charge that is proportional to the voltage across one of the capacitors in the electronic circuit,

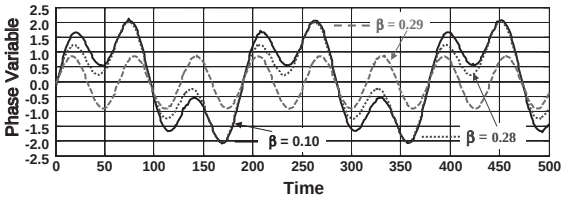

Fig. 4. Time plots for electronic circuit experiment.

and its time derivative (i.e., the instantaneous current). While a small difference between the plots for $\beta = 0.10$ and 0.27 is observed, there is no clearly visible difference between the plots for $\beta = 0.27$ and 0.28 in Fig. 3. However, the phase plots for $\beta = 0.28$ and 0.29 display a very large difference indicating period doubling possibly due to onset of bifurcation. Fig. 4 displays time responses of the stationary behavior of the phase variable for different values of the parameter β corresponding to the phase plots in Fig. 3. The plots in Fig. 4 are phase-aligned for better visibility. (Note that the proposed anomaly detection method does not require phase alignment; equivalently, the finite-state machine Fig. 2 can be started from any arbitrary state corresponding to no specific initial condition.) While the time responses for $\beta = 0.27$ and 0.28 are

indistinguishable, there is a small difference between those for $\beta = 0.27$ and 0.10. Similar to the phase plots in Fig. 3, the time responses for $\beta = 0.28$ and 0.29 display existence of period doubling due to a possible bifurcation.

Additional exogenous stimuli have been identified, which also lead to significant changes in the stationary behavior of the electronic system dynamics for other ranges of β. For example, with the same amplitude $A = 22.0$, stimuli at the excitation frequencies of $\omega = 2.0$ and $\omega = 1.67$ (not shown in Figs. 3 and 4) detect small changes in the ranges of $0.18 < \beta < 0.20$ and $0.11 < \beta < 0.12$, respectively [21]. These observations reveal that exogenous stimuli at different excitation frequencies can be effectively used for detection of small changes in β over an operating range (e.g., $0.10 < \beta < 0.35$).

Having obtained the phase plots from the time-series data, the next step is to find a partition of the phase space for symbol sequence generation. This is a difficult task especially if the time-series data is noise-contaminated. Several methods of phase-space partitioning have been suggested in literature (for example, [1,8,13]). Apparently, there exist no well-established procedure for phase-space partitioning of complex dynamical systems; this is a subject of active research. In this paper, we have introduced a new concept of symbol sequence generation, which uses wavelet transform to convert the time-series data to time-frequency data for generating the symbol sequence. 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. The resulting *scale series* data in the wavelet space is analogous to the time-series data in the phase space. 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 partition is associated with a symbol in the alphabet. For a given stimulus, partitioning of the wavelet space must remain invariant at all epochs of the slow time scale. Nevertheless, for different stimuli, the partitioning could be chosen differently. (The concept of proposed wavelet-space partitioning would require significant theoretical research before its acceptance for application to a general class of dynamical systems for anomaly detection; and its efficacy needs to be compared with that of existing phase-space partitioning methods such as false nearest neighbor partitioning [13].)

The procedure, described in the subsection IV-B constructs a D-Markov machine and obtains the connection matrix $\mathcal{T}$ and the state vector $\mathbf{p}$ from the symbol sequence corresponding to each β. For this analysis, the wave space generated from each data set has been partitioned into eight (8) segments, which makes the alphabet size $|\mathcal{A}| = 8$ to generate symbol sequences from the *scale series* data. At each value of β, the generated symbol sequence has been used to construct several D-Markov Machines starting with $D = 1$ and higher integers. It is observed that, the dominant probabilities of the state vector (albeit having different dimensions) for different values of D are virtually similar. Therefore, a fixed-structure D-Markov Machine with alphabet size $|\mathcal{A}| = 8$ and depth $D = 1$, which yields the number of states $|\mathcal{A}|^D = 8$, is chosen to generate state probability ($\mathbf{p}$) vectors for the symbol sequences.

The electronic circuit system is assumed to be at the nominal condition for the dissipation parameter $\beta = 0.10$, which is selected as the reference point for calculating the anomaly measure. The anomaly measure $\mathcal{M}$ is computed based on two different computation methods as discussed in Section 5. Fig. 5 exhibits three plots of the normalized anomaly measure $\mathcal{M}$ versus the dissipation parameter β, where $\mathcal{M}$ is computed based on different metrics. In each

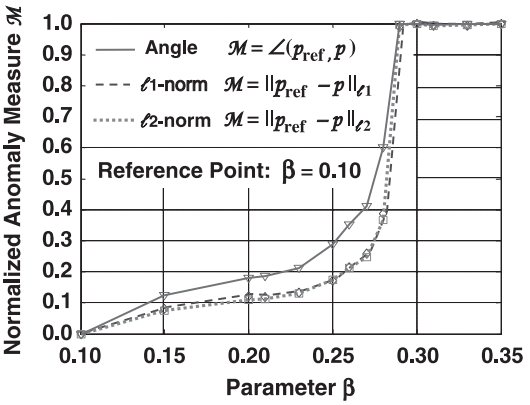

Fig. 5. Anomaly measure versus parameter β.

case, the reference point of nominal condition is represented by the parameter $\beta = 0.10$. The first plot, shown in solid line, shows $\mathscr{M}$ expressed as the angle (in radians) between the $\mathbf{p}$ vectors of the state machines under nominal and anomalous conditions, i.e., $\mathscr{M} = \angle(\mathbf{p}_{\text{ref}}, \mathbf{p}) \equiv \cos^{-1}(\frac{|\langle \mathbf{p}_{\text{ref}}, \mathbf{p} \rangle|}{\|\mathbf{p}_{\text{ref}}\|_{\ell 2} \|\mathbf{p}\|_{\ell 2}})$. The remaining two plots, one in dashed line and the other in dotted line, show the anomaly measure expressed as the ℓ_1-norm and ℓ_2-norm of the difference between the $\mathbf{p}$ vectors of the state machines under nominal and an anomalous conditions, i.e., $\mathscr{M} = \|\mathbf{p}_{\text{ref}} - \mathbf{p}\|_{\ell 1}$ and $\mathscr{M} = \|\mathbf{p}_{\text{ref}} - \mathbf{p}\|_{\ell 2}$, respectively. In each of the three plots, $\mathscr{M}$ is normalized to unity for better comparison. All three plots in Fig. 5 show gradual increase in the anomaly measure $\mathscr{M}$ for β in the approximate range of 0.10–0.25. At $\beta \approx 0.25$ and onwards, $\mathscr{M}$ starts increasing at a faster rate and finally saturates at $\beta \geqslant 0.29$. The large values of anomaly measure at $\beta = 0.29$ and beyond indicate the occurrence of period reduction as seen in Figs. 3 and 4. This abrupt disruption, preceded by gradual changes, is analogous to a phase transition in the thermodynamic sense [4], which can also be interpreted as a catastrophic disruption in a physical process. Hence, observation of modest changes in the anomaly measure may provide very early warnings for a forthcoming catastrophic failure as indicated by the gradual change in the $\beta - \mathscr{M}$ curve.

Following the steps (I1)–(I5) of the inverse problem in Section 5.1, the state probability vector $\mathbf{p}$ can be obtained for the stationary behavior under the known stimulus. The a priori information on the anomaly measure, generated in the step F6 of the forward problem in the Section 5.1, can then be used to determine the possible range in which β lies. Solutions of the forward problem can generate more information on different ranges of β under different input stimuli. Thus, the range of the unknown parameter β can be further narrowed down by repeating this step for other known stimuli as reported earlier [21]. This ensemble of information provides inputs for the inverse problem for detecting anomalies based on the sensor data collected in real time, during the operation of machineries.

7. Summary and conclusions

This paper presents a novel concept of anomaly detection in complex systems based on the tools of Symbolic Dynamics, Finite State Automata, and Pattern Recognition. It is assumed that dynamical systems under consideration exhibit nonlinear dynamical behavior on two time scales. Anomalies occur on a slow time scale that is (possibly) several orders of magnitude larger than the fast time scale of the system dynamics. It is also assumed that the unforced dynamical system (i.e., in the absence of external stimuli) is stationary at the fast time scale and that any non-stationary behavior is observable only on the slow time scale. This concept of small change detection in dynamical systems is elucidated on an active electronic circuit representing the forced Duffing equation with a slowly varying dissipation parameter. The time-series data of stationary phase trajectories are collected to create the respective symbolic dynamics (i.e., symbol sequences) using wavelet transform. The resulting state probability vector of the transition matrix is considered as the vector representation of a 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. This vector representation of anomalies is more powerful than a scalar measure. The major conclusion of this research is that Symbolic Dynamics along with the stimulus-response methodology and having a vector representation of anomaly is effective for early detection of small anomalies.

The D-Markov machine, proposed for anomaly detection, is a suboptimal approximation of the ε-machine. It is important that this approximation is a sufficiently accurate representation of the nominal behavior. Research in this direction is in progress and the results will be presented in a forthcoming publication. Further theoretical research is recommended in the following areas:

- Separation of information-bearing part of the signal from noise.
- Identification of a relevant submanifold of the phase space and its partitioning to generate a symbol alphabet.
- Identification of appropriate wavelet basis functions for symbol generation and construction of a mapping from the wavelet space to the symbol space.
- Selection of the minimal D for the D-Markov machine and identification of the irreducible submatrix of the state transition matrix that contains relevant

information on anomalous behavior of the dynamical system.

Acknowledgements

The author wishes to thank Dr. Cosma Shalizi for making the CSSR code available and also for clarifying some of the underlying concepts of the ε-machine. The author is also thankful to Dr. Matthew Kennel for providing him with the software code on phase-space partitioning using symbolic false nearest neighbors. The author acknowledges technical contributions of Mr. Amit Surana and Mr. David Friedlander in developing the anomaly detection concept and of Mr. Venkatesh Rajagopalan for design and fabrication of the experimental apparatus on active electronic circuits.

Appendix A. Information theoretic quantities

This appendix introduces the concepts of standard information-theoretic quantities: *entropy rate*, *excess entropy* and *statistical complexity* [11], which are used to establish the anomaly measure in Section 5.

Entropy rate (h_μ): The entropy rate of a symbol string $\mathscr{S}$ is given by the Shannon entropy as follows:

$$h_\mu = \lim_{L\to\infty} \frac{H[L]}{L}, \tag{A.1}$$

where $H[L] \equiv -\sum_{s^L\in\mathscr{A}^L} P(s^L)\log_2(P(s^L))$ is the Shannon entropy of all L-blocks (i.e., symbol sequences of length L) in $\mathscr{S}$. The limit is guaranteed to exist for a stationary process [5]. The entropy rate quantifies the irreducible randomness in sequences produced by a source: the randomness that remains after the correlation and the structures in longer and longer sequence blocks are taken into account. For a symbol string $\mathscr{S}$ represented as an ε-machine, $h_\mu = H[\overrightarrow{S}^1|\mathscr{S}]$.

Excess entropy (**E**): The excess entropy of a symbol string $\mathscr{S}$ is defined as

$$\mathbf{E} = \sum_{L=1}^{\infty} [h_\mu(L) - h_\mu] \tag{A.2}$$

where $h_\mu(L) \equiv H[L]-H[L-1]$ is the estimate of how random the source appears if only L-blocks in $\mathscr{S}$ are considered. Excess entropy measures how much additional information must be gained about the sequence in order to reveal the actual per-symbol uncertainty h_μ, and thus measures difficulty in the prediction of the process. Excess entropy has alternate interpretations such as: it is the intrinsic redundancy in the process; geometrically it is a sub-extensive part of $H(L)$; and it represents how much historical information stored in the present is communicated to the future.

Statistical complexity (C_μ) [11]: The information of the probability distribution of causal states, as measured by Shannon entropy, yields the minimum average amount of memory needed to predict future configurations. This quantity is the *statistical complexity* of a symbol string $\mathscr{S}$, defined by Crutchfield and Young [7] as

$$C_\mu \equiv H(\mathscr{S}) = -\sum_{k=0}^{n-1} [Pr(S_k)\log_2 Pr(S_k)], \tag{A.4}$$

where n is the number of states of the finite-state machine constructed from the symbol string $\mathscr{S}$. As shown in [11], $\mathbf{E} \leqslant C_\mu$ in general, and $C_\mu = \mathbf{E} + Dh_\mu$.

Appendix B. Finite-type shift and sofic shift

This appendix very briefly introduces the concept of shift spaces with emphasis on finite shifts and sofic shifts that respectively characterize the D-Markov machine and the ε-machine described in the Section 4.4. The shift space formalism is a systematic way to study the properties of the underlying grammar, which represent the behavior of dynamical systems encoded through symbolic dynamics. The different shift spaces provide increasingly powerful classes of models that can be used to represent the patterns in the dynamical behavior.

Definition 2.1. Let $\mathscr{A}$ be a finite alphabet. The *full $\mathscr{A}$-shift* is the collection of all bi-infinite sequences of symbols from $\mathscr{A}$ and is denoted by:

$$\mathscr{A}^Z = \{x = (x_i)_{i\in Z}\colon x_i \in \mathscr{A}\ \forall i \in Z\}. \tag{A.5}$$

Definition 2.2. The *shift map* σ on the full shift $\mathscr{A}^Z$ maps a point x to a point $y=\sigma(x)$ whose ith coordinate is $y_i = x_{i+1}$.

A block is a finite sequence of symbols over $\mathscr{A}$. Let $x \in \mathscr{A}^Z$ and w be a block over $\mathscr{A}$. Then w occurs in x

if $\exists$ indices i and j such that $w = x_{[i,j]} = x_i x_{i+1} \cdots x_j$. Note that the empty block ε occurs in every x.

Let $\mathscr{F}$ be a collection of blocks, i.e., finite sequences of symbols over $\mathscr{A}$. Let $x \in \mathscr{A}^Z$ and w be a block over $\mathscr{A}$. Then w occurs in x if $\exists$ indices i and j such that $w = x_{[i,j]} = x_i x_{i+1} \cdots x_j$. For any such $\mathscr{F}$, let us define $X_{\mathscr{F}}$ to be the subset of sequences in $\mathscr{A}^Z$, which do not contain any block in $\mathscr{F}$.

Definition 2.3. A *shift space* is a subset X of a full shift $\mathscr{A}^Z$ such that $X = X_{\mathscr{F}}$ for some collection $\mathscr{F}$ of forbidden blocks over $\mathscr{A}$.

For a given shift space, the collection $\mathscr{F}$ is at most countable (i.e., finite or countably infinite) and is non-unique (i.e., there may be many such $\mathscr{F}$'s describing the shift space). As subshifts of full shifts, these spaces share a common feature called *shift invariance*. Since the constraints on points are given in terms of forbidden blocks alone and do not involve the coordinate at which a block might be forbidden, it follows that if $x \in X_{\mathscr{F}}$, then so are its shifts $\sigma(x)$ and $\sigma^{-1}(x)$. Therefore $\sigma(X_{\mathscr{F}}) = X_{\mathscr{F}}$, which is a necessary condition for a subset of $\mathscr{A}^Z$ to be a shift space. This property introduces the concept of shift dynamical systems.

Definition 2.4. Let X be a shift space and $\sigma_X : X \to X$ be the shift map. Then (X, σ_X) is known as a *shift dynamical system*.

The shift dynamical system mirrors the dynamics of the original dynamical system from which it is generated (by symbolic dynamics). Several examples of shift spaces are given in [15].

Rather than describing a shift space by specifying the forbidden blocks, it can also be specified by allowed blocks. This leads to the notion of a *language* of a shift.

Definition 2.5. Let X be a subset of a full shift, and let $\mathscr{B}_n(X)$ denote the set of all n-blocks (i.e., blocks of length n) that occur in X. The language of the shift space X is defined as:

$$\mathscr{B}(X) = \bigcup_{n=0}^{\infty} \mathscr{B}_n(X). \tag{A.6}$$

Sliding block codes: Let X be a shift space over $\mathscr{A}$, then $x \in X$ can be transformed into a new sequence $y = \cdots y_{-1} y_0 y_1 \cdots$ over another alphabet $\mathscr{U}$ as follows. Fix integers m and n such that $-m \leqslant n$. To compute y_i of the transformed sequence, we use a function Φ that depends on the "window" of coordinates of x from $i - m$ to $i + n$. Here $\Phi : \mathscr{B}_{m+n+1}(X) \to \mathscr{U}$ is a fixed *block map*, called a $(m + n + 1)$-*block map* from the allowed $(m + n + 1)$-blocks in X to symbols in $\mathscr{U}$. Therefore,

$$y_i = \Phi(x_{i-m} x_{i-m+1} \cdots x_{i+n}) = \Phi(x_{[i-m,i+n]}). \tag{A.7}$$

Definition 2.6. Let Φ be a block map as defined in Eq. (A.7). Then the map $\phi : X \to (\mathscr{U})^Z$ defined by $y = \phi(x)$ with y_i given by Eq. (A.7) is called the *sliding block code* with memory m and anticipation n induced by Φ.

Definition 2.7. Let X and Y be shift spaces, and $\phi : X \to Y$ be a sliding block code.

- If $\phi : X \to Y$ is onto, then ϕ is called a *factor code* from X onto Y.
- If $\phi : X \to Y$ is one-to-one, then ϕ is called an *embedding* of X into Y.
- If $\phi : X \to Y$ has a inverse (i.e., $\exists$ a sliding block code $\psi : Y \to X$ such that $\psi(\phi(x)) = x \; \forall x \in X$ and $\phi(\psi(y)) = y \; \forall \; y \in Y$), then ϕ is called a *conjugacy* from X to Y.

If $\exists$ a conjugacy from X to Y, then Y can be viewed as a copy of X, sharing all properties of X. Therefore, a conjugacy is often called topological conjugacy in literature.

Finite-type shifts: We now introduce the concept of finite-type shift that is the structure of the shift space in the D-Markov machine proposed in the Section 4.2.

Definition 2.8. A *finite-type shift* is a shift space that can be described by a finite collection of forbidden blocks (i.e., X having the form $X_{\mathscr{F}}$ for some finite set $\mathscr{F}$ of blocks).

An example of a finite shift is the *golden mean shift*, where the alphabet is $\Sigma = \{0, 1\}$ and the forbidden set $\mathscr{F} = \{11\}$. That is, $X = X_{\mathscr{F}}$ is the set of all binary sequences with no two consecutive 1's.

Definition 2.9. A finite-type shift is M-step or has memory M if it can be described by a collection of forbidden blocks all of which have length $M+1$.

The properties of a finite-type shift are listed below:

- If X is a finite-type shift, then $\exists M \geqslant 0$ such that X is M-step.
- The language of the finite-type shift is characterized by the property that if two words overlap, then they can be glued together along their overlap to from another word in the language. Thus, a shift space X is an M-step finite-type shift iff whenever uv, $vw \in \mathscr{B}(X)$ and $|v| \geqslant M$, then $uvw \in \mathscr{B}(X)$.
- A shift space that is conjugate to a finite-type shift is itself a finite-type shift.
- A finite-type shift can be represented by a finite, directed graph and produces the collection of all bi-infinite walks (i.e. sequence of edges) on the graph.

Sofic shifts: The sofic shift is the structure of the shift space in the ε-machines [7,20] in Section 4.1. Let us label the edges of a graph with symbols from an alphabet $\mathscr{A}$, where two or more edges are allowed to have the same label. Every bi-infinite walk on the graph yields a point in $\mathscr{A}^Z$ by reading the labels of its edges, and the set of all such points is called a *sofic shift*.

Definition 2.10. A graph G consists of a finite set $\mathscr{V} = \mathscr{V}(G)$ of vertices together with a finite set $\mathscr{E} = \mathscr{E}(G)$ of edges. Each edge $e \in \mathscr{E}(G)$ starts at a vertex denoted by $i(e) \in \mathscr{V}(G)$ and terminates at a vertex $t(e) \in \mathscr{V}(G)$ (which can be the same as $i(e)$). There may be more that one edge between a given initial state and terminal state; a set of such edges is called a set of multiple edges. An edge e with $i(e) = t(e)$ is called a self-loop.

Definition 2.11. A *labelled graph* $\mathscr{G}$ is a pair $(G, \mathscr{L})$, where G is a graph with edge set $\mathscr{E}$, and $\mathscr{L} : \mathscr{E} \to \mathscr{A}$ assigns a label $\mathscr{L}(e)$ to each edge e of G from the finite alphabet $\mathscr{A}$. The *underlying graph* of $\mathscr{G}$ is G.

Definition 2.12. A subset X of a full shift is a *sofic shift* if $X = X_{\mathscr{G}}$ for some labelled graph $\mathscr{G}$. A *presentation* of a sofic shift X is a labelled graph $\mathscr{G}$ for which $X_{\mathscr{G}} = X$.

An example of a sofic shift is the *even shift*, which is the set of all binary sequences with only even number of 0's between any two 1's. That is, the forbidden set $\mathscr{F}$ is the collection $\{10^{2n+1}: n \geqslant 0\}$.

Some of the salient characterization of sofic shifts are presented below [15]:

- Every finite-type shift qualifies as a sofic shift.
- A shift space is sofic iff it is a factor of a finite-type shift.
- The class of sofic shifts is the smallest collection of shifts spaces that contains all finite-type shifts and also contains all factors of each space in the collection.
- A sofic shift that does not have finite-type subshifts is called a *strictly sofic*. For example, the *even shift* is strictly sofic [15].
- A factor of a sofic shift is a sofic shift.
- A shift space conjugate to a sofic shift is itself sofic.
- A distinction between finite-type shifts and sofic shifts can be characterized in terms of the *memory*. While finite-type shifts use finite-*length* memory, sofic shifts require finite *amount* of memory. In contrast, context-free shifts require infinite amount of memory [12].

References

[1] H.D.I. Abarbanel, The Analysis of Observed Chaotic Data, Springer, New York, 1996.
[2] R. Badii, A. Politi, Complexity Hierarchical Structures and Scaling in Physics, Cambridge University Press, UK, 1997.
[3] R.B. Bapat, T.E.S. Raghavan, Nonnegative Matrices and Applications, Cambridge University Press, Cambridge, 1997.
[4] C. Beck, F. Schlogl, Thermodynamics of Chaotic Systems: an Introduction, Cambridge University Press, UK, 1993.
[5] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley, New York, 1991.
[6] J.P. Crutchfield, The calculi of emergence: Computation, dynamics and induction, Physica D 75 (1994) 11–54.
[7] J.P. Crutchfield, K. Young, Inferring statistical complexity, Phys. Rev. Let. 63 (1989) 105–108.
[8] R.L. Davidchack, Y.C. Lai, E.M. Bolt, H. Dhamala, Estimating generating partitions of chaotic systems by unstable periodic orbits, Phys. Rev. E 61 (2000) 1353–1356.
[9] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley, New York, 2001.

[10] D.P. Feldman, Computational mechanics of classical spin systems, Ph.D. Dissertation, University of California, Davis, 1998.
[11] D.P. Feldman, J.P. Crutchfield, Discovering non-critical organization: statistical mechanical, information theoretic, and computational views of patterns in one-dimensional spin systems, Santa Fe Institute Working Paper 98-04-026, 1998.
[12] H.E. Hopcroft, R. Motwani, J.D. Ullman, Introduction to Automata Theory, Languages, and Computation, 2nd Edition, Addison-Wesley, Boston, 2001.
[13] M.B. Kennel, M. Buhl, Estimating good discrete partitions form observed data: symbolic false nearest neighbors, http://arxiv.org/PS_cache/nlin/pdf/0304/0304054.pdf, 2003.
[14] B.P. Kitchens, Symbolic Dynamics: One Sided, Two sided and Countable State Markov Shifts, Springer, New York, 1998.
[15] D. Lind, M. Marcus, An Introduction to Symbolic Dynamics and Coding, Cambridge University Press, UK, 1995.
[16] M. Markou, S. Singh, Novelty detection: a review—parts 1 and 2, Signal Processing 83 (2003) 2481–2521.
[17] A.W. Naylor, G.R. Sell, Linear Operator Theory in Engineering and Science, Springer, New York, 1982.
[18] E. Ott, Chaos in Dynamical Systems, Cambridge University Press, UK, 1993.
[19] R.K. Pathria, Statistical Mechanics, 2nd Edition, ButterworthHeinemann, Oxford, UK, 1998.
[20] C.R. Shalizi, K.L. Shalizi, J.P. Crutchfield, An algorithm for pattern discovery in time series, SFI Working Paper 02-10-060, 2002.
[21] A. Surana, A. Ray, S.C. Chin, Anomaly detection in complex systems, in: Fifth IFAC Symposium on Fault Detection, Supervision and Safety of Technical Process, Washington, DC, 2003.
[22] F. Takens, Detecting strange attractors in turbulence, in: D. Rand, L.S. Young (Eds.), Proceedings of the Symposion Dynamical Systems and Turbulence, Warwick, 1980, Lecture Notes in Mathematical, Vol. no 898, Springer, Berlin, 1981, p. 366.
[23] J.M.T. Thompson, H.B. Stewart, Nonlinear Dynamics and Chaos, Wiley, Chichester, UK, 1986.
[24] D.R. Upper, Theory and algorithms for hidden Markov models and generalized hidden Markov models, Ph.D. Dissertation in Mathematics, University of California, Berkeley, 1997.
[25] H. Zang, A. Ray, S. Phoha, Hybrid life extending control of mechanical systems: experimental validation of the concept, Automatica 36 (2000) 23–36.