AFRL-AFOSR-VA-TR-2022-0064

Dynamic Data-driven Prediction, Measurement Adaptation

Asok Ray
PENNSYLVANIA STATE UNIVERSITY
201 OLD MAIN
UNIVERSITY PARK, PA,
US

12/13/2021
Final Technical Report

Air Force Research Laboratory
Air Force Office of Scientific Research
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
13-12-2021	Final	30 Sep 2015 - 29 Sep 2021

4. TITLE AND SUBTITLE
Dynamic Data-driven Prediction, Measurement Adaptation

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-15-1-0400

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Asok Ray

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
PENNSYLVANIA STATE UNIVERSITY
201 OLD MAIN
UNIVERSITY PARK, PA
US

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AF Office of Scientific Research
875 N. Randolph St. Room 3112
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR RTA2

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-VA-TR-2022-0064

12. DISTRIBUTION/AVAILABILITY STATEMENT
A Distribution Unlimited: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
Scientific advancements in the development of fast dynamic data-driven tools for forecasting and detection of thermo-acoustic instabilities as well as of fatigue damage in aircraft structures: The DDDAS paradigm has been used with streaming sensor data for forecasting & detection, and classification of emerging anomalies. The algorithmic advancements are achieved using differential-geometric concepts of deep neural networks and statistical learning with hidden Markov models for sequential pattern classification, as an alternative to symbolic time series analysis. The proposed methods have been validated on time-series data, generated from a laboratory-scale combustor apparatuses, operating under different protocols at varying air-fuel premixing levels. Applicability of the proposed method has been demonstrated with respect to anomaly detection and regime identification with limited data requirements, making it a potential candidate for near-real-time monitoring and active control of thermo-acoustic instabilities in commercial-scale combustors. Experimental Research and Validation: A computer-instrumented and computer-controlled laboratory apparatus has been designed.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			ERIK BLASCH
U	U	U	UU	35	**19b. TELEPHONE NUMBER** *(Include area code)* 426-7311

Standard Form 298 (Rev.8/98)
Prescribed by ANSI Std. Z39.18

2020 ANNUAL PROGRESS SUMMARY

Subject: Annual Progress Summary **PM:** Dr. Erik Blasch, erik.blasch@gmail.com

Contract/Grant Title: Dynamic Data-driven Prediction, Measurement Adaptation, and Active Control of Combustion Instabilities in Aircraft Gas Turbine Engines

Contract/Grant No: FA9550-15-1-0400 **Reporting Period:** September 30, 2019 to September 29, 2020

PI: Prof. Asok Ray, axr2@psu.edu **Co-PI:** None **Senior Research Personnel**: Dr. Shashi Phoha

Current Graduate Students (2): C. Bhattacharya and X. Chen who have passed their PhD comprehensive and PhD qualifying examinations, respectively.

Former Graduate Students (9) (graduated with PhD):

2020	Y. Fu and S. Mondal
2019	N.J. Ghalyan
2018	M. Hauser and P. Chattopadhyay
2017	N. Virani and S. Xiong;
2016	Y. Li and D.K. Jha.

Annual Accomplishments

Theoretical Research:

Scientific advancements in the development of fast dynamic data-driven tools for forecasting and detection of thermo-acoustic instabilities as well as of fatigue damage in aircraft structures: The DDDAS paradigm has been used with streaming sensor data for *forecasting* & detection, and classification of emerging anomalies. The algorithmic advancements are achieved using differential-geometric concepts of deep neural networks and statistical learning with hidden Markov models for sequential pattern classification, as an alternative to symbolic time series analysis. The proposed methods have been validated on time-series data, generated from a laboratory-scale combustor apparatuses, operating under different protocols at varying air-fuel premixing levels. Applicability of the proposed method has been demonstrated with respect to anomaly detection and regime identification with limited data requirements, making it a potential candidate for near-real-time monitoring and active control of thermo-acoustic instabilities in commercial-scale combustors.

Experimental Research and Validation:

A computer-instrumented and computer-controlled laboratory apparatus has been designed for operation under nominal and off-nominal anticipated & unanticipated transient operating conditions as well as emergency situations in typical combustors of aircraft gas-turbine engines. Construction of a laboratory apparatus under AFOSR DURIP Grant No. **FA9550-18-1-0135** has been completed. This apparatus serves as a test facility for near-real-time prediction and active control of combustion instabilities in aircraft gas-turbine engines.

Changes in research objectives, if any: None.

Change in AFOSR program manager, if any: None.

Extensions granted or milestones slipped, if any: None.

Honors and awards, if any:

- Fellowship Award of **American Association for Advancement of Science (**AAAS) to Dr. Asok Ray (PI)
- Best paper award among all issues of the following Journal in the year 2018:

N. Virani, D.K. Jha, Z. Yuan, I. Sekhawat and A. Ray, "Imitation of Demonstrations using Bayesian Filtering with Nonparametric Data-Driven Models," ASME Journal: Dynamic Systems, Measurement and Control, 140(3), March 2018, pp. 030906 (1 to 9).

Include any new discoveries, inventions, or patent disclosures during this reporting period:
Technology innovation for efficient and fault-mitigating design of gas turbine engine combustors using dynamic data-driven tools:

S. Sarkar, T. Damarla, and A. Ray, Apparatus and Method for Activity Detection and Classification from Sensor Data, U.S. Patent No. 10,621,506, April 14, 2020.

Description of the Laboratory Apparatus for Experimental Validation of the Theoretical Results

A defense university research instrumentation program (DURIP) proposal was funded from March 01, 2018 to February 29, 2020. The funds were used to procure mechanical & electronic components and computational hardware & software for construction of a computer-instrumented and computer-controlled experimental apparatus that serves as a test bed for early prediction and real-time active control of (emulated) combustion instabilities in aircraft gas turbine engines. This Rijke tube test apparatus was specifically designed to be capable of emulating nominal and some of the critical anticipated & unanticipated off-nominal transient operating conditions as well as emergency situations in typical combustors of aircraft gas turbine engines. Figure 1 and Figure 2 respectively present a functional diagram and a schematic diagram of the Rijke tube test apparatus. The apparatus is now fully operational within the tenure of the current research project **FA9550-15-1-0400**, which has been no-cost extended to September 29, 2021.

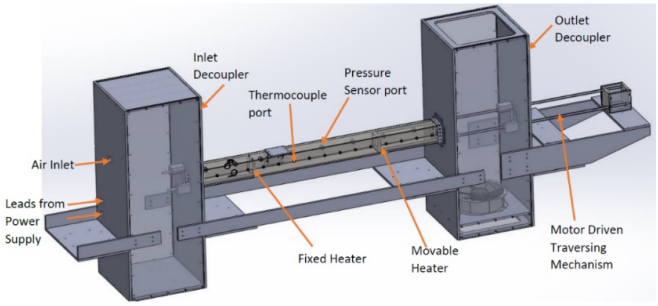

Figure 1. Functional diagram of the Rijke tube test apparatus

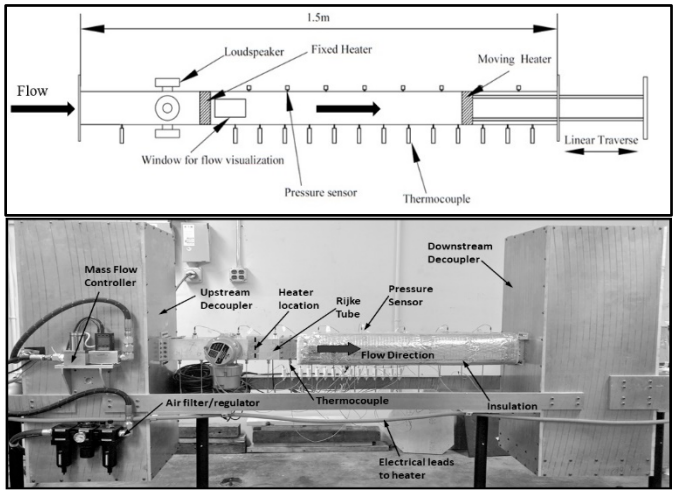

Figure 2. Schematic diagram of the Rijke tube test apparatus

The test apparatus is currently supporting experimental validation of the results of theoretical research in DDDAS, which include algorithms of the following five areas:

- Real-time knowledge extraction from heterogeneous dynamic data in stochastic settings;
- Reduced-order modeling trained by experimental data of spatio-temporal dynamics;
- Machine learning and information fusion at multiple scales of space and time; and
- Uncertainty quantification in real-time monitoring & control of combustion instabilities.
- Reinforcement learning and control of thermos-acoustic oscillations and the concept validation on the Rijke tube experimental apparatus.

Technical Progress in the Last Year (09/30 2019 to 09/29/2020) of the Research Project

The basic research conducted during the last year (09/30 2019 to 09/29/2020) of this research project, which has advanced the theoretical framework of dynamic data-driven application systems (DDDAS) for machine perception & learning for aircraft gas-turbines, has been reported in fourteen (14) archive journal publications, and another ten (10) publications are under review. Some of the results of the theoretical research have been experimentally validated on the experimental apparatus that has been recently constructed under AFOSR DURIP Grant No. FA9550-18-1-0135.

List of Archive Journal Publications Citing the Grants (October 2019- September 2020)

The following publications have been produced in the last year of the research grant:

(A) ***Theoretical research in dynamic data-driven application systems (DDDAS) with experimental validation***

1. N.F. Ghalyan and A. Ray, "Symbolic Time Series Analysis for Anomaly Detection in Measure-Invariant Ergodic Systems, " ASME Journal of Dynamic Systems, Measurement, and Control, vol. 142, June 2020, pp. 061003 (1 to 11). doi: 10.1115/1.4046156
2. C. Bhattacharya and A. Ray, "Online Discovery and Classification of Operational Regimes from an Ensemble of Time Series Data," to appear in ASME Journal of Dynamic Systems, Measurement, and Control, vol. 142, November 2020, pp. 114501 (1 to 7). doi: 10.1115/1.4047449
3. D.J. Miller, N.F.Ghalyan, S. Mondal and A. Ray, "HMM Conditional-Likelihood Based Change Detection with Strict Delay Tolerance," to appear in Mechanical Systems and Signal Processing, vol. 147, January 2021, pp. 10719 (1 t0 17). doi: 10.1016/j.ymssp.2020.107109 (**Collaboration with another DDDAS Project under Grant: FA9550-17-1-0070**)
4. C. Bhattacharya and A. Ray, "Data-driven Detection and Classification of Regimes in Chaotic Systems via Hidden Markov Modeling," ASME Letters in Dynamic Systems and Control, vol. 1, no. 2, April 2021, p. 021009 (5 pages) doi: 10.1115/1.4047817

(B) ***Dynamic data-driven prediction of combustion instabilities in aircraft gas-turbines***

5. C. Bhattacharya, S. Mondal, A. Ray and A. Mukhopadhyay, "Reduced-order Modeling of Thermoacoustic Instabilities in a Two-heater Rijke Tube," Combustion Theory and Modeling, vol. 24, no. 3, June 2020, pp. 530-548. doi: 10.1080/13647830.2020.1714080 . (**Collaboration with Jadavpur University, India)**
6. C. Bhattacharya, S. De, A. Mukhopadhyay, S. Sen, and A. Ray, "Detection and Classification of Lean Blow-Out and Thermoacoustic Instability in Turbulent Combustors," Appled Thermal Engineering, in press. doi: 10.1016/j.applthermaleng.2020.115808 (**Collaboration with Jadavpur University, India)**
7. S. Mondal, S. De, A. Mukhopadhyay, S. Sen, and A. Ray, "Early Prediction of Lean Blowout from Chemiluminescence Time Series Data," Combustion Science and Technology, in press. doi: 10.1080/00102202.2020.1804380 (**Collaboration with Jadavpur University, India)**
8. C. Bhattacharya, J. O'Connor and A. Ray, "Data-driven Early Detection of Thermoacoustic Instability in a Multi-nozzle Combustor," Combustion Science and Technology, in press. doi: 10.1080/00102202.2020.1820495

(C) ***Technology innovation for design of gas turbine engine combustors using dynamic data-driven tools***

9. S. Ghosh, S. Mondal, E. Farnandez, J.S. Kapat, and A. Ray, "Parametric Shape Optimization of Pin-Fin Arrays Using a Surrogate Model based Bayesian Method," AIAA Journal of Thermophysics and Heat Transfer , in press. (**Collaboration with University of Central Florida, Orlando, Florida**)

(D) ***Technology innovation in related fields using dynamic data-driven tools***

10. N.F. Ghalyan, I.F. Ghalyan and A. Ray, "Modeling of Microscope Images for Early detection of Fatigue Cracks in Structural Materials," The International Journal of Advanced Manufacturing Technology, Springer Nature, vol. 104, November 2019, pp. 3899-3913. doi: 10.1007/s00170-019-04108-z
11. X. Chen and A. Ray, "On Singular Perturbation of Neutron Point Kinetics in the Dynamic Model of a PWR Nuclear Power Plant," Sci: An MDPI Journal (ISSN 2413-4155), vol. 30, no. 2, April 2020, pp. 1-10. doi:10.3390/sci2020030
12. S. Mondal, D. Gwynn, A. Ray and A. Basak, "Investigation of Melt Pool Geometry Control in Additive Manufacturing using Hybrid Modeling," Metals: An MDPI Journal, vol. 10, no. 5, May 2020, pp. 1-24. doi:10.3390/met10050683
13. S. Dharmadhikari,, E. Keller, A. Ray, and A.Basak, "A Dual-Imaging Framework for Simultaneous Measurements of Short and Long Crack Initiation and Propagation in Metallic Materials," International Journal of Fatigue , in press. doi: 10.1016/j.ijfatigue.2020.105922
14. H. Alqatani, S. Bharadwaj, and A. Ray, "Classification of Fatigue Crack Damage in Polycrystalline Alloy Structures Using Convolutional Neural Networks," to appear in Engineering Failure Analysis

Potential Applications of Research Results and Technology Transfer to DoD/Industry

■ Potential projects for future technology transfer include:

- *Joint Strike Fighter (JSF)*
- *Unmanned Compact Air System (UCAS)*
- *Versatile Affordable Advanced Turbine Engines (VAATE)*
- *Next Generation Product Family commercial engines.*

Coordination with AFRL and other organizations

AFRL Collaboration POC:	Dr. Alireza Behbahani, Turbine Engine Division at WPAFB, OH
DOD Collaboration POC	Dr. Thyagaraju Damarla, Army Research Laboratory at Adelphi, MD
Other DDDAS Project PIs:	Dr. David Miller (Grant No.FA9550-17-1-0070)
Other Universities:	Dr. J.S. Kapat, University of Central Florida, Orlando, Florida
	Dr. A. Mukhopadhyay, Jadavpur University, India

Research Plan for Next Year (October 2020 - September 2021)

The research plan for the next year is outlined below:

Theoretical Research on Dynamic Data-driven Application Systems (DDDAS):

A summary of the theoretical research for the next year is planned in the following areas:

1) Continuation of the research activities on development of information fusion algorithms based on multimodal sensor (e.g., acoustic pressure and flame image) data for monitoring and control of thermo-acoustic instabilities.
2) Continuation of the research activities on development of robust and resilient algorithms of dynamic data-driven pattern recognition based on the concepts of symbolic time series analysis, convolutional neural network (CNN), and generalization of Markov model construction. These algorithms will address operations under (emulated) nominal and critical anticipated & unanticipated off-nominal transient conditions as well as emergency situations in typical combustors of aircraft gas-turbine engines.

3) Continuation of the research activities including development of statistical learning with hidden Markov models (HMM) for sequential pattern classification, for a fast dynamic data-driven method to detect early onsets of thermo-acoustic instabilities.
4) New research on development of a reinforcement learning concept for active control of thermo-acoustic oscillations and the concept validation on the Rijke tube experimental apparatus.

A brief description of the new research to be conducted on the above item 4 is presented below.

Passive controllers of combustion instabilities, which are largely restricted to acoustic dampers (e.g. Helmholtz resonators) and variable combustor geometry, have been extensively studied. However, there are significant limitations of these techniques, because of complex designs of actual combustors, geometry modifications are unreliable in practice. Addition of extra hardware like dampers can significantly increase weight of combustors and thus negatively affect the vehicle performance. Moreover, passive control mechanisms have a limited frequency range of application.

The dynamics of combustion process in aircraft applications have been traditionally described by coupled nonlinear partial differential equations, which cause significant challenges in modeling and thus may render the model-based controllers unreliable. Therefore, we propose to use reinforcement learning algorithms to directly learn an optimal policy for active control of the secondary flame from data. Before applying these algorithms to actual combustors, it is logical to first evaluate their performance with simulation and experimental data from the Rijke tube apparatus in Figure 1 and Figure 2. Reinforcement learning will determine how agents ought to take actions based on the observations, received from environment, so that the cumulative reward is maximized. In this case, the controller is the agent and the combustor is the environment. The schematic plot of reinforcement learning for active control of combustion instabilities using the secondary heater of the Rijke tube apparatus is shown in Figure 3.

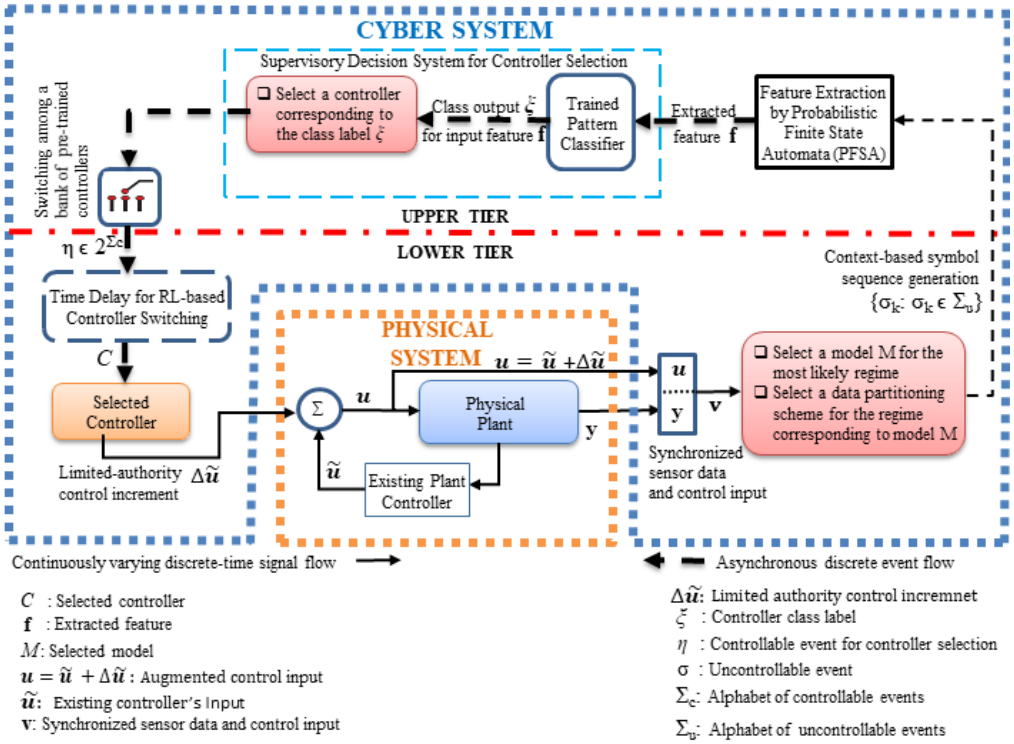

Figure 3. Active control of combustion instabilities in the setting of cyber-physical systems

One of the reinforcement learning algorithms under consideration is trust region policy optimization (TRPO), which is a model free policy search algorithm. It directly models the control policy with a neural network and attempts to maximize the lower bound of the cumulative reward for each iteration. In this application, the controller is represented by a gated recurrent neural network. The input of the controller is the pressure time series observed between the current and the previous action. For example, the output of the controller may take one of the three preassigned values

Experimental Research *on Dynamic Data-driven Application Systems (DDDAS):*
The Rijke tube apparatus, constructed under AFOSR DURIP Grant No. **FA9550-18-1-0135**, will continue to serve as a test facility for experimental validation of the theoretical results for near-real-time monitoring and active control of combustion instabilities in aircraft gas-turbine engines.

Appendix: Three Samples of Publications under this grant

Copies of the following three journal publications under the AFOSR Grant No: FA9550-15-1-0400 are attached in the next pages:

1. C. Bhattacharya and A. Ray, "Online Discovery and Classification of Operational Regimes from an Ensemble of Time Series Data," to appear in ASME Journal of Dynamic Systems, Measurement, and Control, vol. 142, November 2020, pp. 114501 (1 to 7). doi: 10.1115/1.4047449
2. D.J. Miller, N.F.Ghalyan, S. Mondal and A. Ray, "HMM Conditional-Likelihood Based Change Detection with Strict Delay Tolerance," Mechanical Systems and Signal Processing, vol. 147, January 2021, pp. 10719 (1 t0 17). doi: 10.1016/j.ymssp.2020.107109
3. C. Bhattacharya and A. Ray, "Data-driven Detection and Classification of Regimes in Chaotic Systems via Hidden Markov Modeling," ASME Letters in Dynamic Systems and Control, vol. 1, no. 2, April 2021, p. 021009 (5 pages) doi: 10.1115/1.4047817

Online Discovery and Classification of Operational Regimes From an Ensemble of Time Series Data

Chandrachur Bhattacharya
Departments of Mechanical and Electrical Engineering,
Pennsylvania State University,
University Park, PA 16802
e-mail: chandrachur.bhattacharya@gmail.com

Asok Ray
Fellow ASME
Departments of Mechanical Engineering and Mathematics,
Pennsylvania State University,
University Park, PA 16802
e-mail: axr2@psu.edu

One of the pertinent problems in decision and control of dynamical systems is to identify the current operational regime of the physical process under consideration. To this end, there has been an upsurge in (data-driven) machine learning methods, such as symbolic time series analysis, hidden Markov modeling, and artificial neural networks, which often rely on some form of supervised learning based on preclassified data to construct the classifier. However, this approach may not be adequate for dynamical systems with a variety of operational regimes and possible anomalous/failure conditions. To address this issue, the technical brief proposes a methodology, built upon the concept of symbolic time series analysis, wherein the classifier learns to discover the patterns so that the algorithms can train themselves online while simultaneously functioning as a classifier. The efficacy of the methodology is demonstrated on time series of: (i) synthetic data from an unforced Van der Pol equation and (ii) pressure oscillation data from an experimental Rijke tube apparatus that emulates the thermoacoustics in real-life combustors where the process dynamics undergoes changes from the stable regime to an unstable regime and vice versa via transition to transient regimes. The underlying algorithms are capable of accurately learning and capturing the various regimes online in a (primarily) unsupervised manner. [DOI: 10.1115/1.4047449]

Keywords: pattern discovery and classification, symbolic dynamics, anomaly detection

1 Introduction

Large-scale dynamical systems, depending on their complexity, may exhibit a variety of operational regimes and a number of possible anomalous/failure conditions. Thus, when a control command is issued, it needs to be conditioned on both present and intended operational regimes. Accordingly, traditional classification techniques are often based on the concept of supervised learning, wherein the classifier is trained on labeled data corresponding to various regimes and anomalous characteristics of the underlying process.

Pattern recognition is a mature field of research, which has found its applications in many aspects of machine learning. Several methods [1,2] exist for identification of various classes that the observed data emanate from. Many researchers have made use of several machine learning techniques for time series classification in diverse applications. Bagnall et al. [3] have reported a comprehensive review of time series analysis for online detection, classification, and decision and control. Sugimura and Matsumoto [4] have reported feature extraction from time series to generate a decision tree for pattern classification. The concepts of reduced-order Markov modeling and probabilistic finite state automata (PFSA) have been studied for analysis of combustion instabilities [5], failure prognosis of structural materials [6] and rolling-element bearings [7], as well as usage of sensor networks for detection of moving targets [8]. Several researchers have also used deep neural networks [9] as well as recurrent neural networks for time series classification [10]. However, not all of these techniques are suitable for online detection, which requires real-time classification (e.g., by using as short a time series window as possible). Online regime detection and classification have been demonstrated by Mondal et al. [11], in the setting of hidden Markov models and by Lim and Harrison [12] in the setting of neural networks, for example.

In a vast majority of the methods mentioned previously, the models are trained offline (i.e., no simultaneous learning and classification). Furthermore, all of the above algorithms belong to the category of supervised or semisupervised learning in the sense that they need labeled data, where the labeling is done by (human) domain experts. Even when using an unsupervised technique such as K-means [2] for time series analysis [13], a human expert must decide the number of regimes (i.e., the parameter K). Other algorithms such as Gaussian mixture models are also used for learning subclasses in populations; however, for this approach to work, data from all possible classes need to be considered during offline training. These methods are flexible for learning multiple classes only if a human supervisor pre-informs the algorithm about the actual number of classes the given data needs to be divided into. In such a situation, if the algorithm comes across a hitherto unfamiliar anomaly or signal during the online classification phase, it may misclassify the operation as one of the trained regimes. Often, it is not possible to have a training dataset containing all possible events, especially in systems that may be subjected to occurrence of rare events.

This technical brief proposes a low-computational-cost methodology that is built upon a concept that adaptively learns new patterns and signal forms in an online fashion (i.e., in real-time) without the need to know the total number of classes a priori. The efficacy of the proposed methodology is demonstrated on both synthetic data from an unforced Van der Pol oscillator and experimental data of pressure time series from an electrically heated Rijke tube apparatus [11,14,15] that emulates thermoacoustic instabilities (TAI) in real-life combustors, which can lead to failure of mechanical structures, loud noise, and flow-reversal in the combustion system. From these perspectives, major contributions of the technical brief are delineated below.

- *Novelty of the algorithms*: The classifier is trained offline only on a single ensemble of time series data corresponding to an a priori known regime. Subsequently, other regimes are autonomously identified and learned as they are encountered online, which enable the algorithms to decide whether or not a block of time series belongs to one of the trained regimes.
- *Experimental validation*: The proposed algorithms have been tested and validated with time series of both synthetic data and experimental data to establish their feasibility for potential commercial applications.

2 Mathematical Theory

Before embarking on an explanation of the proposed algorithms, it is necessary to provide the background for construction of PFSA (see Sec. 2.1) [16,17] and D-Markov machines (see Sec. 2.2) [18].

Contributed by the Dynamic Systems Division of ASME for publication in the Journal of Dynamic Systems, Measurement, and Control. Manuscript received August 18, 2019; final manuscript received May 21, 2020; published online July 10, 2020. Assoc. Editor: Umesh Vaidya.

2.1 Probabilistic Finite State Automata. Time series of the measured signal is quantized and then symbolized as a symbol string. In this process, the signal space is partitioned into a finite number of cells, where the cardinality $|\Sigma|$ of the (symbol) alphabet Σ is identically equal to the number of cells. A symbol from the alphabet Σ is assigned to each (signal) value corresponding to the cell to which it belongs [19,20]; the details are reported in Ref. [18]. Thus, a symbol is associated with a data point at a given instant of time when the value of that data point is located in the particular cell corresponding to that symbol. The following definitions, which are available in standard literature (e.g., Refs. [16] and [18]), are recalled for completeness of the technical brief.

DEFINITION 1. *A finite state automaton (FSA) G, having a deterministic algebraic structure, is a triple* (Σ, Q, δ) *where:*

- Σ *is a (nonempty) finite alphabet, i.e., its cardinality* $|\Sigma|$ *is a positive integer.*
- Q *is a (nonempty) finite set of states, i.e., its cardinality* $|Q|$ *is a positive integer.*
- $\delta : Q \times \Sigma \to Q$ *is a state transition map.*

DEFINITION 2. *A symbol block, also called a word, is a finite length string of symbols belonging to the alphabet* Σ*, where the length of a word* $w \triangleq s_1 s_2 \cdots s_\ell$ *with every* $s_i \in \Sigma$ *is* $|w| = \ell$*, and the length of the empty word* ϵ *is* $|\epsilon| = 0$*. The parameters of FSA are extended as:*

- *The set of all words, constructed from symbols in* Σ *and including the empty word* ϵ*, is denoted as* $\Sigma^\star$*.*
- *The set of all words, whose suffix (respectively, prefix) is the word w, is denoted as* $\Sigma^\star w$ *(respectively,* $w\Sigma^\star$*).*
- *The set of all words of (finite) length* ℓ*, where* ℓ *is a positive integer, is denoted as* Σ^ℓ*.*

Remark 3. A symbol string (or word) is generated from a (finite length) time series by symbolization.

DEFINITION 4. *A PFSA K is a pair* (G, π)*, where:*

- *The deterministic FSA G is called the underlying FSA of the PFSA K.*
- *The probability map* $\pi : Q \times \Sigma \to [0, 1]$ *is called the morph function (also known as symbol generation probability function) that satisfies the condition:* $\sum_{\sigma \in \Sigma} \pi(q, \sigma) = 1$ *for all* $q \in Q$*.*
- *The* $(|Q| \times |\Sigma|)$ *morph matrix* Π*, which is converted into the* $(|Q||\Sigma| \times 1)$ *morph vector* ν *to serve as a feature in the sequel, is generated by the morph function* π*.*

Equivalently, a PFSA is a quadruple $K = (\Sigma, Q, \delta, \pi)$*.*

2.2 *D*-Markov Machines. The PFSA representation of a *D*-Markov machine generates symbol strings $\{s_1 s_2 \cdots s_\ell : \ell \in \mathbb{N}^+ \text{ and } s_j \in \Sigma\}$ on the underlying Markov process. In the construction of a *D*-Markov machine, it is assumed that the generation of the next symbol depends only on a *finite* history of at most *D* consecutive symbols, i.e., a symbol block of length not exceeding length *D*. A *D*-Markov machine [18] is defined as follows.

DEFINITION 5. *A D-Markov machine [16] is a PFSA in the sense of Definition 4 and it generates symbols that solely depend on the (most recent) history of at most D consecutive symbols, where the positive integer D is called the depth of the machine. Equivalently, a D-Markov machine is a statistically stationary stochastic process* $S = \cdots s_{-1} s_0 s_1 \cdots$*, where the probability of occurrence of a new symbol depends only on the last consecutive (at most) D symbols,* i.e.,

$$P[s_n | \cdots s_{n-D} \cdots s_{n-1}] = P[s_n | s_{n-D} \cdots s_{n-1}] \tag{1}$$

Consequently, for $w \in \Sigma^D$ *(see Definition 2), the equivalence class* $\Sigma^\star w$ *of all (finite length) words, whose suffix is w, is qualified to be a D-Markov state that is denoted as w.*

As the proposed *D*-Markov algorithms need a uniform dimension of the feature vectors, the concept of state-merging [18] has not been included. There are four primary parameters in the algorithms as enumerated below:

(1) *Alphabet size* ($|\Sigma|$): Larger is the alphabet size, more distinct are the different regimes, but more training data would be needed. There are several algorithms for selection of the optimal alphabet size (e.g., Ref. [21]); but, to demonstrate efficacy of the algorithms, an alphabet size $|\Sigma| = 16$ is chosen in the technical brief for the synthetic dataset and an alphabet size $|\Sigma| = 6$ for the experimental dataset. The choice of alphabet size is dependent primarily on how closely spaced the data are from each regime, with higher alphabet sizes typically yielding better regime separability (if needed).
(2) *Partitioning method*: While there are many data partitioning techniques, maximum entropy partitioning (MEP) [18–20], which is a commonly used partitioning technique, has been chosen in this technical brief.
Depth (D) in the D-Markov machine: Higher values of the positive integer *D* may lead to better results at the expense of increased computational time due to larger dimension of the space and need for more training. In this technical brief, $D = 1$ has been chosen in order to keep lower word lengths and smaller PFSAs, which lead to faster training and testing.
(3) *Choice of feature*: The feature needs to be one that best captures the nature (e.g., texture) of the signal. The morph vector ν (see Definition 4) has been chosen in this technical brief as the feature, because it is not only easily computed but also captures pertinent dynamics embedded in the signal.

3 The Proposed Algorithms

To address the problem of online discovery and classification of operational regimes, the proposed algorithms are first executed upon a time series at a given time epoch, not in its entirety, but in a windowed fashion with or without an overlap. The algorithms require only a single regime to be labeled, where a (human) expert is required to label a part of the ensemble of training time series data, which corresponds to a specific regime, called the *base regime*. Typically, this regime would represent the nominal operating condition of the physical process under consideration; however, it is not restricted to be so.

The algorithms have two learning phases: the first is the (offline) learning/training phase of the base regime and the second is the (online) discovery-classification phase for learning and classifying the remaining regimes.

3.1 Learning the Base Regime. A windowed segment of time series from the training data (e.g., corresponding to the base regime) is first analyzed. Each segment is symbolized by MEP [19] with a preset alphabet size, followed by construction of a PFSA[1] as described in Sec. 2. The boundaries of MEP are computed for every window. Training is done for assigned time series, where each time series possibly comprises of multiple windowed segments and the corresponding morph vector serves as the feature of each windowed time series, and is stored. After training, the mean of all morph vectors is taken as the centroid corresponding to the regime under consideration in the higher dimensional space, similar to what is done in *K*-means [1]. The Euclidean distance $||\bullet||_e$ of each of the training morph vectors from the base centroid is obtained as

[1]The PFSA codes used in this paper are developed in-house by the authors and are available at: https://github.com/Chandrachur92/PFSA

$$d_j^i = ||C_j - v_j^i||_e, \quad i = 1, \ldots, N_j \tag{2}$$

where v_j^i is the morph vector of segment i in regime j (= 1 for this phase), and C_j is the centroid of regime j defined as: $C_j \triangleq \frac{1}{N_j}\sum_{i=1}^{N_j} v_j^i$. It is noted that the segment number i runs from 1 to N_j, which is the number of windowed segments used to train regime j. A neighborhood of the base regime (i.e., $j = 1$) is generated about its centroid with radius ρ_1 that is computed by setting $j = 1$ in the following equation:

$$\rho_j = \frac{1}{N_j}\sum_{i=1}^{N_j} d_j^i + \gamma\sqrt{\frac{\sum_{i=1}^{N_j}\left(d_j^i\right)^2}{N_j - 1}} \tag{3}$$

where the first term on the right hand of Eq. (3) is the mean of all distances in regime j, and the second term is the standard deviation multiplied by a user-selected parameter γ. The rationale for choosing the specific structure in Eq. (3) is that the distribution of d_j^i is assumed to be nearly Gaussian, because the cumulative effects of many independent random variables in the construction of a PFSA tend to yield a Gaussian distribution. For a Gaussian distribution, the 4σ band contains almost 100% of the data and also ensures that obvious spurious outliers do not contaminate the estimate of the radius; the parameter $\gamma = 4$ is chosen in this technical brief. However, for a non-Gaussian (e.g., fat-tailed) distribution, γ should be appropriately chosen.

It is noted that, at the end of this primary training, the algorithms are only aware of the feature centroid and neighborhood corresponding only to the base regime and are unaware of any other regimes (e.g., anomalies or other operational states). At this point, the number of trained regimes is $T = 1$.

3.2 Online Discovery and Classification. The algorithms executes on a window of data, hereafter referred to as a data segment, from an unknown time series. The partitioning parameters of MEP are recomputed with the same alphabet size to construct a PFSA from the data segment and to compute the morph vector to serve as an extracted feature. If the morph vector is within the neighborhood of a known regime, the time-window is classified to belong to that regime; otherwise, the algorithms treat the data segment as belonging to a newly *discovered* regime.

The algorithms keep track of the number of segments that they receive corresponding to the new regime and store the feature vector for each data segment; it is noted that the segments need not be observed consecutively. As the algorithms *discover* a new regime, the feature vector (v_{new}^1) is taken to be the initial guess for the centroid of the new regime (C_{new}). However, there may not be enough data to estimate the radius of the neighborhood. To allow for larger initial search regions, the parameters, $\alpha_j > 1$, are defined as follows:

$$\alpha_j \triangleq \frac{\frac{1}{N_j}\sum_{i=1}^{N_j} d_j^i + \beta\sqrt{\frac{\sum_{i=1}^{N_j}\left(d_j^i\right)^2}{N_j - 1}}}{\frac{1}{N_j}\sum_{i=1}^{N_j} d_j^i - \beta\sqrt{\frac{\sum_{i=1}^{N_j}\left(d_j^i\right)^2}{N_j - 1}}}, \quad j = 1, 2, \ldots, T \tag{4}$$

where T is the number of trained regimes and the user-set parameter β determines how much of the symmetric tail of the probability density is removed. The value of the parameter $\beta = 1.5$ has been used for both the cases (i.e., synthetic data and Rijke tube data) in Sec. 4. For each trained regime, α_j is computed, and a single parameter α is obtained as the minimum of α_1 (corresponding to the base regime $j = 1$) and the average of α_j for all T trained regimes

$$\alpha \triangleq \min\left(\alpha_1, \frac{1}{T}\sum_{j=1}^{T}\alpha_j\right) \tag{5}$$

A restriction $\alpha_1 \geq (\alpha_1)_{\min}$ is imposed to ensure numerical stability, where the user-set lower bound is selected in this paper as $(\alpha_1)_{\min} = 1.5$.

Now, the yet unknown radius, (ρ_{new}), of a newly discovered regime is initially assigned a value computed as the mean of ρ_j's, over all T trained regimes, multiplied by the factor α (see Eq. (5)), as below:

$$\rho_{\text{new}} = \frac{\alpha}{T}\sum_{j=1}^{T}\rho_j \text{ as the first (initial) guess} \tag{6}$$

For the subsequent $(M_1 - 1)$ online training segments, (where $M_1 \geq 1$ is a user-defined parameter) the radius ρ_{new} is updated as

$$\rho_{\text{new}} \leftarrow \left(\frac{\rho_{\text{new}} N_{\text{new}} + \alpha\,||C_{\text{new}} - v_{\text{new}}^{N_{\text{new}}}||_e}{N_{\text{new}} + 1}\right) \tag{7}$$

where N_{new} is number of segments in the discovered regime; and $v_{\text{new}}^{N_{\text{new}}}$ is the observed morph vector from the N_{new}th segment that was classified to belong to the "new" discovered regime.

The procedure in Eqs. (6) and (7) allows for larger initial search regions until the algorithms yield a better estimate of the radius of the new regime. This is necessary because each regime may have a different variance and would need a different neighborhood for accurate classification. The updating of α in Eq. (5) ensures that, as the algorithms become more confident, it attempts to make less conservative (i.e., tighter) estimates of the neighborhood radius.

The next M_2 online training segments, where M_2 is also a user-set parameter, the radius ρ_{new} is updated as

$$\rho_{\text{new}} \leftarrow \frac{1}{N_{\text{new}}}\sum_{i=1}^{N_{\text{new}}} d_{\text{new}}^i + \gamma\sqrt{\frac{\sum_{i=1}^{N_{\text{new}}}\left(d_{\text{new}}^i\right)^2}{N_{\text{new}} - 1}} \tag{8}$$

For each of the above $(M_1 + M_2)$ online training segments, the centroid of the regime is shifted as $C_{\text{new}} \triangleq \frac{1}{N_{\text{new}}}\sum_{i=1}^{N_{\text{new}}} v_{\text{new}}^i$ as the mean of all observed feature vectors of that regime changes.

Once the number of training segments in a given regime reaches the set number of $M_1 + M_2$, the training for that regime is deemed completed and no further parameter updating occurs for that regime. The values of M_1 and M_2 largely determine how long the algorithms will consider a given detected regime as *untrained*. Low values of M_1 and M_2 may degrade the quality of training. On the other hand, high values of M_1, and M_2 would enhance learning in general; however, it may cause poor learning in the presence of an unknown regime (e.g., a rare event) occurring for a short duration, because of insufficient data.

In the event that a feature vector of an observed segment (v^{obs}) lies in the intersection of two or more yet untrained regimes, the segment is assigned to the regime, in which it has a higher probability of belonging to, as described below:

$$\text{classified regime} = \arg\max_j\left(\frac{||C_j - v^{\text{obs}}||_e}{\rho_j}\right) \tag{9}$$

After the classifier has seen most of the typical regimes, it can decide on one of these regimes or detect a new regime. These algorithms are capable of learning and augmenting the regime library, and the operation can be executed in real-time for simultaneous learning and classification.

The pseudo-code of Algorithm 1 describes the first phase of base regime training, which is repeatedly followed by the second phase of discovery and classification for each new segment of time series as described in the pseudo-code of Algorithm 2.

Remark 6. The proposed method is also capable of merging regimes when they are too close to each other. This task of merging is achieved by checking if the new centroid of an untrained regime lies within the neighborhoods of any trained regime. In that case, all morph vectors corresponding to the untrained regime are assigned to the trained regime and the new values of centroid and radius of the amalgamated regime is recomputed using Eqs. (4)–(8).

Algorithm 1 Base regime training from ensemble of data

Input: Time series dataset corresponding to the base regime
Initialization: (#segments in base regime dataset) N_1; (#untrained regimes) U = 0; (#trained regimes) T = 1
User-set parameters: β, γ, and $(\alpha_1)_{min}$

Output: *Base regime information*: Centroid C_1; Neighborhood radius ρ_1; (#Segments) N_1; Parameters α_1 and α

for $i = 1$ to N_1 **do**
  Symbolize time series segment i and generate PFSA
  Store morph vector v_1^i as the extracted feature
end for
$C_1 = \frac{1}{N_1}\sum_{i=1}^{N_1} v_1^i$
for $i = 1$ to N_1 **do**
$d_1^i = ||C_1 - v_1^i||_e$ % Euclidean norm
end for
$\rho_1 = \frac{1}{N_1}\sum_{i=1}^{N_1} d_1^i + \gamma\sqrt{\frac{\sum_{i=1}^{N_1}(d_1^i)^2}{N_1 - 1}}$
$\alpha_1 = \max\left((\alpha)_{min}, \frac{\frac{1}{N_1}\sum_{i=1}^{N_1} d_1^i + \beta\sqrt{\frac{\sum_{i=1}^{N_1}(d_1^i)^2}{N_1-1}}}{\frac{1}{N_1}\sum_{i=1}^{N_1} d_1^i - \beta\sqrt{\frac{\sum_{i=1}^{N_1}(d_1^i)^2}{N_1-1}}}\right)$
$\alpha = \alpha_1$
return C_1, ρ_1, N_1, α_1, α

Algorithm 2 Regime identification from time series window

Input: Time series data segment windowed from the process
Initialization: (# Untrained regimes) U; (# Trained regimes) T; Parameter α_1 generated in Algorithm 1; Parameter α generated in Algorithm 1 and possibly updated in a previous execution of Algorithm 2; (# Segments) N_j, Centroid C_j, and neighborhood radius ρ_j, $j = 1, \ldots, U + T$
User-set parameters: β, γ, M_1 and M_2

Output: Regime that the time series data belong to (i.e., the classified regime); (# Untrained regimes) U; (# Trained regimes) T; Centroid C_j and neighborhood radius ρ_j, $j = 1, \ldots, U + T$; Parameter α

Symbolize time series and generate D-Markov Machine
Extract morph vector v^{new} to serve as a feature
for $j = 1$ to $(T + U)$ **do**
  if $||C_j - v||_e \leq \rho_j$ **then**
    if $N_j > (M_1 + M_2)$ **then**
      Classified regime = j % as trained regime j
  else
    $C_j = \frac{1}{N_j+1}\left(\sum_{i=1}^{N_j} v_j^i + v^{new}\right)$
    $d_j^{N_j+1} = ||C_j - v^{new}||_e$ and $v_j^{N_j+1} = v^{new}$
    for $i = 1$ to N_j **do**
$d_j^i = ||C_j - v_j^i||_e$ % Euclidean norm
    end for
    if $N_j \leq M_1$ **then**
$\rho_j \leftarrow \frac{\rho_j N_j + \alpha\, d_j^{N_j+1}}{N_j+1}$
$N_j = N_j + 1$
    else if $M_1 < N_j < M_2$ **then**
$N_j = N_j + 1$
$\rho_j = \frac{1}{N_j}\sum_{i=1}^{N_j} d_j^i + \gamma\sqrt{\frac{\sum_{i=1}^{N_j}(d_j^i)^2}{N_j - 1}}$
    end if
    if $N_j \geq M_1 + M_2$ **then**
$T \leftarrow T + 1$ %#trained regime increased by 1
$U \leftarrow U - 1$ %#untrained regime decreased by 1
      for $\tilde{j} = 1$ to T **do**
$\alpha_{\tilde{j}} = \frac{\frac{1}{N_{\tilde{j}}}\sum_{i=1}^{N_{\tilde{j}}} d_{\tilde{j}}^i + \beta\sqrt{\frac{\sum_{i=1}^{N_{\tilde{j}}}(d_{\tilde{j}}^i)^2}{N_{\tilde{j}} - 1}}}{\frac{1}{N_{\tilde{j}}}\sum_{i=1}^{N_{\tilde{j}}} d_{\tilde{j}}^i - \beta\sqrt{\frac{\sum_{i=1}^{N_{\tilde{j}}}(d_{\tilde{j}}^i)^2}{N_{\tilde{j}} - 1}}}$
      end for
$\alpha = min\left(\alpha_1, \frac{1}{T}\sum_{\tilde{j}=1}^{T}\alpha_{\tilde{j}}\right)$
    end if
    Classified regime = j % as untrained regime j
  end if
else
  $N_{T+U+1} = 1$ $ZN_{T+U+1} = 1$; $C_{T+U+1} = v$; $\rho_{T+U+1} = \alpha \times \frac{1}{T}\sum_{j=1}^{T}\rho_j$
  Classified regime = $T + U + 1$ % as a new regime
  $U = U + 1$ %#untrained regime increased by 1
  end if
end for
return Classified regime

4 Results and Discussion

This section addresses testing and validation of the proposed algorithms, where time series data have been obtained from two sources. The first source is synthetically designed using the (unforced) Van der Pol equation [22], wherein the change-points are very firmly defined. The second source generates an ensemble of experimental data of pressure oscillations, collected from an electrically heated Rijke tube apparatus [11,23], where the operational regimes of the physical process are changed from stable to unstable and vice versa through various stages of transience.

4.1 Synthetic Data. The synthetic time series consist of data points of length 50,000 corresponding to an observation period of 10,000 s at a sampling frequency of 5 Hz. This ensemble of datasets is constructed from the standard unforced Van der Pol oscillator whose governing equation is given below:

$$\frac{d^2y}{dt^2} + \mu\left(y^2 - 1\right)\frac{dy}{dt} + y = 0 \quad \text{for} \quad \mu > 0 \tag{10}$$

The generated dataset comprises of four regimes, each corresponding to a distinct value of the damping parameter μ, where μ takes the values of 0.1, 1.0, 1.5, and 2.5. The change-points have been randomly selected for each time series that has at most nine

Table 1 Confusion matrix: regime classification using synthetic data

	Classified As				
True regime	Regime 1	Regime 2	Regime 3	Regime 4	Regime 5 (error)
Regime 1	89.14%	0.1%	3.66%	0.1%	7.00%
Regime 2	0.19%	84.27%	7.18%	0.23%	8.13%
Regime 3	0.06%	0.67%	93.11%	0.34%	5.82%
Regime 4	0.11%	0.11%	2.75%	93.95%	3.08%

sections, where each section has any one of the above regimes randomly assigned to it. Sixty such time series have been generated. This ensemble of time series serves as the ground truth for testing the proposed algorithms, because it is generated by a known algorithm with specified regimes and the associated change-points. In order to verify whether an algorithm is working as expected, the ensemble of time series is split into three parts: (i) the first part (30% of total data) for training the base regime, which is randomly chosen as any one of the above-mentioned four regimes, (ii) the second part (60% of total data) to learn the other regimes online, and (iii) the third part (the remaining 10% of total data) to test whether the regimes have been learned correctly by verification against the ground truth.

For conversion of the time series into online data, individual datasets are windowed, with the windowing occurring every 100 data points and each window having a size of 1000 data points. In this analysis, the PFSA has an alphabet size $|\Sigma| = 16$ and depth $D = 1$. Values of the parameters M_1 and M_2 have been chosen as 50 and 250, respectively. The average error has been computed over 20 different runs, where each of these runs is executed on a newly generated dataset, a new randomly selected base regime, and newly trained algorithm centroids and radii.

The results are presented in Table 1 as a confusion matrix, where the failure to correctly identify a classified regime (i.e., the algorithms erroneously deciding that the given data belongs to none of the four actual regimes) is called an error (regime 5), with the algorithms having an overall average error-rate below 10%. This is considered to be a low error because the assigned regimes are very close to each other in the feature space. While the algorithms could have identified more than four regimes in the first few trials, it has been capable of refining the results to the four true regimes by merging of regimes in subsequent runs.

4.2 Experimental Rijke Tube Data. This subsection demonstrates the efficacy of the proposed methodology by showing its capability to perform on an emulated real-life problem. To this end, the underlying algorithms have been tested and validated on experimental data of pressure time series from an electrically heated Rijke tube apparatus [11,23]. The Rijke tube [14] is a commonly used experimental apparatus that emulates the phenomena of TAI [15] encountered in real-life gas-turbine combustors. It is known that occurrence of TAI can be detrimental to the safe operation and health of a combustor. Thus, there is a need to be able to detect and identify the regime that the combustor is presently operating at and identify the transient regime, wherein the combustor is expected to move either from a stable regime to an unstable regime or vice versa. Regime identification is also the first step to perform monitoring and control on the combustor system. Several techniques have been suggested for detection of TAI using different algorithms (e.g., Refs. [5], [11], [24], and [25] to name a few). However, as mentioned in Sec. 1, all these techniques need the assistance of a human expert

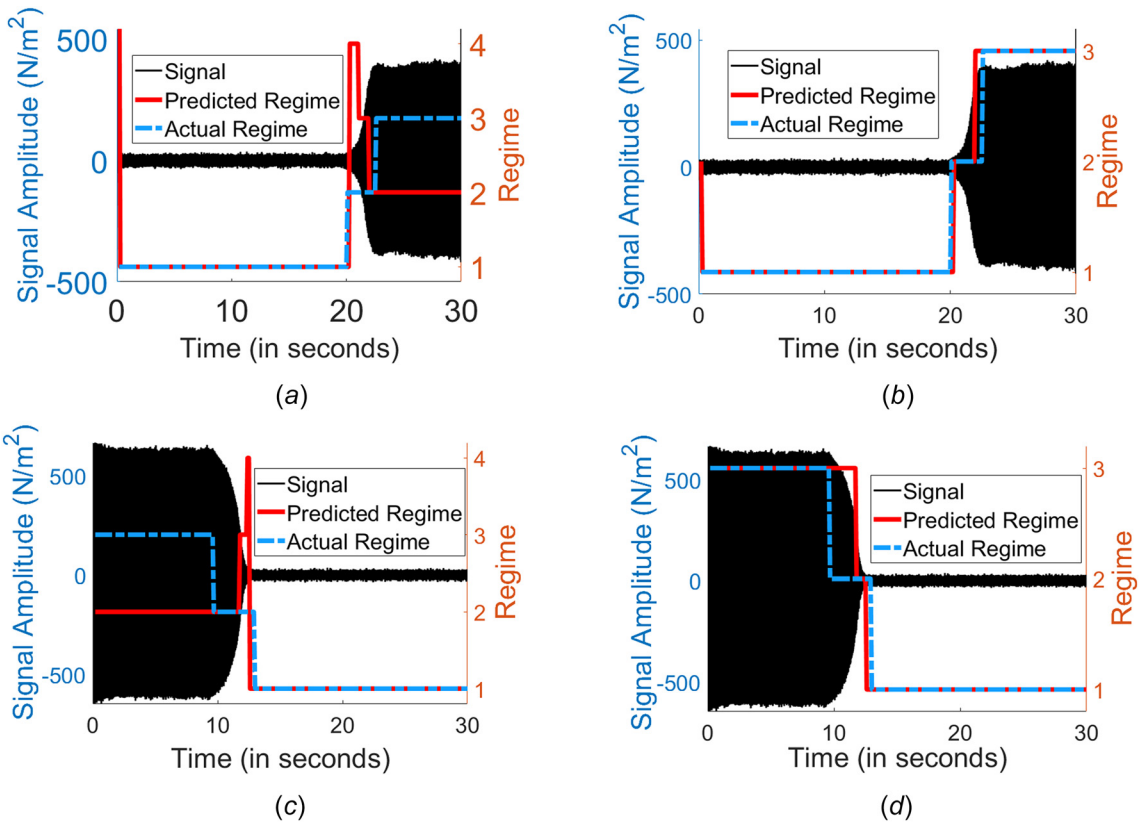

Fig. 1 Detection and classification from Rijke tube data: (*a*) time series 1 with separate transient regimes, (*b*) time series 1 with amalgamated transient regimes, (*c*) time series 2 with separate transient regimes, and (*d*) time series 2 with amalgamated transient regimes

Table 2 Confusion matrix: regime classification using rijke tube data

Truly	Classified stable	Classified transient	Classified unstable
Stable	99.73%	0.27%	0%
Transient	11.21%	57.76%	31.03%
Unstable	0%	0%	100%

for classifying the training samples and to create appropriate labels.

The Rijke tube apparatus, described by Mondal et al. [11], has been used to generate transient signals described above. In this work, the detection algorithms are trained for the nominal stable condition as the base regime. Using the methodology described above, other transient and unstable regimes are identified as shown in Fig. 1.

The ensemble of data consists of a total 40 time series that were collected at a sampling rate of 8192 Hz and subsequently filtered with a cut-off frequency of 40 Hz [11]. Then, data windowing has been done at ~10 Hz, i.e., the length of each data window is ~800. The time series is split into three groups, 20% of total data for training the base regime (i.e., regime 1), 70% for online identification, and the remaining 10% for testing.

The D-Markov machines (see Sec. 2.2) have been constructed from the ensemble of above windowed data with an alphabet size $|\Sigma| = 6$ and depth $D = 1$. The parameters M_1 and M_2 (see Sec. 3.2) have been chosen as 20 and 180, respectively; the rationale for choosing these low values is that transience occurs over a very short time span and so the algorithms need to train quickly (e.g., within 36 datasets), and the reduced values of M_1 and M_2 achieve the goal. In this process, there is no available ground truth as to when the transient state begins and ends. Various researchers have used different approximations for deciding the change-points for similar events. For example, a cumulative sum approach has been commonly used to approximately determine the change-points ([11]), for which the regimes identified by the algorithm have been cross-checked.

The four plates in Fig. 1 show regime predictions for two sets of time series of experimental data. The two left-hand plates, (*a*) and (*c*), in Fig. 1 show the actual regimes (i.e., the a priori estimated ground truth) for the time series 1 and 2, respectively. The objective here is to identify the change of regime from stable to transient and then to unstable. In this study, the algorithms also identify a transient regime that is not apparent otherwise. In the right-hand images in the plates, (*b*) and (*d*), of Fig. 1, the two predicted transient regimes have been amalgamated into a single transient regime for ease of comparison with the estimated ground truth, where it is seen that the ground truth and the predictions match closely, but not perfectly, because the ground truth itself may be flawed. However, the algorithms are capable of correctly detecting the changes and still have good accuracy, especially in identifying the learned unstable regime. This is seen in the confusion matrix (computed as an average over four test series across 20 repeated runs) in Table 2. Figure 1 also shows that the algorithms identify the most important change (i.e., from stable to transient states nearly perfectly). This capability is very useful for initiating a control action to prevent instability, because the transient regime is where the active controller should quench the pressure oscillations as early as possible.

5 Summary, Conclusions, and Future Work

This technical brief has developed a (partially) unsupervised methodology for online identification and classification of operational regimes in dynamical systems. An objective here is to discover new regimes in an online fashion, based on the a priori knowledge of only a single selected regime. Although the proposed method constructs centroids and regime-radii (in the feature space) representing each regime, it is not iterative like the standard K-means [1,2] and thus can be used for online discovery. As a new time series from an unknown regime is observed, the algorithm is updated online (without the need to iterate) until the regime model converges. The underlying algorithms have been tested and validated with synthetic data from an unforced Van der Pol equation and also with experimental data from an electrically heated Rijke tube apparatus that emulates pressure oscillations in real-life combustors.

While there are many areas of theoretical and experimental research, which must be investigated before the proposed methodology can be implemented in real-life applications, the following topics of future research are suggested.

- Enhancement of the algorithms to accommodate smaller data window lengths, i.e., faster detection and identification of regimes.
- Modification of the algorithms for guaranteed robustness to noise and uncertainties.
- Usage of other machine learning tools (e.g., hidden Markov models, Gaussian process modeling or neural networks [1,2]) as alternatives to symbolic dynamics.
- Verification of the algorithms' efficacy to be able to classify other standard dynamical systems, especially chaotic systems like the forced Van der Pol oscillator and the Rössler dynamical system.
- Autonomous learning of the algorithms' hyper-parameters, namely $|\Sigma|$, γ, β, $(\alpha_1)_{\min}$, M_1, and M_2.
- Further investigation within a sensor fusion framework to improve identification of unforeseen regimes that may emerge under rare events.

Acknowledgment

The authors are grateful to Dr. Sudeepta Mondal at Penn State, who kindly provided the experimental data. The reported work has been supported in part by the U.S. Air Force Office of Scientific Research (AFOSR) under Grant Nos. FA9550-15-1-0400 and FA9550-18-1-0135 in the area of dynamic data-driven application systems (DDDAS). The first author is also thankful to Indo-U.S. Science and Technology Forum (IUSSTF) for granting him the Research Internship for Science and Engineering (RISE) scholarship. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

Funding Data

- U.S. Air Force Office of Scientific Research (AFOSR) (Grant Nos. FA9550-15-1-0400 and 558 FA9550-18-1-0135; Funder ID: 10.13039/100000181).

References

[1] Bishop, C., 2006, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag, Berlin.

[2] Murphy, K., 2012, *Machine Learning: A Probabilistic Perspective*, The MIT Press, Cambridge, MA.

[3] Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E., 2017, "The Great Time Series Classification Bake Off: A Review and Experimental Evaluation of Recent Algorithmic Advances," Data Min. Knowl. Discovery, **31**(3), pp. 606–660.

[4] Sugimura, H., and Matsumoto, K., 2011, "Classification System for Time Series Data Based on Feature Pattern Extraction," IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, AK, Oct. 9–12, pp. 1340–1345.

[5] Sarkar, S., Chakravarthy, S., Ramanan, V., and Ray, A., 2016, "Dynamic Data-Driven Prediction of Instability in a Swirl-Stabilized Combustor," Int. J. Spray Combust. Dyn., **8**(4), pp. 235–253.

[6] Ghalyan, N., and Ray, A., 2020, "Symbolic Time Series Analysis for Anomaly Detection in Measure-Invariant Ergodic Systems," ASME J. Dyn. Syst. Meas. Control, **142**(6), p. 061003.

[7] Jha, D., Virani, N., Reimann, J., Srivastav, A., and Ray, A., 2018, "Symbolic Analysis-Based Reduced Order Markov Modeling of Time Series Data," Signal Process., **149**, pp. 68–81.

[8] Li, Y., Jha, D. K., Ray, A., and Wettergren, T. A., 2018, "Information-Theoretic Performance Analysis of Sensor Networks Via Markov Modeling of Time Series Data," IEEE Trans. Cybern., **48**(6), pp. 1898–1909.
[9] Wang, Z., Yan, W., and Oates, T., 2017, "Time Series Classification From Scratch With Deep Neural Networks: A Strong Baseline," International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, May 14–19, pp. 1578–1585.
[10] Husken, M., and Stagge, P., 2003, "Recurrent Neural Networks for Time Series Classification," Neurocomputing, **50**, pp. 223–235.
[11] Mondal, S., Ghalyan, N., Ray, A., and Mukhopadhyay, A., 2019, "Early Detection of Thermoacoustic Instabilities Using Hidden Markov Models," Combust. Sci. Technol., **191**(8), pp. 1309–1328.
[12] Lim, C., and Harrison, R., 2003, "Online Pattern Classification With Multiple Neural Network Systems: An Experimental Study," IEEE Trans. Syst., Man, Cybern., Part C, **33**(2), pp. 235–247.
[13] Huang, X., Ye, Y., Xiong, L., Lau, R., Jiang, N., and Wang, S., 2016, "Time Series k-Means: A New k-Means Type Smooth Subspace Clustering for Time Series Data," Inf. Sci., **367–368**, pp. 1–13.
[14] Rijke, P. L., 1859, "Notiz Über Eine Neue Art, Die in Einer an Beiden Enden Offenen Röhre Enthaltene Luft in Schwingungen zu Versetzen," Ann. Phys., **183**(6), pp. 339–343.
[15] Matveev, K., 2003, "Thermoacoustic Instabilities in the Rijke Tube: Experiments and Modeling," Ph.D. thesis, California Institute of Technology, Pasadena, CA.
[16] Ray, A., 2004, "Symbolic Dynamic Analysis of Complex Systems for Anomaly Detection," Signal Process., **84**(7), pp. 1115–1130.
[17] Gupta, S., and Ray, A., 2007, "Symbolic Dynamics Filtering for Data-Driven Pattern Recognition," *Pattern Recognition: Theory and Application*, Nova Science Publishers, Hauppage, NY, pp. 17–71.
[18] Mukherjee, K., and Ray, A., 2014, "State Splitting and Merging in Probabilistic Finite State Automata for Signal Representation and Analysis," Signal Process., **104**, pp. 105–119.
[19] Rajagopalan, V., and Ray, A., 2006, "Symbolic Time Series Analysis Via Wavelet-Based Partitioning," Signal Process., **86**(11), pp. 3309–3320.
[20] Subbu, A., and Ray, A., 2008, "Space Partitioning Via Hilbert Transform for Symbolic Time Series Analysis," Appl. Phys. Lett., **92**(8), p. 084107.
[21] Sarkar, S., Chattopdhyay, P., and Ray, A., 2016, "Symbolization of Dynamic Data-Driven Systems for Signal Representation," Signal, Image Video Process., **10**(8), pp. 1535–1542.
[22] Cartwbight, M. L., 1960, "Balthazar Van Der Pol," J. London Math. Soc., **s1-35**(3), pp. 367–376.
[23] Bhattacharya, C., Mondal, S., Ray, A., and Mukhopadhyay, A., 2020, "Reduced-Order Modeling of Thermoacoustic Instabilities in a Two-Heater Rijke Tube," Combust. Theory and Model., **23**(3), pp. 530–548.
[24] Gotoda, H., Amano, M., Miyano, T., Ikawa, T., Maki, K., and Tachibana, S., 2012, "Characterization of Complexities in Combustion Instability in a Lean Premixed Gas-Turbine Model Combustor," Chaos, **22**(4), p. 043128.
[25] Nair, V., and Sujith, R., 2014, "Multifractality in Combustion Noise: Predicting an Impending Combustion Instability," J. Fluid Mech., **747**, pp. 635–655.

Mechanical Systems and Signal Processing 147 (2021) 107109

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

HMM conditional-likelihood based change detection with strict delay tolerance

David J. Miller [a,*], Najah F. Ghalyan [b,c], Sudeepta Mondal [c], Asok Ray [d]

[a] School of Electrical Engineering and Computer Science, Pennsylvania State University, University Park, PA 16802, USA
[b] Department of Mechanical Engineering, The University of Kerbala, Kerbala 56001, Iraq
[c] Department of Mechanical Engineering, Pennsylvania State University, University Park, PA 16802 USA
[d] Departments of Mechanical Engineering and Mathematics, Pennsylvania State University, University Park, PA 16802, USA

ARTICLE INFO

Article history:
Received 13 July 2019
Received in revised form 1 June 2020
Accepted 28 June 2020

Keywords:
Anomaly detection
Change point detection
Hidden Markov modeling
Conditional likelihood
Forward recursion
Delay tolerance
Fatigue failure
Combustion instability

ABSTRACT

Hidden Markov models (HMMs) have been widely used for anomaly and change point detection due to their representation power and computational efficiency in capturing statistical dependencies in time series. However, often information is integrated over relatively long observation windows, with detections made when the observed sequence's likelihood under the (null) HMM deviates significantly from its typical range. Three related limitations are: i) use of long windows entails large decision delay, which may e.g. fail to prevent machine failure/damage; ii) typical approaches do not narrowly identify an interval within which the change point occurred. Such information could be useful e.g. for process control, where one wants to know how long it takes for control inputs to induce desired change points; iii) The decision statistic is usually the likelihood of the data in the current window, without consideration of past observations. This is suboptimal – this likelihood should be conditioned on past observations to optimally account for statistical dependency in the time series. In this paper, we propose a framework for change point detection which overcomes all of these limitations: i) it applies a standard HMM Forward recursion, but used to evaluate the likelihood of an observation subsequence conditioned on the subsequence's entire past. This approach is used to efficiently evaluate the conditional likelihoods of all intervals of fixed length (hence with fixed delay, d), until a change point is first detected. Here d is a design parameter whose proper value (needed to have a quick response/mitigate damage) may be known for a given application domain; ii) the algorithm narrowly estimates the interval within which a detected change point lies; iii) we propose a novel performance criterion well-matched to low-delay, narrowly localized change point detection – the true detection interval rate (TDIR) – and also evaluate the false positive rate (FPR) and the bias and variance of the estimated change point, all as a function of d. The proposed method is shown to outperform a CUSUM algorithm, symbolic time series analysis (STSA) methods, and a standard HMM method (evaluating the unconditioned likelihood) for instability onset in combustion systems and fatigue failure initiation in a material.

* Corresponding author.
E-mail address: djmiller@engr.psu.edu (D.J. Miller).

https://doi.org/10.1016/j.ymssp.2020.107109
0888-3270/

1. Introduction

1.1. Detection literature review

Change point detection [1] has important applications e.g. to speech recognition, radar systems, condition-based maintenance, fatigue failure detection, geology signals, and even to DNA analysis and malware detection (see, e.g., [2,3] and references therein). If the distribution of observations before a change point occurs (the null distribution) and the distribution after a change point are both known, the problem can be treated e.g. as a standard (two-class) hypothesis testing problem. However, as is more realistic in practice, the distribution after change is typically unknown. In this case, change point detection involves choosing a cost function that measures goodness-of-fit of the generated time series to the null model, and identifying signal segments for which the goodness of fit falls below a threshold. One of the standard methods is the cumulative sum (CUSUM) technique, developed by Page [4], which has been widely used [5–7]. CUSUM is a sequential method which involves the calculation of a cumulative statistic, thresholded to decide whether a change has occurred or not. The decision threshold can be chosen to fix the false alarm rate based on a collection of time series for which it is known that no change point has occurred. In another approach, Bai and Perron [8] considered a time series generated by a linear regression model that undergoes multiple structural changes (breaks) at unknown times with the goodness of fit the sum of squared residuals between the regression model outputs and the observed time series. Later, in [9] they addressed the problem of estimating the change points, introducing a computationally-efficient algorithm that finds the global minimum of the sum of squared residuals using dynamic programming. Bai [10] also proposed a likelihood-ratio model that can detect multiple structural changes in (generally non-stationary) regression models based on hypothesis testing. The null hypothesis is that there are l (known or estimated) breaks in the time series, while the alternative hypothesis is that there are $l+1$ breaks. When $l=0$ a single break/change point is sought.

The approach in [9] considers the entire time series in making change point detections and thus imposes no constraint on the amount of allowed delay in making detections. In many applications, detection needs to be achieved with low delay and/or using short observation windows, e.g. in order to trigger machine damage mitigation. If the true change point occurs at time τ, detection should occur using observations only up to time $\tau+d$, where d, a delay tolerance design parameter, is chosen sufficiently small that detection can result in e.g. expedient damage mitigation. The maximum tolerable value of d may in fact be known for a given application based on past experience [11]. While some detection algorithms such as CUSUM possess optimality properties in minimizing the expected detection delay (given a fixed false positive detection rate) [6,12], this is not the same as the strict delay requirement (d) considered here – achieving an average detection delay of d, or even a value significantly below d, does not ensure that the probability of detection delay exceeding d is small. Moreover, such optimality results assume something is known about the distribution after change, e.g. that the distribution parametric form is the same, post-change, but with a change in the mean parameter, or the variance parameter. More generally, nothing is known about the post-change distribution. Moreover, since d may be very small, a generalized likelihood ratio detection framework, where one estimates the putative distribution post-change and uses it in a likelihood ratio test, may not be feasible, since there will in general be too few observations (d) post-change to accurately estimate the post-change distribution.

Various scenarios are considered in the change point detection literature. First, as aforementioned, one may have knowledge of the observed data distribution both before and after a change has occurred [6]. We do not assume such information here. Second, there may be at most one change point or possibly multiple change points (each possibly to different states, with different data distributions). In the latter case, if there is no delay requirement, one can view the problem as signal segmentation given the whole observed time series [6], as is done in [9]. In [11], the authors distinguish a quickest change detection (QCD) problem and a transient change detection (TCD) problem. In the former, the duration of the change state is infinite, whereas in the latter, the duration of the change state is finite, before a return to the null/normal state or perhaps to a failure state. In this latter (TCD) problem setting, for particular applications, strict detection delay tolerance is needed, while strict delay tolerance is not usually considered in the QCD setting. Some works that establish theoretically optimal detectors, minimizing average detection delay, again based on knowledge of what statistics are changing, include [13,14]. In this work, we address detection problems where the change state duration may be finite or infinite; regardless, we impose strict delay tolerance in order to ensure that true detections will allow damage mitigation. More specifically, we consider a non-Bayesian scenario[1] that well-captures certain machine/process failure for condition-based maintenance applications: the machine is first operating in a normal state and may continue to do so for its entire operating cycle. Alternatively, at some point, there may be a change to an abnormal state. This abnormal state may itself be a system fault/failure state or it may be a transient state prior to entering fault/failure. Either way, there is no transition back to the normal state. The former case is a QCD problem and the latter essentially a TCD one because the duration of the (transient) change state is finite.[2] Irrespective, strict detection delay tolerance may be needed to ensure that damage mitigation (or recovery of normal state) is possible.

[1] The change point treated as deterministic but unknown, i.e. even if the change point is random we have no knowledge of its distribution.

[2] It may be possible to force the system back to the normal state if a proper control action is made in a timely fashion (although in this paper we are not addressing such control actions, and therefore in our case the final transition is to a failure state, not back to the normal state).

1.2. STSA-based detection

Symbolic time series analysis (STSA) has been efficiently used for anomaly detection [15–19]. In STSA, the time series is converted to a symbol sequence from a finite-cardinality alphabet $\mathscr{A}$. The resulting discrete-valued sequence is modeled as a Probabilistic Finite State Automata [20] (PFSA) that mirrors (tracks) the original discrete-time dynamical system [21]. The PFSA states are concatenations of symbols from $\mathscr{A}$. Anomaly detection using STSA is done by first training a null PFSA on data generated from the nominal phase. The resulting states' stationary probability vector is used as a null phase "reference". Once new data arrives, a new PFSA is constructed, whose states' stationary probability vector is computed and compared with that of the null. Based on the Kullback–Leibler "distance" between these two vectors, an anomaly is declared if this distance exceeds a user-defined threshold. One main drawback of STSA anomaly detection is that it discards information in the initial discretization (quantization) step that yields a (finite cardinality) symbol sequence.

1.3. HMM-based detection

Alternatively, Hidden Markov models (HMMs) have been widely used for both change point and anomaly detection (AD) in diverse applications (see e.g. [22–35]). In contrast to STSA, HMMs exploit a latent discrete symbol sequence representation that does not require (hard, information-lossy) quantization of the time series. Moreover, in HMMs one sums over all possible discrete state sequences, in evaluating the log-likelihood fit of the observed data to the (null) model, unlike in STSA – an HMM's joint probability (or joint likelihood) for an observation sequence $x_{n:n+M} = (x_n, x_{n+1}, \ldots, x_{n+M})$, starting at time n, is found by marginalizing over the set of all possible hidden state sequences. This computation is made efficiently via the iterative Forward algorithm [36], which can be used to assign likelihoods to observation sequences in an on-line manner, as each observation arrives. In an AD setting the (null) HMM model is trained using observed time series known to represent normal behavior (using the same (null phase) data used to estimate a null STSA model). Then, when the HMM is applied operationally, if a change point occurs within an observation sequence, the likelihood under the HMM of an observed subsequence that contains the change point is expected to deviate significantly from the typical likelihood [37], once enough observations following the change point are included in the subsequence.[3] Based on a properly chosen threshold (fixed e.g. to a specified false detection rate), one can decide whether change has occurred within a given subsequence. Using HMM inference to narrow down the change point in a long time series to within a small interval requires computing the joint-likelihoods of many possible observation subsequences. However, these many evaluations can be efficiently made by exploiting the standard HMM Forward recursion. Therefore, HMMs have been efficiently used for anomaly detection with the observation subsequence likelihood the decision statistic [22,38].

1.4. Low-delay detection

In this paper, we introduce a detection algorithm specifically designed to make strictly low-delay detections and to accurately identify a narrow interval within which the true change point is estimated to lie. The algorithm exploits the standard Forward recursion for HMM inference to successively evaluate the likelihoods of the observation subsequences $x_{n:n+d-1} = (x_n, \ldots, x_{n+d-1})$, conditioned on $(x_1, x_2, \ldots, x_{n-1}), n = 1, 2, \ldots$, until a change point is detected, where d is a detection delay parameter that controls the subsequence size and which is user-chosen, based on the delay tolerance of the application. Note that there is a tradeoff in the choice of d – if d is very small, only very low delay detections are tolerated, but the true detection rate may be too low (due to an insufficient number of post-change samples in the window). On the other hand, choosing larger d tends to increase the true detection rate, but may introduce an unacceptable delay in making detections. Note that past works such as [25–28,34,35] use HHMs with a sliding window for detection. However they do not condition on the past, in evaluating an observation subsequence's likelihood, for detection/inference purposes. In Appendix A, we theoretically support our use of the conditional likelihood for detection by proving that use of the true Bayes class posterior, conditioning on all available observations, is optimal not only in the well-known sense of minimum expected classification error but also in the sense of maximum true detection rate given a fixed false positive rate. Moreover, we prove that the true detection rate for such a detector, given fixed false positive rate, is non-decreasing as d is increased. Note also that evaluating likelihoods for a fixed window size, d, rather than for a growing window starting at the beginning of the time series, is expected to yield both quicker detection (many "normal" samples will "contaminate" a growing window's anomaly decision statistic, which should lead to delayed detections) and narrow identification of the change point interval.

1.5. Novel performance criteria

Furthermore, for our strictly low-delay setting, care is needed in defining detection events. Specifically, suppose the true change point occurs at time t_c. Then, we define a true detection interval event as one wherein the change is detected at a time t (based on observations up to t) such that $t \geqslant t_c$ and $t \leqslant t_c + d$, i.e. detections that are either too premature or too late

[3] Note that this use of a time-domain window of observations is commonly applied, both for detection as well as for estimation, i.e. Wiener filter smoothing, to estimate one random process given a window of observations from a statistically related process.

are not considered to be true. Also, consider a time series (corresponding to a machine cycle) that does not contain a change point. If a change point is detected anywhere in this time series (which is finite-length for a given machine's cycle, but may still be quite long), this we consider to be a false positive detection event. This is in contrast to a definition of false positives that includes events involving time series that do contain a change point, but where the detection is made too prematurely, or too late. Our false positive rate is meaningful in real applications because detection of even a single change may lead to a (costly) interruption of the machine's cycle, which is warranted only if the detection is likely to be that of a true change point.

Accordingly, we propose to evaluate several criteria that give a rich characterization of low-delay tolerant change point detection performance. Specifically, as a function of delay tolerance, d, we propose to evaluate: 1) bias of the estimated change point; 2) variance of the estimated change point; 3) We define the event that an estimated interval contains the true change point. Accordingly, we then evaluate a novel criterion proposed here, the true detection interval rate (TDIR), the probability that the true change point lies in the estimated interval. TDIR is a more demanding measure of performance than the true detection rate (TDR), which for successful detection would simply require a detection to be made after the change point has occurred; 4) Finally, we evaluate the false positive rate (FPR), the probability that a detection is made when in fact there is no change point in the given time series.

Contributions of the paper: The main contributions of the paper are:

- Detection algorithm: A novel HMM-based detection algorithm is developed that makes strictly low-delay change point detections and identifies a narrow interval within which the true change point is estimated to lie. The algorithm applies the standard HMM Forward recursion, but to evaluate the conditional likelihoods of all successive observation subsequences of length d, given the entire past of the time series relative to the subsequence, with the algorithm terminated either when a detection is first made or when the observation sequence has been exhausted with no detection made.
- Detection evaluation criteria: Criteria for evaluating performance in our strict delay setting, with at most one change point, are proposed, including the novel TDIR, FPR, and bias and variance in estimation of the change point, all as a function of delay tolerance (d) and the associated observation window length (also d).
- Evaluation on real-life application domains: Performance of the proposed algorithm is experimentally validated and compared with a CUSUM, STSA, and standard HMM detection techniques [38,22] on two different applications: detection of combustion instability onset and detection of fatigue failure start in polycrystalline alloys. The results for both applications show excellent performance of the proposed scheme to detect and estimate the change point and to define a narrow interval within which the change point lies. Substantial performance gains, with respect to all four criteria, are achieved compared with the abovementioned detection methods.

Organization of the paper: Section 2 introduces an algorithm for change point detection and performance criteria suitable for low-delay detection. Section 3 experimentally validates the proposed scheme in comparison with several popular methods. Section 4 summarizes and concludes the paper along with recommendations for future research. Appendix A presents a proof that theoretically supports our use of the conditional likelihood for detection and that detection performance improves with increasing d.

2. A conditional likelihood based HMM change point detection algorithm

2.1. Conditional likelihood calculation

Consider a null first-order HMM (learned using the standard Baum-Welch re-estimation algorithm for HMMs [39] based on a null time series, containing "normal" observations with no change point), described by $\Lambda = \{\{\pi_i\}, \{a_{j/i}\}, \{b_j(x)\}\}$, where $\{\pi_i\}$ is the initial state probability vector, $\{a_{j/i}\}$ is the state-transition probability matrix, and $\{b_j(x)\}$ are the state-conditional observation densities. Consider a new time series $x_{1:N} = (x_1, \ldots, x_N)$ being operationally interrogated for possible detection of a change point. Let $s_{1:N} = \{s_1, s_2, \ldots, s_N\}$ be a particular hidden state sequence realization that could have been used to generate the given observed time series, under the null HMM model. Recall the standard Forward variable for HMMs [39], i.e. $\alpha_n[i] = P[x_{1:n}, S_n = i; \Lambda], i = 1, \ldots, L, L$ the number of HMM states. Here, S_n is a random variable representing the state at time n and i is its (state) realization. These Forward variables can be recursively computed and used to evaluate the joint likelihood of an observation sequence, as follows [39][4]:

- Initialization:

$$\alpha_1[i] = \pi_i b_i(x_1), \ i = 1, \ldots, L \tag{1}$$

- Recursion:

[4] To avoid numerical underflow, we apply the scaling procedure for the Forward algorithm suggested in [39], which does not introduce any loss in precision, in computing the log-likelihood.

$$\alpha_{n+1}[j] = \left[\sum_{i=1}^{L} \alpha_n[i] a_{j/i}\right] b_j(x_{n+1}) \quad (2)$$
$$j = 1, \ldots, L, \; n = 1, \ldots, N-1$$

- Termination:

$$P[x_1, x_2, \ldots, x_N; \Lambda] = \sum_{i=1}^{L} \alpha_N(i) \quad (3)$$

Now consider the potential application of this Forward algorithm to detection of a change point within a given subsequence $x_{n:n+d-1}$. An approach that has been commonly applied in prior works [38,22] is to make the approximation that the time series in fact begins at time n. Thus, the Forward recursion above is run, but with the initialization step replaced by: $\hat{\alpha}_n[i] = \pi_i b_i(x_n), i = 1, \ldots, L$. Here, we use $\hat{}$ to denote the fact that an approximate Forward variable is being computed. The result of running the Forward recursion starting from this approximate initialization at time n is an approximation of the joint likelihood: $\widehat{P}[x_{n:n+d-1}; \Lambda] = \sum_{i=1}^{L} \hat{\alpha}_{n+d-1}[i]$. Use of this in making detections will be referred to as standard HMM inference for detection.

There are in fact two limitations of this standard approach. First, $\widehat{P}[x_{n:n+d-1}; \Lambda]$ is clearly only an approximation of the joint likelihood – the true joint likelihood can only be obtained by marginalizing out the random variables $X_1, \ldots, X_{n-1}$ from $P[X_1, \ldots, X_{n-1}, x_n, x_{n+1}, \ldots, x_{n+d-1}; \Lambda]$. In general, this is analytically intractable. Second, it would in fact be suboptimal to perform such marginalization (see Appendix A). Rather, it is optimal to account for the observed past by conditioning on it, i.e. evaluating the conditional likelihood $P[x_{n:n+d-1} | x_{1:n-1}; \Lambda]$. This conditional likelihood can in fact be efficiently evaluated. Note in particular that

$$P[x_{n:n+d-1} | x_{1:n-1}; \Lambda] = \frac{P[x_{1:n+d-1}; \Lambda]}{P[x_{1:n-1}; \Lambda]} = \frac{\sum_{i=1}^{L} \alpha_{n+d-1}[i]}{\sum_{i=1}^{L} \alpha_{n-1}[i]}, \quad (4)$$

The latter equality is achieved simply by realizing that the joint likelihoods in the numerator and denominator are efficiently computed by running the Forward HMM recursion described above in (1)–(3), i.e. starting from the beginning of the time series, and terminating at the appropriate times for the numerator and denominator likelihood calculations, respectively. We have seen little prior work using the conditional likelihood for HMM-based inference and HMM-based anomaly detection.

2.2. Detection algorithm based on the conditional likelihood

Consider a time series $x_{1:N}$, and assume there is at most one change point. Such an assumption is reasonable in many real-life applications, such as fault and machine failure detection. In these examples, it is valuable to be able, with low delay, to detect the beginning of the change from nominal behavior so as to make a proper action to avoid further consequences of such change. Therefore, we want to detect the start of the change, which corresponds to a single time point within the time series.

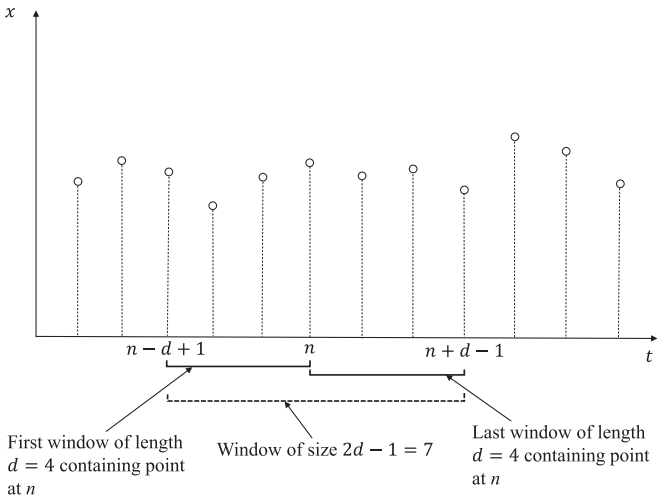

Fig. 1. Schematic representation of the sliding window in Algorithm 1, illustrating d subintervals of length d containing a given candidate change point, n.

To do so for allowed delay tolerance d, we consider intervals of length d that contain a candidate change time point, n. There are d such intervals, as indicated in Fig. 1. We evaluate the conditional log-likelihood of each such interval, using (4), and make a detection at time point n if the minimum absolute deviation from a (null) reference log-likelihood value, over all the d intervals, exceeds a threshold. Thresholding the minimum deviation is a conservative detection rule, with the resulting algorithm called HMM$^{(2)}$-min. At the other extreme, an aggressive detection rule would be to threshold the maximum of the d deviations, with the resulting algorithm called HMM$^{(2)}$-max. Both such detection rules are in fact evaluated in the sequel. If a detection is made, the detection interval is chosen as $[n-d+1, n+d-1]$ and the change point itself is estimated as n. If no detection is made, the window (in Fig. 1), centered on n, is slid one position to the right, with the new candidate change point as $n+1$. Pseudocode for our detection procedure is given below in Algorithm 1.

Algorithm 1 Proposed Detection Approach

Input: Time series $x_{1:N}$, a null HMM Λ, delay tolerance d, null (reference) normalized (by d) log-likelihood $\bar{L}$, and threshold δ.

Output: Either a "no change point" decision ($z_d = 0$) or a "change point detected" decision ($z_d = 1$) and the estimated change point $\hat{n}$, which determines the estimated interval $[\hat{n}-d+1, \hat{n}+d-1]$.

1: $z_d \leftarrow 0$, $n \leftarrow d$;
2: Run the Forward recursion to compute $\alpha_1[i], \ldots, \alpha_{2d-2}[i], i = 1, \ldots, L$.
3: Set $\sum_{i=1}^{L} \alpha_0[i] = 1$.
4: **do**
5: Take one Forward recursion step to compute $\alpha_{n+d-1}[i],\ i = 1, \ldots, L$.
6: Compute $\log P\left[x_{n-d+1+j:n+j} | x_{1:n-d+j}; \Lambda\right] = \log\left(\frac{\sum_{i=1}^{L} \alpha_{n+j}[i]}{\sum_{i=1}^{L} \alpha_{n-d+j}[i]}\right),\ j = 0, 1, \ldots, d-1$.
7: $D \leftarrow \min_{j \in \{0,1,\ldots,d-1\}} \left| \frac{1}{d} \log P\left[x_{n-d+1+j:n+j} | x_{1:n-d+j}; \Lambda\right] - \bar{L} \right|$.
8: **if** $D > \delta$ **then**
9: $\hat{n} \leftarrow n$
10: Estimated interval is: $[\hat{n}-d+1, \hat{n}+d-1]$.
11: $z_d \leftarrow 1$
12: **end if**
13: $n \leftarrow n+1$
14: **while** $((n < N)$ AND $(z_d == 0))$

Comments:

1. The algorithm starts by considering the candidate change point $n = d$, with associated observation window $[1, \ldots, 2d-1]$. The first interval evaluated for this window is $[1, \ldots, d]$. For this interval, there are no past observations. Thus, the "conditional" likelihood is actually the unconditioned likelihood $P[x_{1:d}; \Lambda]$. The algorithm computes this likelihood by using $\sum_{i=1}^{L} \alpha_0[i]$ in the denominator term in (4), which in fact equals 1.
2. For each candidate change point, just one new forward recursion step is needed. Thus, the detection algorithm complexity scales with the complexity of the Forward algorithm which, for a sequence of length M, is $O\left(L^2 M\right)$.
3. The reference normalized log-likelihood $\bar{L}$ can be computed in different ways, customized to the application domain of interest: i) We can simply set $\bar{L} = 0$ if we expect the conditional likelihood of the sliding interval to significantly change once the change point is involved in the interval, ii) We can compute an ensemble average $\bar{L}$ from multiple "normal" time series that are known to contain no change point. Moreover, since this reference average log-likelihood may be a function of both n and d, we can compute an ensemble average value for each (n, d) pair; iii) $\bar{L}$ can be time series realization dependent and can again be a function of n and d (for example, it can be evaluated by locally, temporally averaging the log-likelihoods for each of the d time intervals of length d in the window $[n-d+1, n+d-1]$), for all values n that are "early enough" such that it is known that $[1, \ldots, n]$ does not contain a change point; iv) It can again be time series dependent and a function of d but not a function of n. In this case, we assume that $\bar{L}$ does not change much, as the past context (as n) increases; v) For each time series and at any given time n, we can compute $\bar{L}$ as the time-average of the conditional likelihoods of all intervals of length d up to (but not including) time n (i.e., including all times up until the present that have been rejected as change points). In the sequel we will exposit how we chose $\bar{L}$ in our experiments.
4. We use the absolute deviation from a null reference log-likelihood because for some applications anomalies will be associated with lower likelihoods than typical under the null whereas, for others, anomalies will actually yield higher likelihoods than are null-typical. This will be seen in our fatigue failure experiments.
5. The detection threshold δ can be set in one of several ways, e.g. to control the FPR for a given d.

3. Experimental results

In this section we validate the proposed HMM-based detection method using experiments from two different domains: instabilities in combustion systems and fatigue failure in a polycrystalline alloy material. In both cases we have time series generated from experiments with failures (instabilities) and with no failures (no instabilities). For simplicity, we assume there are two hidden (null) HMM states and we use Gaussian mixtures for the state-conditional density functions. The number of mixture components for each state is selected using the Bayesian Information Criterion (BIC) [40]. We also apply a CUSUM algorithm and the symbolic time series analysis technique (STSA) [41] with K-means and maximum entropy partitioning (MEP) [42] used to perform the symbolization (quantization) required by STSA. Finally, we compare with standard use of HMMs for detection, evaluating the joint likelihood of observations in the window [38,22], as described in 2.1, rather than the conditional likelihood. We will compare the results for these methods with our proposed method.

In this paper, we evaluate an STSA change point algorithm that mimics the structure of Algorithm 1. Specifically, a null time series (known to contain exclusively "normal" observations) is quantized into a discrete symbol sequence (e.g., using the K-means algorithm), which is then used to estimate a null PFSA. Then, operationally, for any new (test) window of observations, a new PFSA is estimated using the same quantizer partition as was used to construct the null PFSA. Therefore, we have a null PFSA (and its steady state probability vector), which was estimated based on an observation window for which it is known no anomaly is present, and a new PFSA, which is estimated based on a test observation window. The states' stationary probability vector at a given time n is obtained from the new PFSA at time n, and a surrogate for the log-likelihood deviation defined in Algorithm 1 is obtained by computing the Kullback–Leibler (KL) divergence between the new probability vector at time n and the probability vector obtained from the null PFSA. Then, mirroring the HMM-based algorithm, we find the minimum KL value over all d windows of length d that contain the point n and compare this to a detection threshold. Hence, for STSA, we apply the proposed Algorithm 1 structure (unlike previous change point detection algorithms based on STSA), but based on the KL measure rather than the absolute log-likelihood deviation. The KL measure is expected to increase when an anomaly occurs.

3.1. Detection of thermoacoustic instability onset in combustion systems

Thermoacoustic instabilities (TAI) in combustion systems are usually caused by spontaneous excitation of one or more natural modes of acoustic waves [43]. TAI are typically manifested by large-amplitude self-sustained chaotic pressure oscillations in the combustion chamber [44], which may lead to damage in mechanical structures if the pressure oscillations match one of the natural frequencies of the system. Traditional techniques of pressure-oscillation measurement for TAI detection, reported in the literature [45,46], attempt to extract the growth-rate information using the entire envelope of pressure oscillations. These techniques thus observe the entire acoustic time series and may not be suitable for online estimation of transient growth of oscillations and early detection of instability. Moreover, the time scales of TAI are on the order of milliseconds, which, given practical sampling rates, mandates an algorithm that can accurately detect an onset of TAI based on short-length sensor data.

The detection approach given by Algorithm 1 is our proposed candidate for such detection. The performance of this method is compared here with CUSUM and STSA techniques based on K-means and MEP [42] partitions. We also evaluate the standard HMM method, in which the conditional likelihood $P\left[x_{n-d+1+j:n+j}|x_{1:n-d+j};\Lambda\right]$ used in Algorithm 1 is replaced by the unconditioned likelihood $\widehat{P}\left[x_{n-d+1+j:n+j}|\Lambda\right]$. This approach is consistent with the HMM detection approaches in [38,22]. In the sequel, HMM$^{(1)}$ stands for the conventional HMM-based method, and HMM$^{(2)}$ stands for our conditional-likelihood HMM-based method.

For each experiment, the null HMM Λ is estimated by the Baum-Welch algorithm [36] using the first 1/10-th of the observations generated in the stable phase. Since the observations' variance increases as the system becomes unstable, the log conditional likelihood is expected to significantly decrease. Therefore, we set $\bar{L}=0$. The threshold δ is computed as

$$\delta = \max_{n\in\{1,\ldots,d\}} D(n) + \epsilon \tag{5}$$

where $D(n)$ is the anomaly measure, at time n, used in Algorithm 1, and ϵ is a hyperparameter chosen by splitting the experiments into two equal-sized sets; training and test set experiments. We evaluate a grid of candidate ϵ values and pick the one which achieves the best TDIR performance over the training experiments. Then, the performance of the proposed algorithm is evaluated (for this ϵ choice) using only the time series in the test set. For fair comparison, the same method is used for computing the threshold for the STSA techniques. Moreover, for each specimen, the same observations used to train the null HMM are also used to train the STSA null model.

3.1.1. Description of the experimental apparatus

Experiments in this subsection are obtained by using an electrically heated Rijke tube apparatus shown in Fig. 2 (see [34] for more details), where the process starts with a stable combustion that gradually becomes unstable. Experiments have been conducted by varying the air flow rate (Q) and the power input to the heater (E_{in}). A time series of pressure oscillations is collected over 30 s for each experiment, sampled at 8192 Hz and high-pass-filtered to attenuate the effects of low-

frequency environmental acoustics. For each experiment, the setup is heated to steady state with a power input of 200 W to the heater. Then the power input is abruptly raised to a high value, which eventually showed a limit cycle behavior in each experiment. A sample pressure signal is shown in Fig. 3, where instability is indicated by a large amplitude limit cycle. As shown in the figure, the pressure signal starts with a relatively small amplitude, and once the process starts becoming unstable the amplitude starts growing due to the onset of a Hopf bifurcation in the system [47,45]. This is a TCD problem (as indicated from the figure), with a short transient stage between the stable and unstable stages. Thermoacoustic instability is triggered by changing Q (ranging between 130 LPM and 250 LPM at increments of 20 LPM) and E_{in} (ranging between 800 W and 2000 W at increments of 200 W). For some values of Q and E_{in}, instability does not set in, which is depicted in the stability map of the system, as described in the work by Mondal et al. [34].

The onset time of Hopf bifurcation (m) changes from one experiment to another. To visually identify a unique change time as ground truth, against which we compare our method's estimate, we remove a small block of the time series that includes the point x_m. By doing so, we remove visual uncertainty, creating certitude that all the points in the time series up to the beginning of the removed block are from the stable phase, with points after the removed block from the unstable phase. However, the block removed is on the order of 100 points, while the total transient block is on the order of $10,000$ points; thus the removed block does not significantly change the gradual transition from stable to unstable combustion. Therefore we consider the time point immediately after the removed block as the true change time. Since each original pressure signal has millions of sample points, we consider the last $1/10$-th of the signal before the transition from stable to unstable combustion, the whole transient zone, and the first $1/10$-th of the signal after the end of the transient zone. The resulting signal is further downsampled by 100 so that the signal length is reduced while maintaining the main shape and features of the original signal. This is for signals that do contain a transition from stable to unstable combustion. For signals with purely stable combustion, we consider a $1/10$-th segment of the signal and downsample by 100. We use the same process for the ultrasonic signals considered in the next subsection. Fig. 4 shows samples of the resulting pressure signals after downsampling. Fig. 5 shows a sample pressure signal after downsampling, and a magnification of the transient part, which shows that the transition from stable to unstable combustion is gradual, not abrupt.

3.1.2. Experimental validation

We conducted 145 experiments. In each, there is a single change from stable combustion to the unstable phase, with the change time not fixed across the experiments. Then for each experiment we extract two pressure time series. One is safely confined to cover a sub-interval of the stable phase, with the second one consisting of the full time series. Therefore, we have 290 time series: 145 with transitions from stable to unstable combustion (72 used in the training set for setting ϵ and 73 in the test set), and 145 with purely stable combustion (all in the test set). For each time series, we attempt to make a single change point detection. By applying a detection algorithm to each of the time series, we can measure the TDIR, FPR, and bias and variance of the estimator, for increasing values of the sliding window length d.

We start with the CUSUM method. One standardized version of the CUSUM algorithm returns the first index of the time series $x_{1:N}$ that has drifted k standard deviations outside the nominal mean. This detection approach is suitable for a deviation in mean. However, as shown in Fig. 3, our data shows a change in variance. Therefore, we use a CUSUM variant suitable for change due to variance deviation. Inspired by [48] we introduce the modified CUSUM algorithm described by the following stopping rule:

$$T_s = \inf\{n : |\rho(n)| \geqslant \delta\} \tag{6}$$

$$\rho(n) = \sum_{k=1}^{n} x_k^2 - n\sigma_0^2 \tag{7}$$

$$\sigma_0^2 = \frac{1}{N_0}\sum_{k=1}^{N_0} x_k^2 \tag{8}$$

where T_s is the stopping time at which change detection is declared, N_0 is the number of samples taken from the nominal state to compute σ_0^2, and with the data shifted to have zero mean (so that σ_0^2 is a variance estimate). We tried different values of the threshold δ and picked the one with best TDIR performance. The results are shown in Fig. 6. At each value of the window size, d, the modified CUSUM is applied to the window $x_{n:n+d-1}$ to make a detection, where the window $x_{n:n+d-1}$ is slid from the beginning until the end of the time series $x_{1:N}$, until a detection is made. As shown in Fig. 6, the detection performance generally improves with window size. Furthermore, TDIR achieves its maximum attainable value, 1.0, at $d \sim 425$, and FPR achieves zero for the same value of d. These delay values are much larger than for the comparison methods, as will be seen shortly.

Let us now apply STSA techniques. We consider two STSA methods; the first uses K-means clustering for partitioning and the second uses the maximum entropy partition (MEP) [40,42]. The results for K-means are given in Fig. 7, and for MEP in Fig. 8, both evaluated for different values of the symbol alphabet size (number of clusters, K). As shown in these two figures, STSA performance is much better than modified CUSUM. We notice for both STSA methods how the TDIR converges close to one with window size ~ 120, much less than ~ 425 required for the modified CUSUM. A comparison between the two figures for K-means and MEP shows a little improvement for MEP in TDIR performance. However, the opposite is true for FPR, where

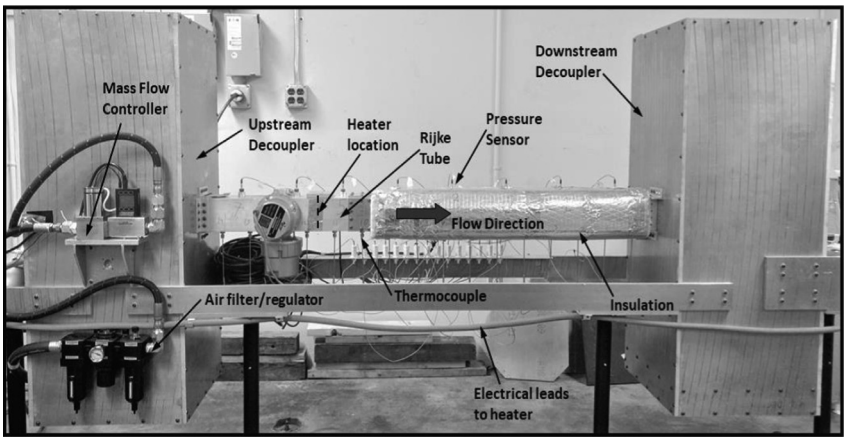

Fig. 2. Rijke Tube combustion apparatus.

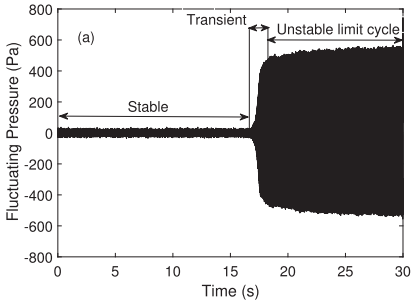

Fig. 3. A pressure signal showing the transition from a stable combustion to unstable limit cycle.

MEP converges to zero at window size $\geqslant 80$ with alphabet sizes 3 and 4, while K-means achieves zero FPR at window size $\geqslant 60$ (with alphabet sizes 3, 4, and 5), both having much better FPR performance compared to the modified CUSUM.

Next, we pick the K-means and MEP solutions with best TDIR performance (i.e., alphabet size = 5 for K-means and 3 for MEP), and compare them with the results obtained for the proposed HMM-based detection method (Algorithm 1), denoted by $\text{HMM}^{(2)}$. The results are given in Fig. 9, which shows excellent results of the proposed HMM scheme for all performance measures: TDIR, FPR, bias and variance of the estimated change time, with consistent improvement achieved over the STSA techniques. Note also that the STSA results in Figs. 7 and 8 are much better than CUSUM in Fig. 6. Thus, the HMM method, which dominates the STSA methods, is also substantially better than CUSUM, with respect to all four performance measures

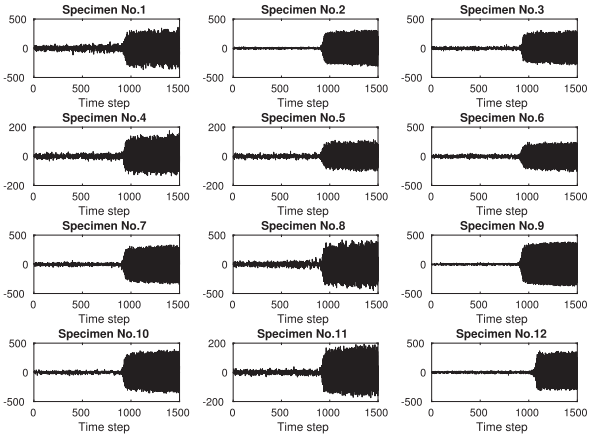

Fig. 4. Samples of pressure signals after downsampling.

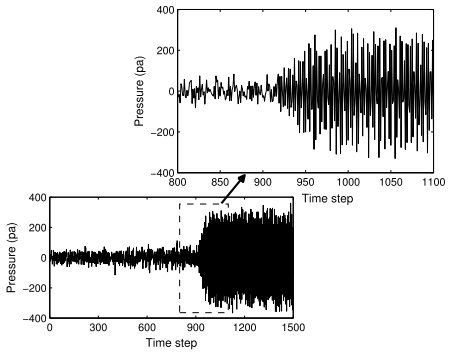

Fig. 5. A sample pressure signal after downsampling, showing the gradual transition from a stable combustion to unstable limit cycle.

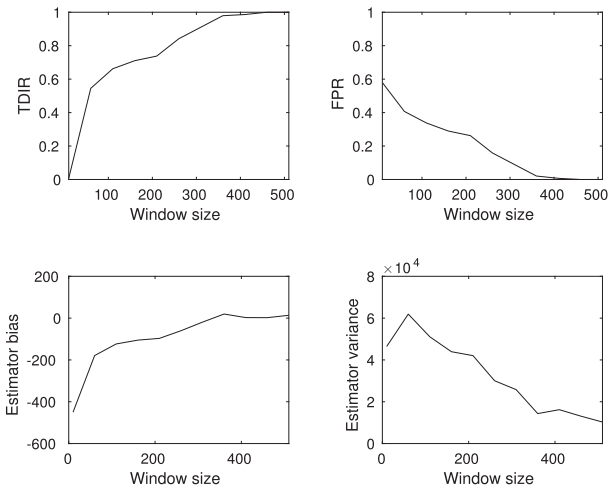

Fig. 6. Detection of thermoacoustic instabilities using modified CUSUM.

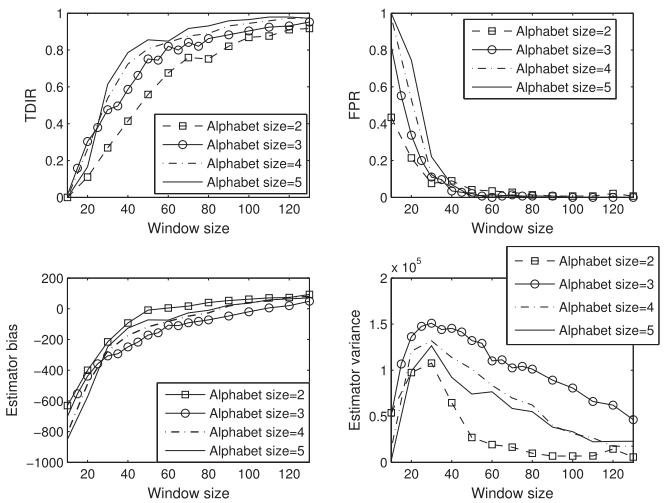

Fig. 7. Detection of thermoacoustic instabilities using STSA with K-means.

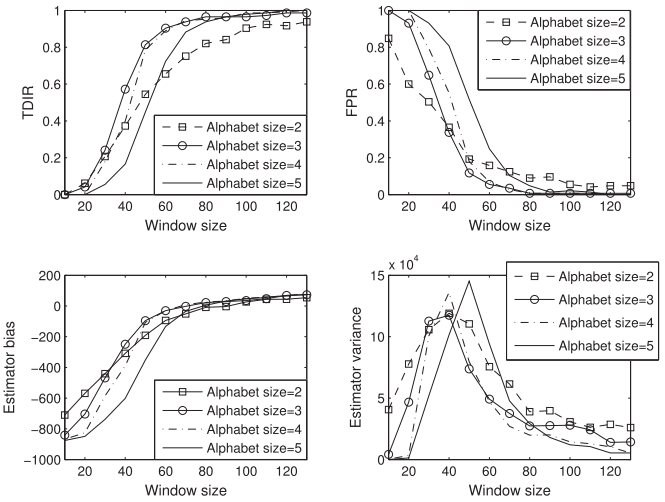

Fig. 8. Detection of thermoacoustic instabilities using STSA with MEP.

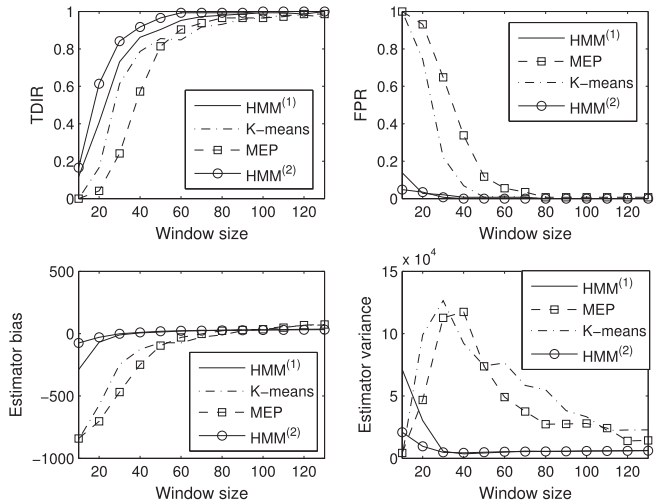

Fig. 9. Detection of thermoacoustic instabilities using K-means (alphabet size = 5), MEP (alphabet size = 3), $HMM^{(1)}$ (number of states = 2), and $HMM^{(2)}$ (number of states = 2).

evaluated. The HMM method's FPR goes to zero much faster than the STSA methods; the HMM method dominates at all window sizes. But most notable is the much better variance characteristic of the HMM method. Also, the figure shows the performance of the standard HMM-based method, $HMM^{(1)}$, which uses the approximate joint subsequence likelihood, i.e. which ignores the past when evaluating the likelihood of a subsequence. Note that the conditional likelihood based approach outperforms the joint likelihood based variant for all window sizes, d, with respect to all four performance measures, but especially with respect to TDIR.

Fig. 10 shows the performance of the proposed HMM-based method with the anomaly measure obtained by using the min (denoted as $HMM^{(2)}$-min) and max (denoted as $HMM^{(2)}$-max) operators (over all subsequences of length d for the current candidate change point). The figure shows generally comparable results, with $HMM^{(2)}$-max giving better performance for TDIR and $HMM^{(2)}$-min giving better performance for FPR, the bias, and variance of the estimated change point. These results are not surprising since 'max' is expected to be a more sensitive (but less specific) statistic than 'min'. Note also in Figs. 6–10 that, for all the methods, TDIR increases, and FPR decreases, with increasing window size, d. This experimental observation is theoretically supported in Appendix A. This suggests one might want to choose d quite large. However, for a given application, d must be restricted in order to achieve low-latency detection and the potential for damage mitigation/response.

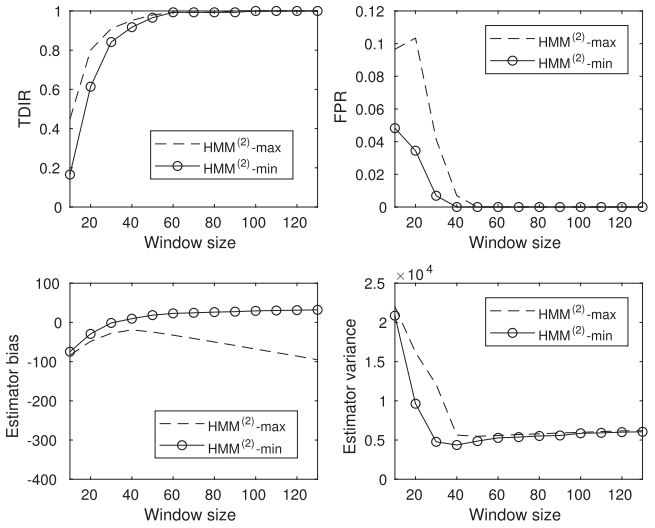

Fig. 10. HMM-based method performance with min and max operators used to compute the anomaly measure.

3.2. Detection of fatigue crack initiation in a polycrystalline alloy material

Modeling fatigue failure in mechanical structures has received great attention by many researchers, e.g. [49,50]. In any mechanical structure, millions of initial materials' defects (such as dislocations, voids, inclusions and slip bands) exist inside the microstructure even before the structure is used. In general, fatigue damage is critically dependent on these initial defects, from which cracks start to nucleate and merge together, generating bigger cracks, leading to the failure of the structure [51]. These initial defects are usually distributed in a highly random fashion, producing large uncertainties in the crack initiation and propagation process even under identical loading. Therefore, fatigue failure is considered an unpredictable and highly stochastic process. In the following, we give a brief description of the apparatus we use to get ultrasonic signals correlated with fatigue damage.

3.2.1. Description of the experimental apparatus

Although structural fatigue damage is not easily measured directly, damage may be correlated with signals that can be measured and used for fatigue damage detection. In this work we use ultrasonic signals, which have been commonly used for real time damage sensing in the aerospace and nuclear power industries to detect flaws in structures [52]. Fig. 11 shows the experimental apparatus, built upon an MTS 831.1 fatigue testing machine, which can be used to apply external load to test specimens with the desired cyclic properties: amplitude, frequency, and the shape of the force function; it is also capable of applying random loading. The other component is the ultrasonic part, which functions by emitting high frequency ultrasonic pulses by using a piezoelectric transducer. A Matec TB1000 Gated Amplifier PC add-in card drives the piezoelectric transducer with a gated sine wave with amplitude of 300 V. The generated signal consists of short bursts of a sine wave of constant amplitude interrupted by relatively long periods of 0 V. This signal propagates through the specimen and is captured by transducers arrayed on the other side of the notch. The received signal is routed through a high frequency selector switch to a National Instruments NI5911 Oscilloscope card in the PC. The acquisition of the ultrasonic signal is synchronized with the load applied to the specimen so that data is acquired at the low load and at the high load (see [53] for more details). When cracks occur, part of the ultrasonic signal will be reflected, and hence the received signal is attenuated. This attenuation increases as more cracks initiate and propagate, until the specimen breaks and only noise signal is left. In this paper, data are collected at a sampling rate $\sim 21,800$ Hz.

As we did for the combustion signals, we remove a small block from the signal at the visually assessed change point. We then consider the last 1/10-th of the signal just before the transition from healthy to damaged state, the whole transient zone (except for the small removed block), and the first 1/10-th segment of the signal just after the end of the transient zone. Then the resulting signal is further downsampled by 100 so that the signal length is reduced while maintaining its main shape and features. This is for signals that contain a transition from a healthy to a damaged state. For signals with purely healthy state, we only consider a 1/10-th segment of the signal, and then downsample by 100. Thus the data are collected at an effective sampling rate ~ 218 Hz.

Fig. 12 shows such ultrasonic signals, after this segmentation and downsampling, for 12 sample specimens made of steel-aluminum alloy. As shown in the figure, the signal begins to significantly attenuate at a certain time instant. Such time instants roughly estimate phase transitions in fatigue damage, where cracks reach critical lengths and the damage process starts growing aggressively (until the specimen breaks and only a noise signal remains). As the figure indicates, this is a QCD

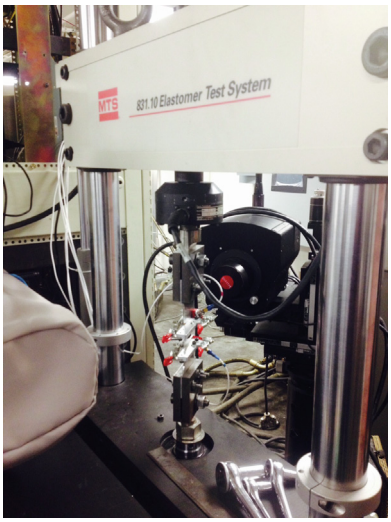

Fig. 11. Apparatus for fatigue failure experiments.

problem, with the system remaining in the transient (increasing damage, indicated by increasing signal attenuation) phase until the specimen breaks (with the signal then flat). Although all the specimens used in this work have the same dimensions and are made from the same material, the plots in Fig. 12 show that the change time is different from one specimen to another. This difference is due to initial microstructural defects which are specimen-dependent.

3.2.2. Experimental validation

We conducted 17 experiments and proceeded in a similar way as for the combustion experiments to generate 34 ultrasonic time series – 17 with transitions from healthy to a gradually damaged state until the specimen breaks (8 used in the training set for setting ϵ and 9 in the test set); another 17 with healthy condition, for which there is no detectable fatigue failure. As we mentioned, the ultrasonic time series obtained here are highly stochastic and highly noisy. Moreover, unlike the pressure signals in the combustion experiments, the ultrasonic signals at the receiver tend to decrease in variance (approaching the null density mean value – zero) as damage accumulates in the specimens. In this situation, unlike for the combustion experiment, the likelihood under the null is expected to (atypically) increase as the data variance decreases, because after the change point the observations tend to hover close to the null mean – this is why we use the absolute deviation in Algorithm 1. It covers both atypically small and atypically large log-likelihoods compared to the null (reference) log-likelihood.

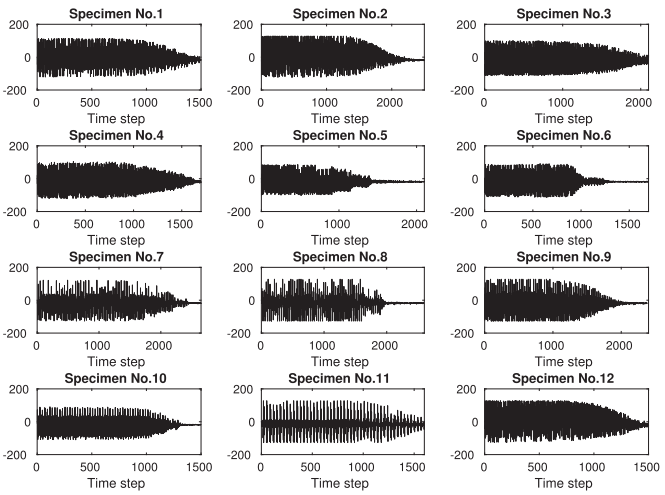

Fig. 12. Ultrasonic signals for sample specimens, with signal strength value on the y-axis and time on the x-axis.

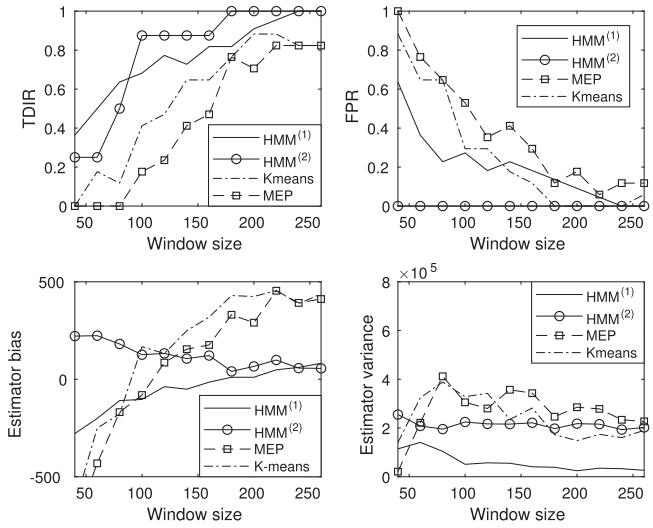

Fig. 13. Detection of fatigue failure onset using K-means (alphabet size = 3), MEP (alphabet size = 4), HMM$^{(1)}$ (number of states = 2), and HMM$^{(2)}$ (number of states = 2).

We applied STSA methods using K-means and MEP for partitioning, and HMM$^{(2)}$ inference with the proposed forward recursion, both combined with the proposed detection approach in Algorithm 1. We also applied the standard HMM method, HMM$^{(1)}$, which uses the approximate joint likelihood instead of the conditional likelihood. Again, for STSA, the surrogate quantity for the deviation in log-likelihood in Algorithm 1 is the Kullback divergence between the current PFSA states' stationary probability vector and the null PFSA states' stationary probability vector [41,42].

We follow the same methods we used in the combustion experiments to compute the null models and the thresholds for both HMM and STSA techniques. However, since the likelihood under the null is expected to increase as the data variance decreases, it is not reasonable to set $\overline{L} = 0$, as we did in the combustion experiments. Alternatively, we compute $\overline{L}$ as follows. For HMM$^{(1)}$, $\overline{L}$ is the mean of the log-likelihoods of the first d intervals of length d, i.e.,

$$\overline{L} = \frac{1}{d}\sum_{n=1}^{d} L(n) \tag{9}$$

However, this choice of $\overline{L}$ is not plausible for HMM$^{(2)}$. In this case, $\overline{L}$ should be time-varying, consistent with the conditional log likelihood depending on the growing past. Therefore, we compute a time-varying $\overline{L}$ as

$$\overline{L}(n) = \begin{cases} \frac{1}{n}\sum_{m=1}^{n} L(m), & \text{if } n \leqslant T_0 \\ \frac{1}{T_0}\sum_{m=1}^{T_0} L(m), & \textit{otherwise} \end{cases} \tag{10}$$

for all $n > 0$, where T_0 is chosen to be well within the normal phase (much less than the true change instant). We used $T_0 = \frac{\tau}{10}$, where τ is the true change instant.

Fig. 13 shows the results for STSA and HMM methods. For K-means and MEP, we used different values of alphabet size, ranging from 2 to 5, and picked the one with the best TDIR performance. Specifically, we used alphabet size = 4 for MEP and 3 for K-means. The results show excellent performance of the HMM algorithms (HMM$^{(1)}$ and HMM$^{(2)}$) in terms of TDIR, FPR, and estimator bias and variance, with consistent improvement achieved over the STSA techniques. Again, it is seen that the conditional likelihood based (proposed) method (HMM$^{(2)}$) overall outperforms the joint likelihood based variant of our approach (HMM$^{(1)}$) with respect to TDIR and FPR. The bias of the two methods is comparable and the variance of HMM$^{(1)}$ is smaller.

4. Conclusion

The proposed strictly low-delay, localizing, HMM conditional-likelihood based transient change detection algorithm has been shown to achieve substantially improved performance over comparison methods with respect to appropriate performance criteria proposed here: a novel TDIR, FPR, the bias in estimating the change point, and the variance of the change point

estimate, all as a function of the delay tolerance parameter d. We also proved, within a supervised setting, that conditioning (as we do) is required for optimal detection and that detection performance is monotonically non-decreasing with increasing window size, d, consistent with our experimental results. Since many systems that involve change point detection involve control inputs, in our future work we may consider the framework of input–output hidden Markov models (IOHMMs) [54] in devising a scheme that chooses control inputs to e.g. maximize TDIR under strict detection delay given a fixed FPR. Such control inputs could also be used to induce desired change points (associated with improved performance, rather than with system failure).

CRediT authorship contribution statement

David J. Miller: Conceptualization, Methodology, Writing - original draft, Supervision. **Najah F. Ghalyan:** Methodology, Software, Validation, Writing - review & editing. **Sudeepta Mondal:** Validation, Writing - review & editing. **Asok Ray:** Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The work reported in this paper has been supported in part by U.S. Air Force Office of Scientific Research (AFOSR) under Grant Nos. FA9550-17–1-0070, FA9550-15–1-0400 and FA9550-18–1-0135 in the area of dynamic data-driven application systems (DDDAS). Any opinions, findings, and conclusions in this paper are those of the authors and do not necessarily reflect the views of the sponsoring agencies. The second author would like to thank the Higher Committee for Education Development (HCED) in Iraq for their financial support.

Appendix A. Proof that i) conditioning on past observations is required for optimality in a (supervised) detection setting and ii) detection performance is monotonically non-decreasing with increasing delay tolerance *d*

In this appendix, to simplify theoretical analysis, we consider detection based on a window of observations $\underline{x}_n = \{x_n, \ldots, x_{n+d-1}\}$. Further, solely for clarity's sake, suppose that, if there is a change point in the time series, it is at x_n, i.e., in this case all the observations in $\{x_n, \ldots, x_{n+d-1}\}$ were generated according to the new (change state) random process. Also define the growing window vector $\underline{\underline{x}}_n = \{x_0, \ldots, x_{n+d-1}\}$. Denote the joint density of $\underline{\underline{X}}_n$ under the null hypothesis by $f_{\underline{\underline{X}}_{n/0}}(.;\Lambda_0)$. The alternative hypothesis is that $\{x_0, \ldots, x_{n-1}\}$ was generated according to the nominal random process and $\{x_n, \ldots, x_{n+d-1}\}$ was generated according to the new process (starting with change point x_n), $f_{\underline{X}_{n/1}}(.;\Lambda_1)$, independent of past observations. Suppose we wish to make optimal (minimum error rate) detection, given knowledge of $f_{\underline{\underline{X}}_{n/0}}(.)$ and $f_{\underline{X}_{n/1}}(.)$. The error rate can be expressed as:

$$P_e[n] = P_1 \int_{R_0} f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{z}}_{n-d};\Lambda_0\right) f_{\underline{X}_{n/1}}(\underline{z}_n;\Lambda_1) d\underline{\underline{z}}_n + P_0 \int_{R_1} f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{z}}_n;\Lambda_0\right) d\underline{\underline{z}}_n, \tag{11}$$

where $\underline{\underline{z}}_n$ is a vector of dummy variables of integration and where R_0 is the decision region assigned to the no-change state and R_1 is the decision region assigned to the change state. If the normal and changed state class priors P_1 and P_0 are unknown, they can be set to $[1/2,\ 1/2]$. Using a standard simplification, this is re-expressed as:

$$P_e[n] = P_0 + \int_{R_0} \left(P_1 f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{z}}_{n-d};\Lambda_0\right) f_{\underline{X}_{n/1}}(\underline{z}_n;\Lambda_1) - P_0 f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{z}}_n;\Lambda_0\right) \right) d\underline{\underline{z}}_n,$$

All the values for which the integrand is negative should be assigned to R_0, to minimize $P_e[n]$. All the values for which the integrand is positive should be assigned to R_1 (so that they do not make positive contribution to the integral). Thus, we have the Bayes-optimal rule:

$$R_0 = \left[\underline{\underline{x}} : P_0 f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{x}};\Lambda_0\right) \geqslant P_1 f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{x}}_{n-d}\right) f_{\underline{X}_{n/1}}(\underline{x}_n;\Lambda_1) \right],$$

with R_1 the complement of this region in $\mathbb{R}^d$. This can be rewritten, cancelling the common term $f_{\underline{\underline{X}}_{n-d/0}}(.;\Lambda_0)$, as

$$R_0 = \left[\underline{\underline{x}} : P_0 f_{\underline{X}_n/\underline{\underline{x}}_{n-d},0}\left(\underline{x}_n/\underline{\underline{x}}_{n-d};\Lambda_0\right) \geqslant P_1 f_{\underline{X}_{n/1}}(\underline{x}_n;\Lambda_1) \right].$$

Note that the minimum error rate rule is based on the (conditional) density, considering observations (from 0 to $n-d$) outside the current window.

Now, consider the more relevant (detection) problem, with the goal to minimize the missed detection rate given a fixed false positive rate. That is, the problem:

$$\min_{R_0} P_1 \int_{R_0} f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{z}}_{n-d};\Lambda_0\right) f_{\underline{X}_{n/1}}(\underline{z}_n;\Lambda_1) d\underline{\underline{z}}_n$$

subject to

$$P_0 \int_{\overline{R}_0} f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{z}}_n;\Lambda_0\right) d\underline{\underline{z}}_n = \gamma$$

The associated Lagrangian objective is:

$$L = P_1 \int_{R_0} f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{z}}_{n-d};\Lambda\right) f_{\underline{X}_{n/1}}(\underline{z}_n;\Lambda_1) d\underline{\underline{z}}_n + \lambda\left(P_0 \int_{\overline{R}_0} f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{z}}_n;\Lambda_0\right) d\underline{\underline{z}}_n - \gamma\right)$$

Now, dividing by $P_1 + \lambda P_0$, note that minimizing L with respect to R_0 is equivalent to minimizing

$$\frac{P_1}{P_1+\lambda P_0} \int_{R_0} f_{\underline{\underline{X}}_{n-d/0}}\left(\underline{\underline{z}}_{n-d};\Lambda\right) f_{\underline{X}_{n/1}}(\underline{z}_n;\Lambda_1) d\underline{\underline{z}}_n + \frac{\lambda P_0}{P_1+\lambda P_0}\left(\int_{\overline{R}_0} f_{\underline{\underline{X}}_{n/0}}\left(\underline{\underline{z}}_n;\Lambda_0\right) d\underline{\underline{z}}_n - \gamma'\right) \tag{12}$$

where $\gamma' = \gamma/P_0$. Now simply recognize, ignoring the γ' term which does not affect the solution, Eq. (12) can be reinterpreted as a probability of error expansion, just as Eq. (11), but with class priors $\left[\frac{P_1}{P_1+\lambda P_0} \;\; \frac{\lambda P_0}{P_1+\lambda P_0}\right]$.

Thus, our previous probability of error analysis also holds for Eq. (12), i.e. the optimal decision rule, minimizing Eq. (12) and thus achieving least missed detection rate given fixed false positive rate, must use the (conditional) density, conditioning on $\underline{\underline{X}}_{n-d} = \underline{x}_{n-d} = \{x_0, \ldots, x_{n-1}\}$, just as one must do to achieve the Bayes-optimal minimum error rate classifier.

Moreover, consider two window sizes d and $d', d' < d$. Note that making decisions based on the window of observations $\{x_n, \ldots, x_{n+d'-1}\}$ is equivalent to observing $\{x_n, \ldots, x_{n+d-1}\}$ but only using the observations up to time $n + d' - 1$ to make decisions. But the above formulation shows that the optimal detection rule, determining $\overline{R}_0$, makes use of all the observations in the window $\{x_n, \ldots, x_{n+d-1}\}$. Thus, given a fixed false positive rate γ, the true detection rate of any detector using just the window $\{x_n, \ldots, x_{n+d'-1}\}$ cannot be larger than the true detection rate of the optimal detector making use of $\{x_n, \ldots, x_{n+d-1}\}$. This proves monotonically non-decreasing detection performance, with increasing window size, d.

References

[1] F. Chamroukhi, A. Same, G. Govaert, P. Aknin, Time series modeling by a regression approach based on a latent process, Neural Networks 22 (2009) 593–602.
[2] K. Frick, A. Munk, H. Sieling, Multiscale change point inference, Journal of the Royal Statistical Society. Series B: Statistical Methology 76 (3) (2014) 495–580.
[3] J. Chen, A. Gupta, Parametric Statistical Change Point Analysis: with Applications to Genetics, Medicine, and Finance, second ed., Springer, 2012.
[4] E.S. Page, Continuous inspection schemes, Biometrika 41 (1/2) (1954) 100–115.
[5] C.-M. Kuan, K. Hornik, The generalized fluctuation test: A unifying view, Econometric Reviews 14 (2) (1995) 135–161.
[6] M. Basseville, I.V. Nikiforov, Detection of Abrupt Changes - Theory and Application, Prentice Hall Inc, 1993.
[7] S. Mishra, O.A. Vanli, C. Park, A multivariate cumulative sum method for continuous damage monitoring with lamb-wave sensors, International Journal of Prognostics and Health Management 6 (2015) 1–11.
[8] J. Bai, P. Perron, Estimating and testing linear models with multiple structural changes, Econometrica 66 (1) (1998) 47–78.
[9] B. Jushan, P. Pierre, Computation and analysis of multiple structural change models, Journal of Applied Econometrics, 18 (1), 1–22.
[10] J. Bai, Likelihood ratio tests for multiple structural changes, Journal of Econometrics 91 (2) (1999) 299–323.
[11] D. Egea-Roca, J.A. Lopez-Salcedo, G. Seco-Granados, H. Vincent Poor, Performance bounds for finite moving average tests in transient change detection, IEEE Transactions on Signal Processing 66 (2018).
[12] G.V. Moustakides, Multiple optimality properties of the shewhart test, Sequential Analysis 33 (3) (2014) 318–344.
[13] A. Shiryaev, On optimum methods in quickest detection problems, heory of Probability & Its Applications 8 (1) (1963) 22–46.
[14] A. Tartakovsky, V. Veeravalli, General asymptotic Bayesian theory of quickest change detection, Theory of Probability & Its Applications 49 (3) (2005) 458–497.
[15] S.C. Chin, A. Ray, V. Rajagopalan, Symbolic time series analysis for anomaly detection: A comparative evaluation, Siginal Processing 85 (9) (2005) 1859–1868.
[16] X. Jin, Y. Guo, S. Sarkar, A. Ray, R.M. Edwards, Anomaly detection in nuclear power plants via symbolic dynamic filtering, IEEE Transactions on Nuclear Science 58 (1) (Feb 2011) 277–288.
[17] Y. Li, D.K. Jha, A. Ray, T.A. Wettergren, Information fusion of passive sensors for detection of moving targets in dynamic environments, IEEE Transactions on Cybernetics 47 (1) (Jan 2017) 93–104.
[18] Y. Li, D.K. Jha, A. Ray, T.A. Wettergren, Information-theoretic performance analysis of sensor networks via markov modeling of time series data, IEEE Transactions on Cybernetics 48 (6) (June 2018) 1898–1909.
[19] D. Jha, N. Virani, J. Reimann, A. Srivastav, A. Ray, Symbolic analysis-based reduced order Markov modeling of time series data, Signal Processing 149 (2018) 68–81.
[20] J. Hopcroft, R. Motwani, J. Ullman, Introduction to Automata Theory, Languages, and Computation, 2nd ed., Addison-Wesley, 2001.
[21] B.P. Kitchens, Symbolic Dynamics: One-sided, Two-sided and Countable State Markov Shifts, Springer-Verlag, Berlin Heidelberg GmbH, 1998.
[22] T.M. Luong, V. Perduca, G. Nuel, Hidden Markov model applications in change-point analysis, arXiv preprint arXiv:1212.1778, 2012.
[23] C.-D. Fuh, Sprt and cusum in hidden markov models, The Annals of Statistics, 31 (3) (2003) 942–977. [Online]. Available: doi: 10.1214/aos/1056562468.
[24] A.G. Tartakovsky, On asymptotic optimality in sequential changepoint detection: Non-iid case, IEEE Transactions on Information Theory 63 (2017) 3433–3450.

[25] W. Khreich, E. Granger, R. Sabourin, A. Miri, Combining hidden markov models for improved anomaly detection, in: 2009 IEEE Int Conf on Communications, June 2009, pp. 1–6.
[26] Y. Yuan, Y. Meng, L. Lin, H. Sahli, Y. Anzhi, J. Chen, Z. Zhao, Y. Kong, D. He, Continuous change detection and classification using hidden markov model: A case study for monitoring urban encroachment onto farmland in beijing, Remote Sensing, 7 (11) (2015) 15-318–15-339.
[27] Z. Bouyahia, L. Benyoussef, S. Derrode, Unsupervised sar images change detection with hidden markov chains on a sliding window, Proceedings of SPIE – The International Society for Optical Engineering 6748 (2007) 10.
[28] S.D. Zied Bouyahia, Lamia Ben Youssef, Change detection in synthetic aperture radar images with a sliding hidden markov chain model, Journal of Applied Remote Sensing, 2 (1) (2008) 1–13 – 13 [Online]. Available: doi: 10.1117/1.2957968.
[29] E. Dorj, C. Chen, Anomaly detection approach using hidden Markov model, 2013 IEEE Aerospace Conference (2013) 1–10.
[30] W.R. Blanding, P.K. Willett, Y. Bar-Shalom, S. Coraluppi, Multisensor track management for targets with fluctuating snr, IEEE Transactions on Aerospace and Electronic Systems 45 (4) (2009) 1275–1292.
[31] T.M. Luong, Y. Rozenholc, G. Nuel, Fast estimation of posterior probabilities in change-point analysis through a constrained hidden Markov model, Computational Statistics & Data Analysis 68 (2013) 129–140.
[32] A. Sultana, A. Hamou-Lhadj, M. Couture, An improved hidden markov model for anomaly detection using frequent common patterns, in: 2012 IEEE International Conference on Communications, June 2012, pp. 1113–1117.
[33] H.O. Omoregbee, P.S. Heyns, Fault detection in roller bearing operating at low speed and varying loads using Bayesian robust new hidden Markov model, Journal of Mechanical Science and Technology, 32 (9) (2018) 4025–4036. [Online]. Available: doi: 10.1007/s12206-018-0802-8.
[34] S. Mondal, N.F. Ghalyan, A. Ray, A. Mukhopadhyay, Early detection of thermoacoustic instabilities using hidden markov models, Combustion Science and Technology 191 (8) (2019) 1309–1336.
[35] N.F. Ghalyan, S. Mondal, D.J. Miller, A. Ray, Hidden Markov modeling-based decision-making using short-length sensor time series, The ASME Journal of Dynamic Systems, Measurement, and Control, 141 (10) (2019) 104 502–104 502–6.
[36] L. Rabiner, B.-H. Juang, Fundamentals of Speech Recognition, 1st ed., Prentice Hall, 1993.
[37] T. Cover, J. Thomas, Elements of Information Theory, Wiley-Interscience Publication, 2006.
[38] H. Nguyen, G. McLachlan, P. Orban, P. Bellec, A. Janke, Maximum pseudolikelihood estimation for model-based clustering of time series data, Neural Computation 29 (2017) 990–1020.
[39] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proceedings on IEEE 77 (2) (1989) 257–286.
[40] K. Murphy, Machine Learning: A Probabilistic Perspective, first ed., The MIT Press, 2012.
[41] A. Ray, Symbolic dynamic analysis of complex systems for anomaly detection, Signal Processing 84 (7) (2004) 1115–1130.
[42] S. Bahrampour, A. Ray, S. Sarkar, T. Damarla, N. Nasrabadi, Performance comparison of feature extraction algorithms for target detection and classification, Pattern Recognition Letters 34 (2013) 2126–2134.
[43] T. Lieuwen, V. Yang, Combustion Instabilities in gas turbine engines: operational experience, fundamental mechanisms, and modeling, American Institute of Aeronautics and Astronautics (2005) 3–26, ch. 1.
[44] K. Matveev, Thermoacoustic instabilities in the Rijke tube: experiments and modeling, Ph.D. dissertation, California Institute of Technology, 2003.
[45] G. Rigas, N. Jamieson, L. Li, M. Juniper, Experimental sensitivity analysis and control of thermoacoustic systems, Journal of Fluid Mechanics 787 (2016).
[46] N. Jamieson, G. Rigas, M. Juniper, Experimental sensitivity analysis via a secondary heat source in an oscillating thermoacoustic system, International Journal of Spray and Combustion Dynamics 9 (4) (2017) 230–240.
[47] D. Zhao, Transient growth of flow disturbances in triggering a Rijke tube combustion instability, Combustion and Flame 159 (2012) 2126–2137.
[48] V. Kireichikov, V. Mangushev, I. Nikiforov, Investigation and application of cusum algorithms to monitoring of sensors, in: Statistical Problems of Control (in Russian), 1990, pp. 124–130.
[49] S. Kwofie, N. Rahbar, A fatigue driving stress approach to damage and life prediction under variable amplitude loading, International Journal of Damage Mechanics 22 (2012) 393–404.
[50] A. Aeran, S. Siriwardane, O. Mikkelsen, I. Langen, A new nonlinear fatigue damage model based only on s-n curve parameters, International Journal of Fatigue 103 (2017) 327–341.
[51] S. Suresh, Fatigue of Materials, second ed., Cambridge University Press, 2004.
[52] S. Gupta, A. Ray, Symbolic dynamics filtering for data-driven pattern recognition, Pattern Recognition: Theory and Applications, 2007, pp. 17–71.
[53] E. Keller, Real-time sensing of fatigue crack damage for information-based decision and control, Ph.D. dissertation, Pennsylvania State University, 2001. [Online]. Available: https://books.google.com/books?id=FTH6jwEACAAJ.
[54] Y. Bengio, P. Frasconi, Input-output HMMs for sequence processing, IEEE Transactions on Neural Networks 7 (1996) 1231–1249.

Data-Driven Detection and Classification of Regimes in Chaotic Systems Via Hidden Markov Modeling

Chandrachur Bhattacharya
Departments of Mechanical and Electrical Engineering,
Pennsylvania State University,
University Park, PA 16802
e-mail: chandrachur.bhattacharya@gmail.com

Asok Ray
Departments of Mechanical Engineering and Mathematics,
Pennsylvania State University,
University Park, PA 16802
e-mail: axr2@psu.edu

Chaotic dynamical systems are essentially nonlinear and are highly sensitive to variations in initial conditions and process parameters. Chaos may appear both in natural (e.g., heartbeat rhythms and weather fluctuations) and human-engineered (e.g., thermo-fluid, urban traffic, and stock market) systems. For prediction and control of such systems, it is often necessary to be able to distinguish between non-chaotic and chaotic behavior; several methods exist to detect the presence (or absence) of chaos, specially in noisy signals. A dynamical system may exhibit multiple chaotic regimes, and apparently, there exist no methods, reported in open literature, to classify these regimes individually. This paper demonstrates an application of standard hidden Markov modeling (HMM), which is a commonly used supervised method, as a technique to classify multiple regimes from a time series of dynamical systems, where classified regimes could be chaotic or non-chaotic. The proposed HMM-based method of regime classification has been tested using numerical data obtained from several well-known chaotic dynamical systems (e.g., Hénon, forced Duffing, Rössler, and Lorenz attractor). It is apparently well-suited to serve as a bench mark for the development of alternative data-driven methods to enhance the performance (e.g., accuracy and computational speed) of regime classification in chaotic dynamical systems. [DOI: 10.1115/1.4047817]

Keywords: chaos, classification, hidden Markov modeling, Hénon map, Duffing, Rössler systems, Lorenz attractor, chaotic system, machine learning, nonlinear systems

1 Introduction

Edward Lorenz introduced the concept of chaos in the form of the Butterfly Effect [1], where he suggested that the atmospheric phenomena are highly sensitive to exogenous perturbations. Chaotic dynamical systems are essentially nonlinear and are usually deterministic with an appearance of randomness, but having underlying patterns, feedback loops, and a structure of *repetitive order*, which are very sensitive to initial conditions and perturbations in the form of parametric changes and/or impulse inputs.

Chaos has been studied in great details and initially observed in several natural phenomena like weather fluctuations and climate changes [2]. The presence of chaos has also been seen in several other natural processes like heartbeat rhythms, which has helped physicians and scientists to make predictions on patient health [3], and anthropological studies to mathematically understand human evolution [4]. Chaotic nature also shows up in human-engineered processes such as urban traffic [5] and stock market [6]. Chaos is also seen in physical systems [7] starting from the simple double-pendulum going all the way up to thermo-fluid processes, electronic circuits, and even the fairground Tilt-a-Whirl.

Chaotic systems do follow a deterministic map (often with additive noise), where the initial conditions, process parameters, excitation inputs, and exogenous disturbances may significantly affect the time evolution of system responses. The *map* of a chaotic dynamical system may show significantly different nature for different values of process parameters. The state to which the system converges for a given set of parameters is called an *attractor* for that regime, which remains locally unchanged irrespective of the initial conditions or exogenous perturbations. There are essentially three types of attractors: (i) fixed-point attractors, where the system function converges to a single value, (ii) limit cycles, where the system shows a periodic nature, and (iii) strange attractors, where the system shows complex dynamical behavior that may be locally unstable and yet globally stable, implying that the system output may diverge but never significantly deviate from the attractor. This third kind of attractor is of much interest in the field of chaos.

It is often important to be able to distinguish noise from chaos and also to discriminate between the behaviors of a strange attractor and a limit cycle, which is often a challenge because the differences between these states may not be very apparent in the signal form, because of the presence of *hidden* dynamics and dependencies. Much work has been reported in literature in an attempt to make these distinctions based on time series data from various sources.

The 0–1 test method [8] has been introduced in the last decade to classify a given deterministic dynamical system as chaotic or non-chaotic. This method has been used extensively to study chaos in dynamical systems, including experimental plants (e.g., Ref. [9]). Djurovíc and Rubežić [10] proposed a different approach by using short-time Fourier transform (STFT) to distinguish between chaotic signals and periodic oscillations. Recently, the concept of a simple decision tree structure is introduced by Toker et al. [11] to distinguish between non-chaotic and chaotic signals, which modifies the standard 0–1 test to show its applicability to several standard chaotic systems as well as to biological data.

Nearly all of the above techniques involve binary detection tests to study time series data from a system and decide whether it is chaotic or not. Few other researchers have attempted to classify the type of chaos but not with much rigor, because chaos occurs in complex dynamical processes, and it is not easy to generate a single test method to classify the various types of chaos occurring in the signal.

This study proposes regime classification in chaotic dynamical systems in the setting of a standard supervised data-driven method, called hidden Markov modeling (HMM) [12]. Several other well-known data-driven methods are potentially applicable

Manuscript received April 3, 2020; final manuscript received July 9, 2020; published online August 5, 2020. Assoc. Editor: Reza Tafreshi.

for regime classification; examples are symbolic time series analysis (STSA) [13] and deep neural networks (DNN) using long short-term memory (LSTM) [14]. However, HMMs provide a good trade off with relatively smaller training times compared to that in LSTM-based neural networks with similar accuracy and may have higher accuracy than traditional STSA methods. Thus, in this study, the authors have chosen to demonstrate the efficacy of an HMM classifier in the context of performing the difficult task of discriminating multiple regimes in chaotic systems. To the best of authors' knowledge, there is no reported literature that reports classification of different regimes of a chaotic dynamical system.

The primary objective here is not only to detect the presence of chaos but also to classify the operational regimes of the system, after the classifier is trained with the knowledge of the regimes (i.e., in a supervised fashion). As a classifier, HMMs are capable of identifying the current regime of the system, regardless of whether the regime is chaotic or non-chaotic. In the event that the dynamical system is needed to be controlled, this information could help in choosing a better control action to steer the system to a preferred state. Efficacy of the HMM-based regime classifier is demonstrated on several well-known chaotic dynamical systems that are sensitive to initial conditions and parametric perturbations and that have very different structures of attractors depending on their model parameters.

Contributions: Major contributions of the work reported in this study are as follows:

(1) *Development of a HMM-based regime classifier for chaotic dynamical systems*: The underlying algorithm takes time-series signals as inputs for regime classification and does not require the information on whether the classified regimes are chaotic or not. The output is the identified regime based on previously trained regime class data.
(2) *Efficacy demonstration of the above HMM-based regime classifier*: The classification algorithm has been validated on time series data, generated from four chaotic dynamical systems—Hénon map, forced Duffing, Rössler, and Lorenz attractor.

Organization: The study is organized in five sections. Section 2 introduces the essential concepts of chaos along with a mathematical description of four different chaotic system models that are investigated here. Section 3 provides a brief background of HMM that is the backbone of the regime classification algorithm in this study. Section 4 presents the results for various cases of each of the four chaotic systems. Section 5 summarizes and concludes the reported work along with a few recommendations for future research.

2 Chaotic Dynamical Systems

Chaotic dynamical systems are essentially deterministic; they are sensitive to the initial conditions, only for chaotic regimes, and to system parameters for (possibly) all regimes. This study focuses on chaotic dynamical system models with parameters that can be varied to obtain multiple regimes of operation, including both chaotic and non-chaotic. This section presents the four chaotic dynamical systems used to generate time-series data from various operating regimes, obtained by varying the parameters (see Table 1 for details).

The first chaotic system is the Hénon map [15] that has the following set of governing equations:

$$\begin{aligned} x_{n+1} &= 1 - ax_n^2 + y_n \\ y_{n+1} &= bx_n \end{aligned} \tag{1}$$

The second chaotic system is the forced Duffing system [16], representing a nonlinear spring, whose governing equation is given as

$$\ddot{x} + \delta\dot{x} + \alpha x + \beta x^3 = \gamma \cos \omega t \tag{2}$$

The third chaotic system is the Rössler attractor [17], which represent chemical reaction kinetics, has the following set of three coupled first-order differential equations:

$$\begin{aligned} \dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x - c) \end{aligned} \tag{3}$$

The fourth chaotic system is the Lorenz attractor which is derived by reducing the Navier–Stokes equation [1] and has the following

Table 1 Results of case studies for the different chaotic maps; listing the fixed and varying parameters and initial conditions for training and testing data; the last (right hand) column indicates the total HMM classification error

Case	Chaotic system	Fixed parameter values	Varying parameter values	Initial conditions for training data	Initial conditions for testing data	HMM error
I	Henon	$b = 0.3$	$a = 1$ to 1.4 in increments of 0.05	$(x_0, y_0) = (1, 1)$	$(x_0, y_0) = (1, 1)$	**0.97%**
I-A	Henon	$b = 0.3$	$a = 1$ to 1.4 in increments of 0.05	$(x_0, y_0) = (1, 1)$	$(x_0, y_0) = (0, 0)$	**12.24%**
II	Duffing	$\gamma = 0.2$, $\delta = 0.5$, $\alpha = 1$, $\beta = -1$	$\omega = 0$ to 2 in increments of 0.5	$(x_0, \dot{x_0}) = (0, 0)$	$(x_0, \dot{x_0}) = (0, 0)$	**0%**
II-A	Duffing	$\gamma = 0.2$, $\delta = 0.5$, $\alpha = 1$, $\beta = -1$	$\omega = 0$ to 2 in increments of 0.5	$(x_0, \dot{x_0}) = (0, 0)$	$(x_0, \dot{x_0}) = (0.1, 0)$	**0%**
III	Duffing	$\alpha = -1$, $\beta = 1$, $\delta = 0.3$, $\omega = 1.2$	$\gamma = 0.2$ to 0.60 in increments of 0.1	$(x_0, \dot{x_0}) = (0, 0)$	$(x_0, \dot{x_0}) = (0, 0)$	**4.48%**
IV	Duffing	$\alpha = -1$, $\beta = 1$, $\delta = 0.3$, $\omega = 1.2$	$\gamma = 0.2$, 0.28, 0.29, 0.37, 0.50 & 0.65	$(x_0, \dot{x_0}) = (0, 0)$	$(x_0, \dot{x_0}) = (0, 0)$	**1.11%**
V	Duffing	$\alpha = 1$, $\beta = 1$, $\gamma = 22$, $\omega = 5$	$\delta = 0.1$ to 0.35 in increments of 0.05	$(x_0, \dot{x_0}) = (0, 0)$	$(x_0, \dot{x_0}) = (0, 0)$	**30.77%**
VI	Rossler	$b = 2$, $c = 4$	$a = 0.25$ to 0.55 in increments of 0.05	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (1, 1, 1)$	**1.69%**
VI-A	Rossler	$b = 2$, $c = 4$	$a = 0.25$ to 0.55 in increments of 0.05	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (0, 1, 1)$	**1.69%**
VII	Rossler	$b = 2$, $c = 4$	$a = 0.3547$, 0.398, 0.410, 0.458, 0.503, and 0.550	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (1, 1, 1)$	**1.17%**
VIII	Rossler	$a = 0.2$, $c = 5.7$	$b = 0.2$ to 1.8 in increments of 0.4	$(x_0, y_0, z_0) = (0, 0, 0)$	$(x_0, y_0, z_0) = (0, 0, 0)$	**0.85%**
IX	Rossler	$a = 0.1$, $b = 0.1$	$c = 5$ to 45 in increments of 5	$(x_0, y_0, z_0) = (0.2, 0.3, 0.2)$	$(x_0, y_0, z_0) = (0.2, 0.3, 0.2)$	**0.10%**
X	Lorenz	$\sigma = 10$, $\beta = 8/3$	$\rho = 50$ to 250 in increments of 20	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (1, 1, 1)$	**0%**
X-A	Lorenz	$\sigma = 10$, $\beta = 8/3$	$\rho = 50$ to 250 in increments of 20	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (0, 1, 1)$	**0%**
XI	Lorenz	$\sigma = 10$, $\beta = 8/3$	$\rho = 13$, 14, 15, 28, 99.96, 100.3, 155, 193, 228, 229, 230.5, and 240	$(x_0, y_0, z_0) = (1, 1, 1)$	$(x_0, y_0, z_0) = (1, 1, 1)$	**0%**

governing equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x)\\ \dot{y} &= x(\rho - z) - y\\ \dot{z} &= xy - \beta z\end{aligned} \tag{4}$$

Depending on the parametric values in Eqs. (1)–(4), the evolution of the system state variables may drastically change. Therefore, it is necessary to understand, especially for regime classification, exactly which parametric condition a system is currently in. Some of the challenges related to these phenomena are exemplified in Figs. 1 and 2.

Figure 1 shows the bifurcation diagram for the Rössler system in Eq. (3) by varying the parameter c, while the other system parameters are kept fixed at $a = 0.1$ and $b = 0.1$. It is seen that as c is varied, the system behavior changes drastically, with different attractor basins for different ranges of c. Non-chaotic attractors cause the system to come to respective equilibria, irrespective of initial conditions; for chaotic attractors, the initial conditions may play a major role in the system evolution while the chaotic attractor remains the same irrespective of the initial conditions. Figure 2 shows the phase plots of signals for nine selected values of the parameter c along with a mention of the type of system regime that it belongs to. It is seen that changes of the parameter c from 8.5 to values in the range of 8.7–9 alters the system behavior from periodic with period 4, to periodic with period 8 to chaotic. This demonstrates possible sensitivity of a chaotic system to certain parameters. The plots in Figs. 1 and 2 show that it is not always easy to distinguish between the regimes by traditional analysis of time-series data, and perhaps that is why it is not reported in the current literature.

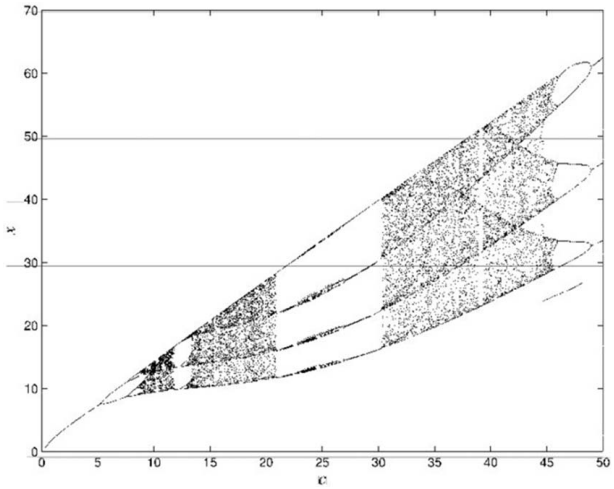

Fig. 1 Bifurcation diagrams of the Rössler system for the varying parameter "c," from Alvarez et al. [18]: $a = 0.1$, $b = 0.1$, $(x_0, y_0, z_0) = (0, 0, 0)$

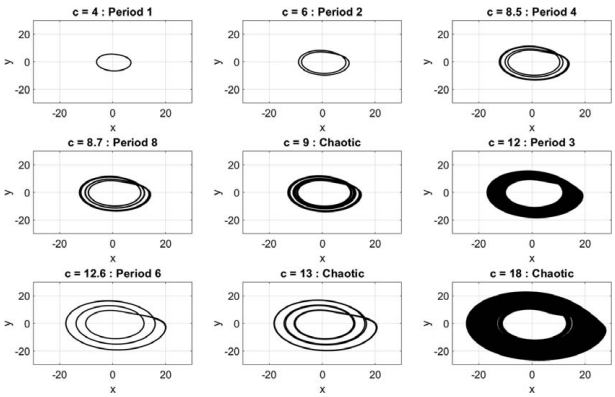

Fig. 2 Phase diagrams of the Rössler system for the varying parameter c indicating the system regime for that value: $a = 0.1$, $b = 0.1$, $(x_0, y_0, z_0) = (0, 0, 0)$

Section 3 addresses the usage of HMM [12] for detection and classification of the exact regime of operation by analyzing the time series of system responses.

3 Hidden Markov Modeling

While the details of HMM are extensively reported in technical literature (e.g., Refs. [12,19]), this section briefly introduces the underlying concept for completeness of this study. HMMs have found applications primarily in speech recognition [20], time series classification [21] and even in image classification [22]. In all of these problems, the HMM method has shown high classification accuracy. Since HMM is a well-known and well-reported technique, the mathematical details are not presented here in detail for brevity; interested readers are referred to the cited references.

In the training phase of a HMM [12], time series data from each regime, i.e., classes $k = 1, \ldots, \mathcal{K}$, are used to learn the HMM. Subsequently, in the testing phase, the learned HMMs are compared to the HMM trained from the unknown regime. A succinct description of HMM is outlined below.

During training, the *Baum-Welch algorithm* [12,19] has been used to learn the HMM models of $\mathcal{K}$ classes. For each class (regime), an ensemble of time-series windows is obtained from the available time-series data. This is done by first downsampling the long data with a downsampling rate DS, and then taking data windows of length L shifted by a distance S; the parameters, DS, L, and S are user-specified [23]. Then, the *Baum-Welch algorithm* [12,19] is applied to train the HMM λ_k which is a triplet $\lambda_k = \{A_k, B_k, \pi_k\}$ [12], where

(1) $A \triangleq [a_{ij}]$ is the $N \times N$ state-transition probability matrix where N is the assumed finite number of hidden states belonging to a set of symbols $\in \mathcal{Q}$:

$$a_{ij} = p(z_{t+1} = q_j | z_t = q_i)\colon\ q_i, q_j \in \mathcal{Q}$$

where $\sum_j a_{ij} = 1\ \forall i$.

(2) $B \triangleq [b_j(y_t)]$ is the probability density of the observation given the state:

$$b_j(y_t) = p(y_t | z_t = q_j)$$

(3) $\pi \triangleq [\pi_i]$ is the probability distribution of the initial state z_1: $\pi_i = p(z_1 = q_i)$, where π is a $1 \times N$ vector with $\sum_i \pi_i = 1$.

This procedure is repeated for each of the $\mathcal{K}$ classes.

During the testing phase, data windows of the signal from an unknown regime, with the same parameters, DS, L, and S, as in the training phase, are provided as inputs to the algorithm. Then, the *Forward Procedure* [12,19] of HMM is used to compute the log likelihood (L_k) of a given window of unknown time series data belonging to each of the $\mathcal{K}$ classes. The final decision, as to which class the unknown data belongs, is made by selecting the class with the largest log likelihood as follows:

$$\text{Selected Class} = \underset{k \in \{1,2,\ldots,\mathcal{K}\}}{\operatorname{argmax}}\ L_k \tag{5}$$

In this study, a continuous HMM formulation has been used, where the emission is assumed to follow a Gaussian mixture model with M Gaussian components and N hidden states. The algorithms and theory for all the above HMM procedures are available in the literature (e.g., Refs. [12,19]).

Remark 1. Although dynamical systems may have multiple regimes of operation, a real-life system would probably show only a few chaotic and periodic regimes. The HMM-based

classification should be applicable to those cases as a subset of the dynamical systems demonstrated in this study.

4 Results and Discussions

This section presents the results of analysis for each of the four chaotic dynamical systems (see Eqs. (1)–(4)). Multiple cases are considered, each having a different set of fixed and varying parameters. For each case, time-series data are obtained numerically for the system parameters using a set of initial conditions of the states. In some cases, the initial conditions are varied to examine robustness of the HMM-based classifier to capture the attractor behavior when the initial conditions are perturbed in the chaotic system.

In the Hénon map (see Eq. (1)), the time evolution is accomplished by forward integration. For Duffing, Rössler and Lorenz systems (see Eqs. (2)–(4)), the signals are generated using the fourth-order Runge–Kutta formulation at a fixed time-step size of 1×10^{-3} s. Data of 5000 s have been generated corresponding to each parameter set, for both training and testing phases, yielding 5,000,000 data points per condition. Furthermore, the signals are artificially contaminated with 10% noise (amplitude ratio formulation), which corresponds to a signal-to-noise ratio (SNR) of 20 dB.

For both HMM training and testing (see Sec. 3), the data are first downsampled by $DS=100$. This is necessary because the time-step of integration is intentionally made very small, which leads to oversampling. The processed data are then windowed with a moving window formulation (see Sec. 3) with a window size of $L=1000$ and a shift value of $S=100$ data points. The parameters DS, L, and S are held constant across all training and test cases.

The HMM parameters, namely, number of Gaussian components (M) and number of hidden states (N), also need to be assigned. The optimal values for each system of equations are slightly different. To allow for easy comparison, a common set of values are taken and then kept unchanged for all tested cases; these parameters are $M=4$ and $N=30$. Higher values of N degrade the performance due to over-fitting and needing more data for proper training. Also computational time increases with increasing N.

Table 1 lists the results for each case, which show both fixed and varying parameters for individual systems. The initial conditions for generation of training and testing data are also given. These parameters and their ranges are taken from cited research publications (e.g., Refs. [1,13,15–17,24,25]) and are reported here for reproducibility of the results.

For the same map, multiple cases have been presented with different fixed and varying parameters. The last (right hand) column in Table 1 provides the overall classification accuracy for various classes, where the varying data are used for both training and testing. The total error is defined as the ratio of the number of data windows erroneously classified, to the total number of data windows analyzed.

Majority of the cases are trained and tested using data from a range of values of the varying parameter, as obtained from literature or from the bifurcation diagrams of the dynamical systems. For cases I, II, VI, and X in Table 1, a second analysis is done, where the initial conditions for the testing data are made different from those of the training data. These cases are appended with "-A" in the first (left hand) column of Table 1. Certain cases, namely, IV, VII, and XI, have been analyzed using specific values of the varying parameter. These values are taken from literature as regions of interest and have been analyzed for illustrating the strengths and weaknesses of HMM-based classification. In general, it is seen that HMM-based classification yields low error rates (e.g., less than 2%). However, it is important to know why HMM-based classification performs somewhat poorly for cases I-A, III, and V, as explained below.

The poor performance in Case I-A is explained by the fact that, as these systems are chaotic, changing the initial conditions causes the system behavior to change significantly. This possibly leads to incorrect classification, especially for the regions that have *strange attractors*.

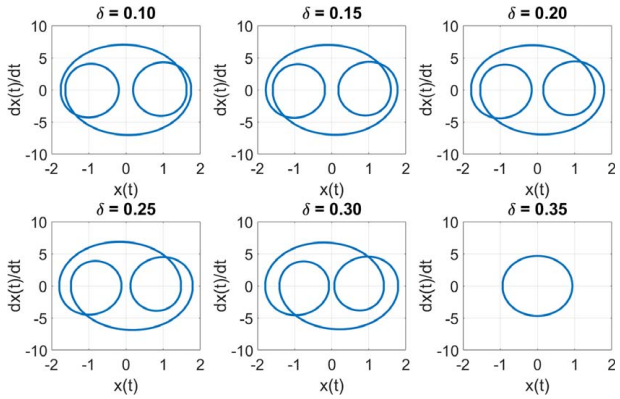

Fig. 3 Phase plots for the generated signal for $\delta=0.10$ to 0.35 in increments of 0.05, where $\alpha=1$, $\beta=1$, $\gamma=22$, $\omega=5$

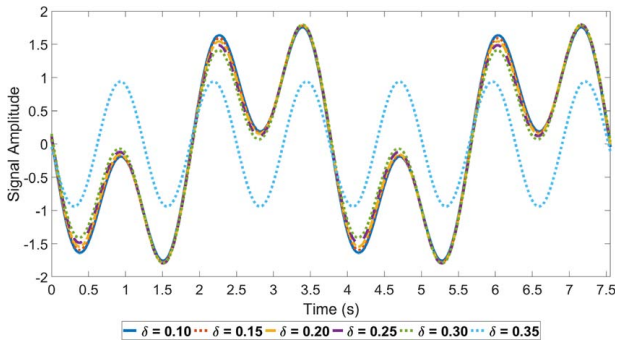

Fig. 4 Phase-aligned noise-less signal for $\delta=0.10$ to 0.35 in increments of 0.05, where $\alpha=1$, $\beta=1$, $\gamma=22$, $\omega=5$

For case III, the chosen range of values of γ is such that several values correspond to signals coming from the same regime with possibly a minor difference in the signal form (e.g., $\gamma=0.3$ and $\gamma=0.4$ have very similar signals); this leads to poor classification. However, in the next case IV, the same fixed parameters are used, with the values of the varying parameter chosen to match regions where the signal nature varies. Here, a better classification accuracy is achieved.

In case V, the signals are from a region where the parameter δ ranges from 0.1 to 0.35. For $\delta=0.1$ to 0.3, the time signals are very similar, and the signal bifurcates to a different regime when $\delta\approx0.35$, as seen in Figs. 3 and 4. Here too, HMM-based classification is unable to capture the difference between the signals when $\delta=0.1$ to 0.3. Thus, in this region of interest [13], the HMM method is apparently insufficient.

5 Summary, Conclusions, and Future Work

This study has demonstrated the applicability of the HMM-based time-series analysis in an attempt to classify regimes of operation in chaotic dynamical systems, which is not apparently explored much in literature. The proposed data-driven regime classification does not need the knowledge of whether the classified regime is chaotic or non-chaotic; this property has been demonstrated by considering various operational cases from four standard chaotic maps, namely, Hénon, Duffing, Rössler, and Lorenz attractor systems. The study has been conducted on global ranges as well as local ranges of interest near bifurcation zones.

The proposed HMM-based method of regime classification yields good accuracy, in general, and it performs sub-par only for a few specific cases, the possible causes for which are discussed

in Sec. 4. Thus, this method, coupled with the standard tests for chaotic nature could help not only to identify chaos but also to classify chaotic "sub"-regimes.

The above results evince that the proposed HMM-based method is expected to be well-suited to serve as a bench mark for development of alternative data-driven methods of regime classification in chaotic dynamical systems.

While there are many areas of research to enhance the work reported in this study, the following topics are suggested to be conducted in the near future.

(1) Development of other data-driven methods (such as Symbolic Time Series Analysis [13,26]) that can improve upon the accuracy of HMM-based classification, especially in *closeby* regions of interest, typically seen just prior to bifurcation.
(2) Enhancement of robustness of data-driven regime classification to initial conditions of chaotic systems.
(3) Application of the proposed classification method to real-life systems that display multiple chaotic regimes.
(4) Development of a data-driven method to identify chaos type with no or partial knowledge of system dynamics (e.g., unsupervised or semi-supervised classification).

Acknowledgment

The reported work has been supported in part by the U.S. Air Force Office of Scientific Research (AFOSR) under Grant No. FA9550-15-1-0400 in the area of dynamic data-driven application systems (DDDAS). Any opinions, findings and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the sponsoring agencies.

Conflict of Interest

There are no conflicts of interest.

References

[1] Lorenz, E., 1972, "Predictability: Does the Flap of a Butterfly's Wing in Brazil Set Off a Tornado in Texas?," American Association for the Advancement of Sciences, 139th Meeting, Washington, DC, Dec. 28.
[2] Lorenz, E., 1963, "Deterministic Nonperiodic Flow," J. Atmos. Sci., **20**(2), pp. 130–141.
[3] Wu, G.-Q., Arzeno, N., Shen, L.-L., Tang, D.-K., Zheng, D.-A., Zhao, N.-Q., Eckberg, D.-L., and Poon, C.-S., 2009, "Chaotic Signatures of Heart Rate Variability and Its Power Spectrum in Health, Aging and Heart Failure," PLoS One, **4**(2), pp. 1–9.
[4] Mosko, D., 2005, *On the Order of Chaos: Social Anthropology and the Science of Chaos*, Berghahn Books, Brooklyn, NY.
[5] Dendrinos, D., 1994, "Traffic-Flow Dynamics: A Search for Chaos," Chaos, Solitons Fractals, **4**(4), pp. 605–617.
[6] Hsieh, D., 1991, "Chaos and Nonlinear Dynamics: Application to Financial Markets," J. Finance, **46**(5), pp. 1839–1877.
[7] Chen, G., 1999, *Controlling Chaos and Bifurcations in Engineering Systems*, CRC Press, Boca Raton, FL.
[8] Gottwald, G., and Melbourne, I., 2004, "A New Test for Chaos in Deterministic Systems," Proc. R. Soc. Lond. Ser. A: Math. Phys. Eng. Sci., **460**(2042), pp. 603–611.
[9] Gopal, R., Venkatesan, A., and Lakshmanan, M., 2013, "Applicability of 0-1 Test for Strange Nonchaotic Attractors," Chaos: Interdiscip. J. Nonlinear Sci., **23**(2), p. 023123.
[10] Djurovíc, I., and Rubežić, V., 2008, "Chaos Detection in Chaotic Systems With Large Number of Components in Spectral Domain," Sig. Process., **88**(9), pp. 2357–2362.
[11] Toker, D., Sommer, F., and D'Esposito, M., 2020, "A Simple Method for Detecting Chaos in Nature," Commun. Biol., **3**(11), pp. 1–13.
[12] Rabiner, L., 1989, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," Proc. IEEE, **77**(2), pp. 257–286.
[13] Ray, A., 2004, "Symbolic Dynamic Analysis of Complex Systems for Anomaly Detection," Sig. Process., **84**(7), pp. 1115–1130.
[14] Karim, F., Majumdar, S., Darabi, H., and Chen, S., 2017, "LSTM Fully Convolutional Networks for Time Series Classification," IEEE Access, **6**, pp. 1662–1669.
[15] Hénon, M., 1976, "A Two-Dimensional Mapping With a Strange Attractor," *The Theory of Chaotic Attractors*, Springer, New York, NY, pp. 94–102.
[16] Thompson, J., and Stewart, H., 1986, *Nonlinear Dynamics and Chaos*, 2nd ed., John Wiley and Sons, Somerset, NJ, USA.
[17] Rössler, O., 1976, "An Equation for Continuous Chaos," Phys. Lett. A, **57**(5), pp. 397–398.
[18] Alvarez, G., and Li, S., 2006, "Some Basic Cryptographic Requirements for Chaos-Based Cryptosystems," Int. J. Bifurcation Chaos, **16**(08), pp. 2129–2151.
[19] Hajek, B., 2015, *Random Processes for Engineers*, 1st ed., Cambridge University Press, Cambridge, UK.
[20] Najkar, N., Razzazi, F., and Sameti, H., 2010, "A Novel Approach to HMM-Based Speech Recognition Systems Using Particle Swarm Optimization," Math. Comput. Modell., **52**(11), pp. 1910–1920. The BIC-TA 2009 Special Issue.
[21] Oates, T., Firoiu, L., and Cohen, P., 2000, "Using Dynamic Time Warping to Bootstrap HMM-based Clustering of Time Series," *Sequence Learning*, R. Sun, and C. L. Giles, eds., Springer, New York, NY, pp. 35–52.
[22] Ali, S. S., and Ghani, M. U., 2014, "Handwritten Digit Recognition Using DCT and HMMs," 2014 12th International Conference on Frontiers of Information Technology, Islamabad, Pakistan, Dec. 17–19, pp. 303–306.
[23] Ghalyan, N., and Ray, A., 2020, "Symbolic Time Series Analysis for Anomaly Detection in Measure-Invariant Ergodic Systems," ASME J. Dyn. Syst. Meas. Control, **142**(6), p. 061003.
[24] Jordan, D., and Smith, P., 2007, *Nonlinear Ordinary Differential Equations: An Introduction for Scientists and Engineers*, Vol. 10, Oxford University Press on Demand, Oxford, UK.
[25] Asghari, H., and Dardel, M., 2020, "Parameter Converting Method for Bifurcation Analysis of Nonlinear Dynamical Systems," Sci. Iranica, **27**(1), pp. 310–329.
[26] Mukherjee, K., and Ray, A., 2014, "State Splitting and Merging in Probabilistic Finite State Automata for Signal Representation and Analysis," Sig. Process., **104**, pp. 105–119.