

Dynamic data-driven prediction of lean blowout in a swirl-stabilized combustor

Soumalya Sarkar[†], Asok Ray[†], Achintya Mukhopadhyay[‡] and Swarnendu Sen[‡]

[†]Department of Mechanical & Nuclear Engineering, Pennsylvania State University, University Park, PA 16802, USA

[‡]Department of Mechanical Engineering, Jadavpur University, Kolkata 700 032, India

(Submission date: June 21, 2014; Revised Submission date: October 29, 2014; Accepted date: December 19, 2014)

ABSTRACT

This paper addresses dynamic data-driven prediction of lean blowout (LBO) phenomena in confined combustion processes, which are prevalent in many physical applications (e.g., land-based and aircraft gas-turbine engines). The underlying concept is built upon pattern classification and is validated for LBO prediction with time series of chemiluminescence sensor data from a laboratory-scale swirl-stabilized dump combustor. The proposed method of LBO prediction makes use of the theory of symbolic dynamics, where (finite-length) time series data are partitioned to produce symbol strings that, in turn, generate a special class of probabilistic finite state automata (PFSA). These PFSA, called D -Markov machines, have a deterministic algebraic structure and their states are represented by symbol blocks of length D or less, where D is a positive integer. The D -Markov machines are constructed in two steps: (i) state splitting, i.e., the states are split based on their information contents, and (ii) state merging, i.e., two or more states (of possibly different lengths) are merged together to form a new state without any significant loss of the embedded information. The modeling complexity (e.g., number of states) of a D -Markov machine model is observed to be drastically reduced as the combustor approaches LBO. An anomaly measure, based on Kullback-Leibler divergence, is constructed to predict the proximity of LBO. The problem of LBO prediction is posed in a pattern classification setting and the underlying algorithms have been tested on experimental data at different extents of fuel-air premixing and fuel/air ratio. It is shown that, over a wide range of fuel-air premixing, D -Markov machines with $D > 1$ perform better as predictors of LBO than those with $D = 1$.

Keywords: Data-driven Dynamics, Lean Blowout, Gas Turbine Combustor, Symbolic Dynamics, Probabilistic Finite State Automata

*Corresponding author email: sv5464@psu.edu; axr2@psu.edu; achintya.mukho@gmail.com; sen.swarnendu@gmail.com

1. INTRODUCTION

Ultra-lean combustion is commonly used for reduction of oxides of nitrogen (NO_x) and is susceptible to thermo-acoustic instabilities and lean blowout (LBO). It is well known that occurrence of LBO could be detrimental for operations of both land-based and aircraft gas turbine engines. For example, LBO in land-based gas turbines may lead to engine shutdown and subsequent re-ignition involves loss of productivity; similarly, LBO in aircraft gas turbines may cause loss of engine thrust especially if the fuel flow is suddenly reduced during a throttling operation, or if air flow reduction takes place at a much slower rate due to the moment of inertia of the compressor. In essence, a sudden decrease in the equivalence ratio may lead to LBO in gas turbine engines, which could have serious consequences. This phenomenon calls for a real-time prediction of LBO and adoption of appropriate measures to mitigate it.

It is noted that a priori determination of the LBO margin may not be feasible, because of flame fluctuations in the presence of thermoacoustic instability. To address this issue, the current paper develops an online LBO prediction tool for both well-premixed (e.g., in land-based gas turbines) and partially premixed (e.g., in aircraft gas turbines) combustors. The LBO limit is dependent on a number of parameters that are related to the combustor configuration and operating conditions, and monitoring of all these parameters would require a complicated and expensive diagnostic system. From this perspective, quantifiable dynamic characteristics of flame preceding blowout have been exploited in the past as LBO precursors by a number of researchers. De Zilwa *et al.* [1] investigated the occurrence of blowout in dump combustors with and without swirl. Chaudhuri and Cetegen [2], [3] investigated the effects of spatial gradients in mixture composition and velocity oscillations on the blowoff dynamics of a bluff-body stabilized burner that resembles representative of afterburners of gas turbine combustors. They used photomultiplier tubes with CH chemiluminescence filters to capture the optical signal and characterized the signal in the vicinity of blowout. Chaudhuri *et al.* [4] and Stohr *et al.* [5] investigated LBO dynamics of premixed and partially premixed flames using combined particle image velocimetry/planar laser-induced fluorescence (PIV/PLIF)-based diagnostics. However, these works did not focus on developing strategies for mitigating LBO.

Lieuwen, Seitzman and coworkers [6], [7], [8], [9] used time series data from acoustic and optical sensors for early detection and control of LBO in laboratory-scale gas turbine combustors. Nair and Lieuwen [8] identified blowout parameters using various (e.g., spectral, statistical, wavelet-based and threshold-based) techniques. For statistical analysis, they used moving average kurtosis. They defined thresholds in terms of cut-offs for peak pressure in a cycle falling below. The number of such events increased sharply near blowout. For wavelet-based analysis, they used the Mexican Hat and a customized wavelet that matches with the time series data of OH* chemiluminescence. Yi and Gutmark [10] identified two indices, namely, normalized chemiluminescence root mean square and normalized cumulative duration of LBO precursor events, for LBO prediction in real time. However, both Lieuwen *et al.* and Gutmark demonstrated their techniques for LBO prediction in premixed combustors. On the other hand, Mukhopadhyay and coworkers [11], [12] developed a number of techniques for early detection of LBO, which

worked satisfactorily over a wide range of fuel-air premixing. Chaudhari *et al.* [11] used flame color to devise a novel and inexpensive strategy for LBO detection. In this work, they used as an LBO detection metric, the ratio of red and blue intensities in the flame image obtained with a commercial color charge-coupled device (CCD) camera. Mukhopadhyay *et al.* [12] used chemiluminescence time series data for online prediction of LBO under both premixed and partially premixed operating conditions. The algorithm was built upon a data-driven symbolic-dynamics-based technique, called *D*-Markov machine [13], [14], where the parameter *D* was set to 1 to construct the probabilistic finite state automata (PFSA), which implies that the memory of the underlying combustion process was restricted only to its immediate past state.

In a number of recent publications, the transition to LBO was correlated with changes in signature of the dynamic characteristics of the system. Kabiraj *et al.* [15] and Kabiraj and Sujith [16] investigated dynamics of a ducted laminar premixed flame and using tools of dynamical systems analysis demonstrated that the system undergoes a number of bifurcations involving quasiperiodic and intermittent behaviors leading to blowout. Gotoda and co-workers [17] used a number of analytical tools to examine the dynamics of a model gas turbine combustor and showed that close to lean blowout the dynamics of the combustor exhibits a self-affine structure indicating fractional Brownian motion but changes to chaotic oscillations showing oscillations with slow amplitude modulation as the equivalence ratio increases. Gotoda *et al.* [18] showed that the relatively regular pressure oscillations undergo transition to intermittent chaotic oscillations as the equivalence ratio is decreased and the system approaches LBO. Translational errors were used to measure the parallelism of neighboring trajectories in the phase space as a control variable for mitigation of LBO. Through similar studies, Sujith and co-workers [19], showed that combustion noise is due to chaotic fluctuations of moderately high dimension that undergoes transition to high amplitude oscillations characterized by periodic behavior. This information was used to determine the loss of chaos in the system dynamics as a precursor for prediction of thermoacoustic instability.

The current paper is a major extension of the earlier work on *D*-Markov machine-based LBO prediction, reported by Mukhopadhyay *et al.* [12]. The significant contributions of the current paper in the context of dynamic data-driven application systems (DDDAS) [20] are delineated below.

- 1) *State splitting and state merging*: The states of the *D*-Markov machine are constructed in two steps [14]: (i) state splitting, i.e., the states are split based on their information contents, and (ii) state merging, i.e., two or more states (of possibly different lengths) are merged together to form a new state without any significant loss of their embedded information.
- 2) *Accommodation of a longer memory of chemiluminescence time series*: Algorithms of the *D*-Markov machine are reformulated with $D > 1$ (instead of $D = 1$) to extract low-dimensional features with a longer history of the combustion process.
- 3) *Bias removal to achieve leaner operating conditions*: The bias (i.e., the non-zero mean) is removed from chemiluminescence time series to avoid mean-based prediction that may result in richer operating conditions with consequent penalties of enhanced NO_x emission.

- 4) *Information-theoretic anomaly measure for LBO prediction*: An anomaly measure [13] is constructed, based on Kullback- Leibler divergence [21], to anticipate the proximity of LBO with increased sensitivity.
- 5) *Pattern classification based on the features extracted from chemiluminescence time series*: The prediction of LBO is posed as a pattern classification problem based on different ranges of the equivalence ratio at several premixing levels. This approach largely alleviates the problem of loss of robustness due to limited data availability for making online decisions [22].

The paper is organized in five sections, including the present one, and three appendices. Section 2 describes a laboratory- scale swirl-stabilized dump combustor, which has been used for prediction of LBO phenomena from time series of optical sensor data. Section 3 explains the method of information extraction from time series as a D -Markov machine in the form of low-dimensional features. Section 4 presents the capability and advantages of the proposed approach for LBO prediction over different parameter ranges. Finally, the paper is summarized and concluded in Section 5 with selected recommendations for future research. Appendix A explains the notion of two-time-scale behavior of combustion dynamics, which are captured from the sensor time series data. Appendix B explains in detail the key mathematical concepts that are necessary for construction of the pattern classification algorithms in the context of D -Markov machines. Appendix C lists the four algorithms of state splitting and state merging in the construction of D -Markov machines, from which the computer codes can be readily developed.

2. APPARATUS FOR EXPERIMENTAL RESEARCH

A swirl-stabilized dump combustor, as depicted in Figure 1, was designed as a laboratory-scale model of a generic gas turbine combustor based on the earlier works of Williams *et al.* [24], Meier *et al.* [25], Nair *et al.* [6], Chaudhuri and Cetegen [2] and Yi and Gutmark [10]. The air is supplied at the ambient temperature from a compressor to the bottom port on the premixing tube and the air flow rate is measured upstream of the combustor by using a calibrated mass flow controller (MFC) (Aalborg range 0 to 500 litres per minute (lpm)). Liquefied petroleum gas (LPG), having a composition of 40% propane (C_3H_8) and 60% butane (C_4H_{10}) by volume, has been used as the fuel in the experiments. The fuel is supplied from a pressurized cylinder fitted with needle valve to control the flow rate and is also measured upstream of the combustor by a calibrated Aalborg mass flow controller (Range: 0 – 10 lpm). To investigate the effects of premixing on flame dynamics, six side ports are provided 50 mm apart along the length of the premixing tube, as seen in Figure 1. This arrangement allows the fuel to be injected at different axial positions of the premixing tube, thereby providing different premixing lengths. The ports are numbered 1 to 6 from the bottom of the premixing tube. Thus the fuel injected through Port 1 allows greatest premixing while Port 6 allows least premixing. The fuel is injected to one of the side ports of the premixing tube. The fuel-air mixture enters the combustor through the inlet swirler in the annulus around a center body, located just prior to the dump plane in the premixing section. The inner diameter of the premixing section is 23 mm, and the diameter of the center-body

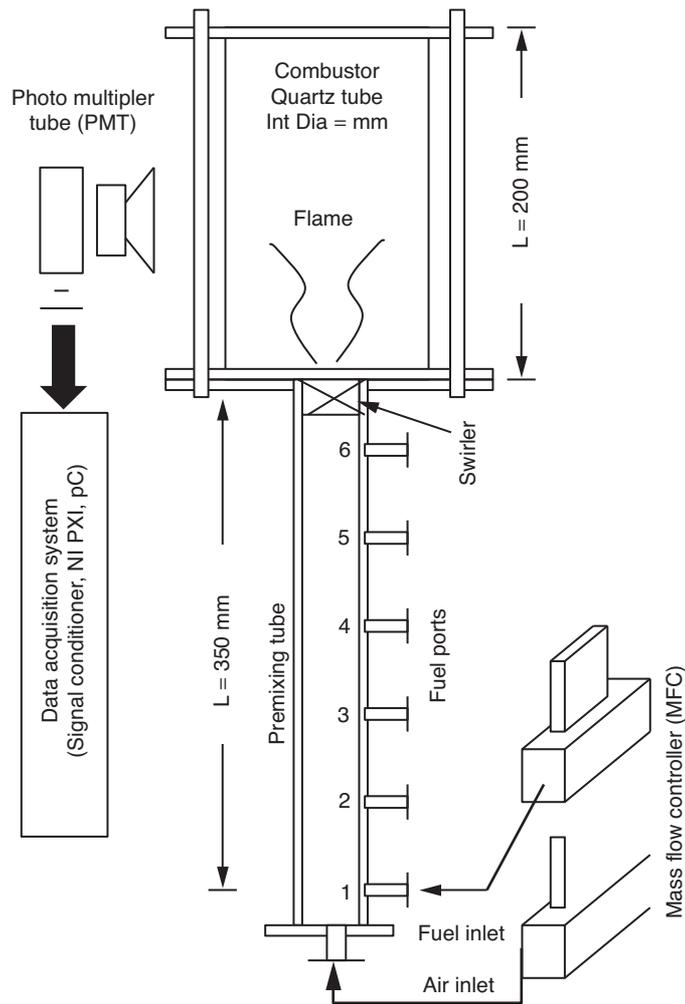


Figure 1: Schematic View of the Laboratory Apparatus.

is 8 mm. The inlet swirler has six vanes positioned at 60° to the flow axis. A quartz tube is provided in the combustion zone having the internal diameter 60 mm and length 200 mm to facilitate optical diagnostics.

The heat release rate is measured by the chemiluminescence emitted from the CH^* radicals (wavelength $\lambda \sim 431 \text{ nm}$) of the flame. The time series data are obtained with a photomultiplier tube (PMT) fitted with an optical band pass filter ($\lambda_{\text{pass}} = 430 \text{ nm}$) with full-width at half-maximum (FWHM) = 10 nm. The PMT output signal (in volts) is acquired using a 16-bit analog input channel on a National Instruments *PXI-6250* data acquisition card that is mounted in a National Instruments *PXI-1050* Chassis

having a built-in 08 channel *SCXI*-1125 signal conditioner module. Time series data with 2^{15} points have been acquired at a sampling frequency of 2 *kHz* in each experiment. Video images of the flame are recorded in order to visualize LBO phenomenology and correlate the same with the optical signal. Still color images of the flame are also acquired simultaneously using the digital single lens reflex (DSLR) camera at suitable exposure to avoid pixel saturation. Further details of the experimental apparatus and instrumentation are available in [23].

Experiments were carried out using liquefied petroleum gas (LPG) as the fuel. A major reason for the choice of LPG as the fuel is that LPG consists of mainly propane and butane, which are the simplest hydrocarbons whose combustion exhibits the chemical behavior, flame speed, and extinction limits closer to the heavier and more complex hydrocarbon fuels [26]. Tests were first conducted with stoichiometric fuel/air mixture (i.e., $\phi = 1$). Then, at each given air flow rate, the fuel supply was gradually decreased to generate progressively lean reacting mixtures. A constant air flow rate in the experiments was maintained to ensure that the Reynolds number of fluid flow remains practically constant because air constitutes the bulk of the incoming reactant mixture.

3. LBO PREDICTION VIA SYMBOLIC ANALYSIS

Symbolic time series analysis (STSA) [27] is built upon the concept of symbolic dynamics [28] that deals with discretization of dynamical systems in both space and time. The notion of STSA has led to the development of a feature extraction tool for pattern classification, in which a time series of sensor signals is represented as a symbol sequence that, in turn, leads to the construction of probabilistic finite state automata (PFSA) [29][30]. Statistical patterns of slowly evolving dynamical behavior in physical processes can be identified from sensor time series data and often the changes in these statistical patterns occur over a slow time scale with respect to the fast time scale of process dynamics. The concept of two time scales is succinctly presented in Appendix A.

Since PFSA models are capable of efficiently compressing the information embedded in sensor time series [13][31], these models could enhance the performance and execution speed of information fusion and information source localization that are often computation-intensive. Rao *et al.* [32] and Bahrampour *et al.* [33] have shown that the performance of this PFSA-based tool as a feature extractor for statistical pattern recognition is comparable (and often superior) to that of other existing techniques (e.g., Bayesian filters, Artificial Neural Networks, and Principal Component Analysis [34]). Mathematical preliminaries and background information on symbolic dynamics are presented in Appendix B.

The major steps for construction of PFSA from sensor signal outputs (e.g., time series) of a dynamical system are as follows.

- 1) Coarse-graining of time series to convert the scalar or vector-valued data into symbol strings, where the symbols are drawn from a (finite) alphabet [35].
- 2) Encoding (i.e., identification of labeled graph structure and pertinent parameters) of probabilistic state machines from the symbol strings [13], [14], [31], [22], [36].

The next step is to construct probabilistic finite state automata (PFSA) from the symbol strings to encode the embedded statistical information so that the dynamical system's behavior is captured by the patterns generated from the PFSA in a compact form. The algebraic structure of PFSA (i.e., the underlying FSA) consists of a finite set of states that are interconnected by transitions [37], where each transition corresponds to a symbol in the (finite) alphabet. At each step, the automaton moves from one state to another (including self loops) via these transitions, and thus generates a corresponding block of symbols so that the probability distributions over the set of all possible strings defined over the alphabet are represented in the space of PFSA. The advantage of such a representation is that the PFSA structure is simple enough to be encoded as it is characterized by the set of states, the transitions (i.e., exactly one transition for each symbol generated at a state), and the transition's probability of occurrence.

D-Markov machines are models of probabilistic languages where the future symbol is causally dependent on the (most recently generated) finite set of (at most) *D* symbols and form a proper subclass of PFSA with applications in various fields of research such as anomaly detection [13] and robot motion classification [38]. The underlying FSA in the PFSA of *D*-Markov machines are deterministic, i.e., the future state is a deterministic function of the current state and the observed symbol. Therefore, *D*-Markov machines essentially encode two entities: (1) probability of generating a symbol at a given state, and (2) deterministic evolution of future states from the current state and the symbol. It is noted that if the symbol is not observed, the description of the state transition in the *D*-Markov machine becomes a probabilistic Markov chain. Furthermore, under the assumption of irreducibility, the statistically stationary distribution of states is unique and can be computed. This class of PFSA (i.e., *D*-Markov machines) is a proper subclass of probabilistic non-deterministic finite state automata which is equivalent to Hidden Markov Models (HMMs) [29]. Even though HMMs form a more general class of models, the deterministic properties of *D*-Markov machines present significant computational advantages. For example, due to the constrained algebraic structure of *D*-Markov machines, it is possible to construct algorithms for efficient implementation and learning.

Since long-range dependencies in the time series rapidly diminish under the assumption of strong mixing [39], such a dynamical system could be modeled as a *D*-Markov machine with a sufficiently large value of *D*. However, increasing the value of *D* may lead to an exponential growth in the number of machine states and hence the computational complexity of the model grows. The main issue addressed in this paper is order reduction of the states of a *D*-Markov machine model for representing the stationary probability distribution of the symbol strings that are generated from the times series of a dynamical system [14]. In addition, this paper addresses the trade-off between modeling accuracy and model order, represented by the number of states, for the proposed algorithm. The power of the proposed tool of PFSA-based *D*-Markov machines is its capability of real-time execution on in-situ platforms for anomaly detection, pattern classification, condition monitoring, and control of diverse physical applications.

A. Construction of D -Markov Machines

This subsection develops the pertinent concepts that are necessary to construct a D -Markov machine. The PFSA model of the D -Markov machine generates symbol strings $\{s_1 s_2 \cdots s_\ell : \ell \in \mathbb{N}, \forall s_\ell \in \Sigma\}$ on the underlying Markov process. The generation of a symbol depends *only* on the current state in the PFSA structure of a D -Markov machine. However, if the state is unknown, the next symbol generation may depend on the history of the symbol generated by the PFSA. In the PFSA model, a transition from one state to another is independent of the previous states. Therefore, the states and transitions form a Markov process, which is a special class of Hidden Markov Models (HMM) [30]. However, from the perspectives of PFSA construction from a symbol sequence, the states are implicit and generation of the next symbol may depend on the complete history of the symbol sequence. In the construction of a D -Markov machine [13], 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 D . A formal definition of the D -Markov machine follows.

Definition 3.1 (*D -Markov Machine [13]*) *A D -Markov machine is a PFSA (see Definition B.5 in Appendix B) and it generates symbols that solely depend on the (most recent) history of at most D symbols in the sequence, where the positive integer D is called the depth of the machine. Equivalently, a D -Markov machine is a statistically stationary stochastic process $S = \dots s_{-1} s_0 s_1 \dots$, where the probability of occurrence of a new symbol depends only on the last D symbols, i.e.,*

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

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

Considering the set of all symbol blocks of length D as the set of states, one may construct a D -Markov machine from a symbol sequence by frequency counting to estimate the probabilities of each transition. Since the number of states increases exponentially as the depth D is increased, state merging might be necessary for order reduction of D -Markov machines with relatively large values of D .

Given a finite-length symbol sequence \mathbb{S} over a (finite) alphabet Σ , there exist several PFSA construction algorithms to discover the underlying irreducible PFSA model K of \mathbb{S} , such as causal-state splitting reconstruction (CSSR) [36] and D -Markov [13], [14]. All these algorithms start with identifying the structure of the PFSA $K \triangleq (Q, \Sigma, \delta, \pi)$ (see Appendix B). Then, to estimate the morph matrix, a $|Q| \times |\Sigma|$ count matrix C is initialized to the matrix, each of whose elements is equal to 1.

Let N_{ij} denote the number of times that a σ_j is generated from the state q_i upon observing the sequence \mathbb{S} . An estimate of the probability map for the PFSA K is computed by frequency counting as

$$\hat{\pi}(q_i, \sigma_j) \triangleq \frac{C_{ij}}{\sum_\ell C_{i\ell}} = \frac{1 + N_{ij}}{|\Sigma| + \sum_\ell N_{i\ell}} \quad (2)$$

The rationale for initializing each element of the count matrix C to 1 is that if no event is generated at a state $q \in Q$, then there should be no preference to any particular symbol and it is logical to have $\hat{\pi}(q, \sigma) = \frac{1}{|\Sigma|} \forall \sigma \in \Sigma$, i.e., the uniform distribution of event generation at the state q . The above procedure guarantees that the PFSA, constructed from a (finite-length) symbol string, must have an (elementwise) strictly positive morph map Π , as stated in Definition B.6 of Appendix B.

B. Algorithm Development

This subsection develops the algorithms for construction of D -Markov machines. The underlying procedure consists of two major steps, namely, *state splitting* and *state merging*. In general, state splitting increases the number of states to achieve more precision in representing the information content in the time series. This is performed by splitting the states that effectively reduce the entropy rate $H(\Sigma|Q)$, thereby focusing on the critical states (i.e., those states that carry more information). Although this process is executed by controlling the exponential growth of states with increasing depth D , the D -Markov machine still may have a large number of states. The subsequent process reduces the number of states in the D -Markov machine by merging those states that have similar statistical behavior. Thus, a combination of state splitting and state merging, described in Algorithms 1, 2, 3 and 4 in Appendix C leads to the final form of the D -Markov machine.

1) *The State Splitting Algorithm*: In D -Markov machines, a symbol block of (finite) length D is sufficient to describe the current state. In other words, the symbols that occur prior to the last D symbols do not affect the subsequent symbols observed. Therefore, the number of states of a D -Markov machine of depth D is bounded above by $|\Sigma|^D$, where $|\Sigma|$ is the cardinality of the alphabet Σ . For example, with the alphabet size $|\Sigma| = 4$ (i.e., 4 symbols in the alphabet Σ) and a depth $D = 3$, the D -Markov machine could have at most $|\Sigma|^D = 64$ states. As this relation is exponential in nature, the number of states rapidly increases as D is increased. However, from the perspective of modeling a symbol string, some states may be more important than others in terms of their embedded information contents. Therefore, it is advantageous to have a set of states that correspond to symbol blocks of different lengths. This is accomplished by starting off with the simplest set of states (i.e., $Q = \Sigma$ for $D = 1$) and subsequently splitting the current state that results in the largest decrease of the entropy rate.

The process of splitting a state $q \in Q$ is executed by replacing the symbol block q by its *branches* as described by the set $\{\sigma q : \sigma \in \Sigma\}$ of words. Maximum reduction of the entropy rate is the governing criterion for selecting the state to split. In addition, the generated set of states must satisfy the self-consistency criterion, which only permits a unique transition to emanate from a state for a given symbol. If $\delta(q, \sigma)$ is not unique for each $\sigma \in \Sigma$, then the state q is split further. In the state splitting algorithm, a stopping rule is constructed by specifying the threshold parameter η_{spl} on the rate of decrease of conditional entropy. An alternative stopping rule for the algorithm is to provide a maximal number of states N_{max} instead of the threshold parameter η_{spl} . The operation of state splitting is described in Algorithm 1 (see Appendix C).

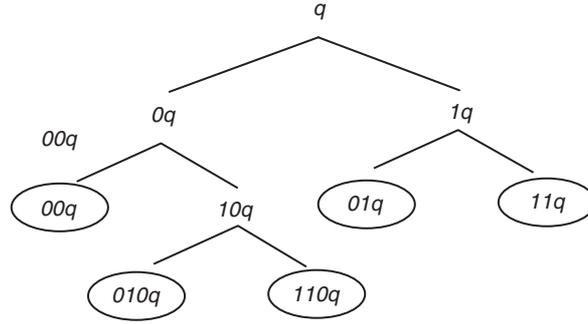


Figure 2: Tree-representation of state splitting in D-Markov machines.

Let q be a D -Markov state (see Definition 3.1), which is split to yield new states σq , where $\sigma \in \Sigma$ and σq represents the equivalence class of all (finite-length) symbol strings with the word σq as the suffix. Figure 2 illustrates the process of state splitting in a PFSA with alphabet $\Sigma = \{0, 1\}$, where each terminal state is circumscribed by an ellipse. For example, the states in the third layer from the top are: $00q$, $10q$, $01q$, and $11q$, of which all but $10q$ are terminal states. Consequently, the state $10q$ is further split as $010q$ and $110q$ that are also terminal states, i.e., $Q = \{00q, 01q, 11q, 010q, 110q\}$, as seen in the split PFSA diagram of Figure 2. Given the alphabet Σ and the associated set Q of states, the morph matrix Π can be computed in the following way.

$$\pi(q, \sigma) = P(\sigma | q) = \frac{P(q\sigma)}{P(q)} \quad \forall \sigma \in \Sigma \quad \forall q \in Q \quad (3)$$

where $P(\cdot)$ and $P(\cdot | \cdot)$ are the same as those used in Definition B.7 and Eq. (14) therein.

For construction of PFSA, each element $\pi(\sigma, q)$ of the morph matrix Π is estimated by frequency counting as the ratio of the number of times, $N(q\sigma)$, the state q is followed (i.e., suffixed) by the symbol σ and the number of times, $N(q)$, the state q occurs. By using the structure of Eq. (2), it follows from Eq. (3) that each element $\hat{\pi}(\sigma, q)$ of the estimated morph matrix $\hat{\Pi}$ is obtained as

$$\hat{\pi}(q, \sigma) \triangleq \frac{1 + N(q\sigma)}{|\Sigma| + N(q)} \quad \forall \sigma \in \Sigma \quad \forall q \in Q \quad (4)$$

where $\sum_{\sigma \in \Sigma} \hat{\pi}(\sigma, q) = 1 \quad \forall q \in Q$.

Similar to Eq. (4) and following the structures of Eq. (2), each element $P(q)$ of the stationary state probability vector is estimated by frequency counting as

$$\hat{P}(q) \triangleq \frac{1 + N(q)}{|Q| + \sum_{q' \in Q} N(q')} \quad \forall q \in Q \quad (5)$$

where $\hat{P}(q)$ is an element of the estimated stationary state probability vector, which implies the estimated stationary probability of the PFSA being in the state $q \in Q$. Wen *et al.* [22] have statistically modeled the error of estimating the state probability vector from finite-length symbol strings.

Now the entropy rate (see Eq. (14) in Appendix B) is computed in terms of the elements of estimated state probability vector and estimated morph matrix as

$$\begin{aligned} H(\Sigma | Q) &= - \sum_{q \in Q} \sum_{\sigma \in \Sigma} P(q) P(\sigma | q) \log P(\sigma | q) \\ &\approx - \sum_{q \in Q} \sum_{\sigma \in \Sigma} \hat{P}(q) \hat{\pi}(q, \sigma) \log \hat{\pi}(q, \sigma) \end{aligned} \quad (6)$$

2) *The State Merging Algorithm*: Once state splitting is performed, the resulting D -Markov machine is a statistical representation of the symbol string under consideration. Depending on the choice of alphabet size $|\Sigma|$ and depth D , the number of states after splitting may run into hundreds. Although increasing the number of states of the machine may lead to a better representation of the symbol string, it rapidly increases the execution time and memory requirements. The motivation behind the state merging is to reduce the number of states, while preserving the D -Markov structure of the PFSA. Of course, such a process may cause the PFSA to have degraded precision due to loss of information. The state merging algorithm aims to mitigate this risk.

In the state merging algorithm, a stopping rule is constructed by specifying an acceptable threshold η_{mrg} on the distance $\Phi(\cdot, \cdot)$ between the merged PFSA and the PFSA generated from the original time series. Before embarking on the state merging algorithm, the procedure for merging of two states is described below.

Notion of merging two states: The process of state merging is addressed by creating an equivalence relation [40], denoted as \sim , between the states. The equivalence relation specifies which states are identified to belong to the same class, thereby partitioning the original set of states into a smaller number of equivalence classes of states, each being a nonempty collection of the original states. The new states are, in fact, equivalence classes as defined by \sim .

Let $K_1 = \{\Sigma, Q_1, \delta_1, \pi_1\}$ be the *split* PFSA, and let $q, q' \in Q_1$ be two states that are to be *merged* together. Initially, an equivalence relation is constructed, where none of the states are equivalent to any other state except itself, i.e., each equivalence class is represented by a singleton set. To proceed with the merging of states q and q' , an equivalence relation is imposed between q and q' , denoted as $q \sim q'$; however, the transitions between original states may not be well-defined anymore, in the following sense: there may exist $\sigma \in \Sigma$ such that the states $\delta_1(q, \sigma)$ and $\delta_2(q', \sigma)$ are not equivalent. In essence, the same symbol may cause a transition to two different states from the merged state $\{q, q'\}$. As the structure of D -Markov machines does not permit this ambiguity of non-determinism [13], the states $\delta_1(q, \sigma)$ and $\delta_2(q', \sigma)$ are also *required* to be merged together, i.e., $\delta_1(q, \sigma) \sim \delta_2(q', \sigma)$ (this procedure is known as *determinization* in the state merging literature). Therefore, the symbol σ will cause a transition from the merged state $\{q, q'\}$ to the merged state $\{\delta_1(q, \sigma), \delta_2(q', \sigma)\}$. This process is recursive

and is performed until no ambiguity in state transitions occurs. Indeed at each iteration, the number of states of the future machine is reduced, and the machine where all the states are merged is always consistent. Therefore, the number of states is a decreasing sequence of positive integers, which must eventually converge. The recursive operation of the equivalence relation \sim is described in Algorithm 2 (see Appendix C).

Let $K_1 = \{\Sigma, Q_1, \delta_1, \pi_1\}$ be the *split PFSA* that is merged to yield the reduced-order PFSA $K_2 = \{\Sigma, Q_2, \delta_2, \pi_2\}$, where the state-transition map δ_2 and the morph function π_2 for the merged PFSA K_2 are defined on the quotient set $Q_2 \triangleq Q_1/\sim$, and $[q] \in Q_2$ is the equivalence class of $q \in Q_1$. Then, the associated morph function π_2 is obtained as:

$$\begin{aligned}
\pi_2([q], \sigma) &= P\left[s_{i+1} = \sigma \mid \bigcup_{\tilde{q} \in [q]} \{X_i = \tilde{q}\}\right] \\
&= \frac{\sum_{\tilde{q} \in [q]} P[s_{i+1} = \sigma; \{X_i = \tilde{q}\}]}{\sum_{\tilde{q} \in [q]} P(X_i = \tilde{q})} \\
&= \frac{\sum_{\tilde{q} \in [q]} P[s_{i+1} = \sigma \mid X_i = \tilde{q}]P(X_i = \tilde{q})}{\sum_{\tilde{q} \in [q]} P(X_i = \tilde{q})} \\
&\approx \frac{\sum_{\tilde{q} \in [q]} \hat{\pi}_1(q, \sigma) \times \hat{P}_1(\tilde{q})}{\sum_{\tilde{q} \in [q]} \hat{P}_1(\tilde{q})} \tag{7}
\end{aligned}$$

As seen in Eq. (7), the morph function π_2 of the merged PFSA K_2 is estimated as the sum of $\hat{\pi}_1$ weighted by the stationary state probabilities \hat{P}_1 of the PFSA K_1 . By construction, δ_2 is naturally obtained as

$$\delta_2([q], \sigma) = [\delta_1(q, \sigma)] \tag{8}$$

Algorithm 3 in Appendix C presents the procedure to obtain the PFSA, where the objective is to merge the states q and q' .

Identification of the states to be merged: The next task is to decide which states have to be merged. States that behave similarly (i.e., have similar morph probabilities) have a higher priority for merging. The similarity of two states, $q, q' \in Q$, is measured in terms of morph functions (i.e., conditional probabilities) of future symbol generation as the distance between the two rows of the estimated morph matrix $\hat{\Pi}$ corresponding to the states q and q' . The ℓ_1 -norm (i.e., the sum of absolute values of the vector components) has been adopted to be the distance function as seen below.

$$\begin{aligned}
\mathcal{M}(q, q') &\triangleq \|\hat{\pi}(q, \cdot) - \hat{\pi}(q', \cdot)\|_{\ell_1} \\
&= \sum_{\sigma \in \Sigma} |\hat{\pi}(q, \sigma) - \hat{\pi}(q', \sigma)| \tag{9}
\end{aligned}$$

A small value of $\mathcal{M}(q, q')$ indicates that the two states have close probabilities of generating each symbol. Note that this measure is bounded above as $\mathcal{M}(q, q') \leq 2 \forall q, q' \in Q$, because $0 \leq \sum_{\sigma \in \Sigma} \hat{\pi}(q, \sigma) \leq 1$ and $0 \leq \sum_{\sigma \in \Sigma} \hat{\pi}(q', \sigma) \leq 1$. Now the procedure of state merging is briefly described below.

First, the two closest states (i.e., the pair of states $q, q' \in Q$ having the smallest value of $\mathcal{M}(q, q')$) are merged using Algorithm 3 (see Appendix C). Subsequently, distance $\Phi(\cdot, \cdot)$ (see Eq. (15) in Subsection B-C) of the merged PFSA from the initial symbol string is evaluated. If $\Phi < \eta_{mrg}$ where η_{mrg} is a specified threshold, then the machine structure is retained and the states next on the priority list are merged. On the other hand, if $\Phi \geq \eta_{mrg}$, then the process of merging the given pair of states is aborted and another pair of states with the next smallest value of $\mathcal{M}(q, q')$ is selected for merging. This procedure is terminated if no such pair of states exist, for which $\Phi < \eta_{mrg}$. The operation of the procedure is described in Algorithm 4 (see Appendix C).

3) *Feature Extraction*: After the D -Markov machine is constructed, the stationary state probability vector is computed in the following way. Similar to Eq. (4), each element of the stationary state probability vector $P(q)$ is estimated by frequency counting as $\hat{P}(q)$ (see Eq. (5)).

The estimated stationary state probability vector $\hat{P}(q)$ serves as a feature vector representing the associated time series for classification of different zones of flame; this feature vector is low-dimensional and can be computed in real time, based on the varying equivalence ratio before the onset of LBO. For example, if the alphabet is $\{1, 2, 3, 4, 5\}$, the set of states after state splitting is $\{1, 2, 13, 23, 33, 43, 53, 14, 24, 34, 44, 54, 5\}$ for port 3 and state merging leads to the set of states, $\{1, 2, 13, 23, 33, 43, 53, \{14, 24, 34\}, 44, 54, 5\}$. The stationary state probability vector computed from the D -Markov machine with proposed state description, served as the feature for this prediction problem. Figure 3 shows the state probability vectors from chemiluminescence time series at three different stages of fuel-air ratios.

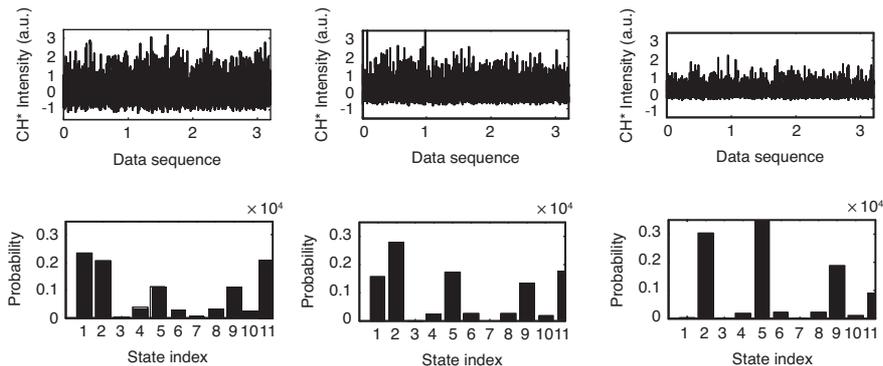


Figure 3: Markov machine features from chemiluminescence time-series for three different stages of fuel-air ratios (from left to right: $\phi = 0.90$, $\phi = 0.74$ and $\phi = 0.66$) for airflow at 150 lpm for $L_{fuel} = 250$ mm (Port 3).

4) *Kullback-Leibler Divergence as LBO measure*: Let \hat{P}^0 be the estimated state probability vector (see Eq. (5)) of the resulting probabilistic finite state automaton (PFSA) model at the reference epoch τ_0 in the slow time scale (see Appendix A) and let \hat{P}^k be the estimated state probability vector of the (possibly evolved) PFSA model at an epoch τ_k . Relative to the reference flame condition at the epoch τ_0 , any anomalous behavior of the current flame condition at the epoch τ_k is expressed in terms of \hat{P}^0 and \hat{P}^k as the (scalar) Kullback-Leibler divergence [21] that is defined as

$$d(\hat{P}^k || \hat{P}^0) \triangleq \sum_{q \in Q} \hat{P}^k(q) \log \left(\frac{\hat{P}^k(q)}{\hat{P}^0(q)} \right) \quad (10)$$

where other choices of the divergence (e.g., standard Euclidean distance) can also be made (for example, see [13]). A special advantage of using the Kullback-Leibler divergence is that it provides an average anomaly measure of the current the flame relative to the reference condition in the log scale. It is noted that, in Eq. (10), the standard conventions $0 \log(0) = 0$ and $p \log \left(\frac{p}{0} \right) = \infty \forall p \in (0, \infty)$ are applicable based on

the continuity arguments. However, such a singularity condition is highly undesirable for continuous monitoring and control of flame stability and it is avoided by appropriate selection of the reference vector \hat{P}^0 , each of whose elements is strictly positive.

5) *Pattern Classification*: The next step is classification of patterns among the features extracted from the time series data. A nested classification architecture, as shown in Figure 4, is proposed based on the range of the non-dimensional ratio of ϕ/ϕ_{LBO} to predict a

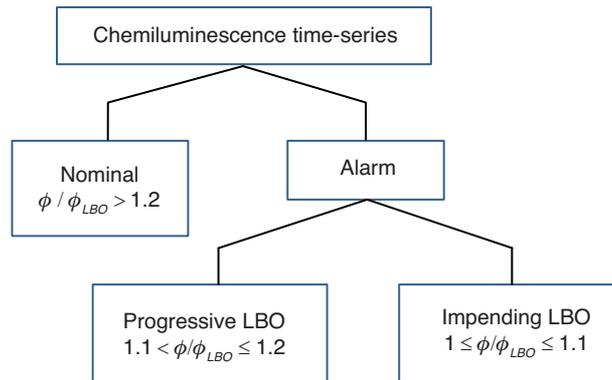


Figure 4: Nested classification for lean blowout (LBO) prediction.

forthcoming LBO, irrespective of the airflow rates for a certain premixing level. Initially, the chemiluminescence time series of duration 16 *sec* for different premixing lengths were grouped into two classes as: *Alarm* ($1 \leq \phi/\phi_{LBO} \leq 1.20$) and *Nominal* ($\phi/\phi_{LBO} > 1.20$). The class *Alarm* was divided into two finer classes as: *Impending LBO* (ILBO) for $1 \leq \phi/\phi_{LBO} \leq 1.1$, and *Progressive LBO* (PLBO) for $1.1 < \phi/\phi_{LBO} \leq 1.2$. Identification of the PLBO phase is crucial for LBO mitigation as the control actions need to be initiated typically near the PLBO-ILBO boundary.

While there are many tools for pattern classification [34], this paper makes use of two well known techniques, namely, *leave-one-out* and *support vector machines (SVM)*. The leave-one-out method is adopted, because of the limited availability of training and test data; it makes use of the cross validation concept, where n groups of $(n - 1)$ training data are used with the remaining one of the n available measurements being treated as test data. In Section 4, the SVM method with either a Gaussian or a linear kernel [34] has been used to determine the classes to which the test features belong.

4. RESULTS AND DISCUSSIONS

This section validates the algorithms of *D*-Markov machines for LBO prediction on the ensemble of time series data that were generated from the swirl-stabilized dump combustor described in Section 2. Multiple experiments have been conducted with liquefied petroleum gas (LPG) fuel at airflow rates of 150, 175 and 200 lpm for three different fuel-air premixing lengths (i.e., distance of fuel injection port from the dump plane) of $L_{fuel} = 350$ mm, 250 mm, and 150 mm for Port 1, Port 3, and Port 5, respectively, where Reynolds numbers based on cold flow conditions have been up to 18, 700 at 200 lpm.

A. Experimental Observations

The combustion process was relatively steady and occupied the whole combustor while the shape of the flame was conical as the fuel was injected through Port 1 (with $L_{fuel} = 350$ mm) at stoichiometric air-fuel mixture (i.e., $\phi = 1$). As the equivalence ratio was reduced to $\phi = 0.81$ case, there was a significant change in the flame color; it became bluish with a reddish tip while retaining a well defined combustion region. For conditions close to LBO (e.g., $\phi \sim 0.75$) the flame shape changed from conical to elongated columnar due to reduced reaction rate and burning velocity of flame near LBO. There were random instances of flame oscillations and flame lift-off from the dump plane. As the unburned fuel was reignited, the flame became detached from the center body and returned to the inlet. The random occurrences of unique extinction and re-ignition events spanned a period of several milliseconds prior to LBO.

The observations with the minimum premixing length $L_{fuel} = 150$ mm were significantly different from those of larger premixing lengths for operations near the LBO limit. The flame was attached to the dump plane and did not show any lift-off pattern at all times due to lower fuel-air premixing. The flame intensity was significantly reduced and, after a subsequent reignition, the flame did not oscillate and the precursor events were not so intense. Furthermore, the flame was not symmetrically attached to the dump plane and exhibited asymmetric spread with a flickering nature.

B. Reduction of Modeling Complexity near Lean Blowout

Each chemiluminescence time-series is converted to zero mean by subtracting the bias which is a dominating factor in generating the anomaly measure in the symbolic analysis, reported by Mukhopadhyay *et al.* [12]. The removal of the bias has yielded a significant improvement in the performance of LBO prediction over the previous strategy [12], because mean-based detection may lead to the controller operating the engine at relatively richer conditions with consequent penalties of enhanced NOx emission. Since there is no bias in the algorithms reported in this paper, the texture of time series data is modeled precisely to predict LBO ahead of time. The reference time series (i.e., at $\phi = 1$) for different premixing lengths are partitioned for an alphabet size of seven (i.e., $|\Sigma| = 7$) via maximum entropy partitioning (MEP) [14], [31]. This information on partitioning has been used to symbolize the data set at different equivalence ratios (i.e., for $\phi < 1$) for the corresponding premixing length. Once the time series data are symbolized, D -Markov machines are constructed from the symbol strings via state splitting and state merging.

Figure 5 shows that the number of states of the D -Markov machine after state splitting and state merging reduces approximately monotonically as the equivalence ratio is dropped for $L_{fuel} = 350$ mm. In this case, state splitting is done with an upper bound of 30 PFSA states (i.e., $N_{max} = 30$ in Algorithm 1 of Appendix C) and state merging with a threshold parameter $\eta_{mrg} = 0.05$ (see Algorithm 4 of Appendix C). In Figure 5, the right-hand endpoints of the curves after $\phi = 0.63$ denote the onset of LBO. Hence, it can be inferred that modeling complexity (i.e., the number of states) for D -Markov machine construction reduces drastically as the flame approaches LBO. For other premixing lengths (e.g., $L_{fuel} = 250$ mm and $L_{fuel} = 150$ mm), the modeling complexity near LBO is also reduced by a large amount compared to that at $\phi = 1$.

It appears that the complexity of PFSA models (e.g., the number of states $|Q|$) tends to reduce as an LBO situation is approached. The dynamics of a combustor close to blowout is a topic of intense current research. For example, Gotoda and coworkers [17][18]

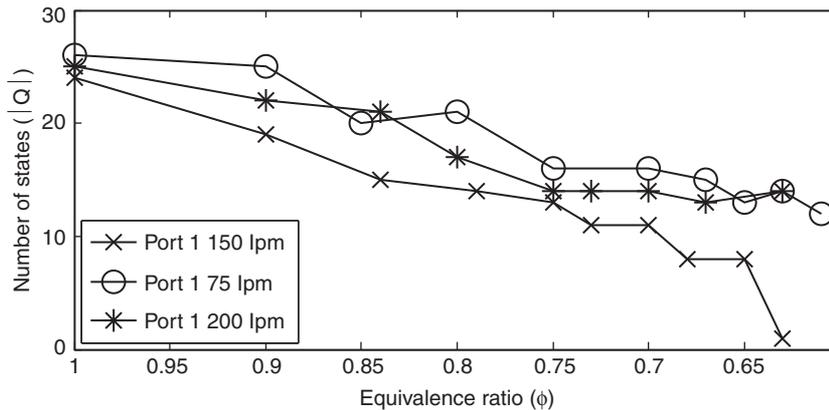


Figure 5: State order reduction for $L_{fuel} = 350$ mm (Port 1).

have reported increase in system complexity as LBO is approached; on the other hand, Kabiraj and Sujith [16] have shown a slight decrease in embedding dimension during intermittency prior to blowout. A possible explanation is that the effects of both destructive and constructive interferences in the combustion process give rise to a large dimension of the phase-space that is represented by a relatively large $|Q|$ in the PFSA model. Similarly, it is observed that the embedding dimension of the combustion system reduces in the vicinity of an LBO. However, since these conjectures are at best qualitative, further theoretical and experimental research is necessary to quantitatively assess their truth or falsity. Therefore, in the context of the current paper, further research is needed to correlate the reduction in $|Q|$ with changes in the dynamic characteristics of the combustion process. These issues are identified as a topic of future research in Section 5.

Although modeling complexity approximately monotonically reduces as the flame approaches LBO, it cannot be treated as a definitive measure for LBO prediction, because the number of states may fluctuate within ranges of equivalence ratios at small premixing length. Hence, an anomaly measure (see Eq. (10) in Subsection 3-B4) is constructed to quantify the proximity of the combustion system to LBO. Furthermore, while approaching LBO, different ranges of equivalence ratio ϕ can be identified, irrespective of the airflow rate at a fixed premixing level.

C. Prediction of Lean Blowout (LBO)

This subsection presents the performance of the anomaly measure to quantify the proximity to LBO under different premixing conditions and also elucidates the classification performance for predicting different combustion regimes before the onset of LBO. Following Figure 5, where the number of states is within a reasonable (i.e., after excluding a few obvious outliers) range of 6 to 13, the maximum number of states was assigned to be $N_{max} = 13$ for state splitting in the D -Markov machine construction. The LBO measure is computed with respect to a reference at a condition far from LBO and it is kept at zero. To compare LBO measures for different airflow rates at a certain premixing level, they are normalized with respect to the LBO measures at their respective blowout point.

Figures 6(a), 6(b), and 6(c) show the performance of the proposed LBO measure for three different flow rates (i.e., 150, 175 and 200 lpm) at well-premixed (Port 1), partially-premixed (Port 3), and poorly-mixed (Port 5) fuel-air conditions, respectively. It is seen that, with the exception of the poorly-mixed condition (i.e., $L_{fuel} = 150$ mm (Port 5)), the average slope of the normalized LBO measure with respect to normalized equivalence ratio (ϕ/ϕ_{LBO}) remains low till ϕ/ϕ_{LBO} reaches ~ 1.2 . As ϕ/ϕ_{LBO} is reduced below 1.2, the slope of the normalized LBO measure starts increasing rapidly even for the well-premixed condition, as seen in Figure 6(a). Below $\phi/\phi_{LBO} = 1.1$, the normalized LBO measure attains a value in the range of 0.4 – 0.6 and it reaches 1 with a steep slope when the flame blows out. The steep rise in slope of the normalized LBO measure close to LBO ($\phi/\phi_{LBO} < 1.2$) is desirable, because a sensitive LBO measure would be capable of detecting the proximity of LBO. It is apparent from Figure 6(c) that the normalized LBO measure for the poorly-mixed is not as sensitive as it is in the cases

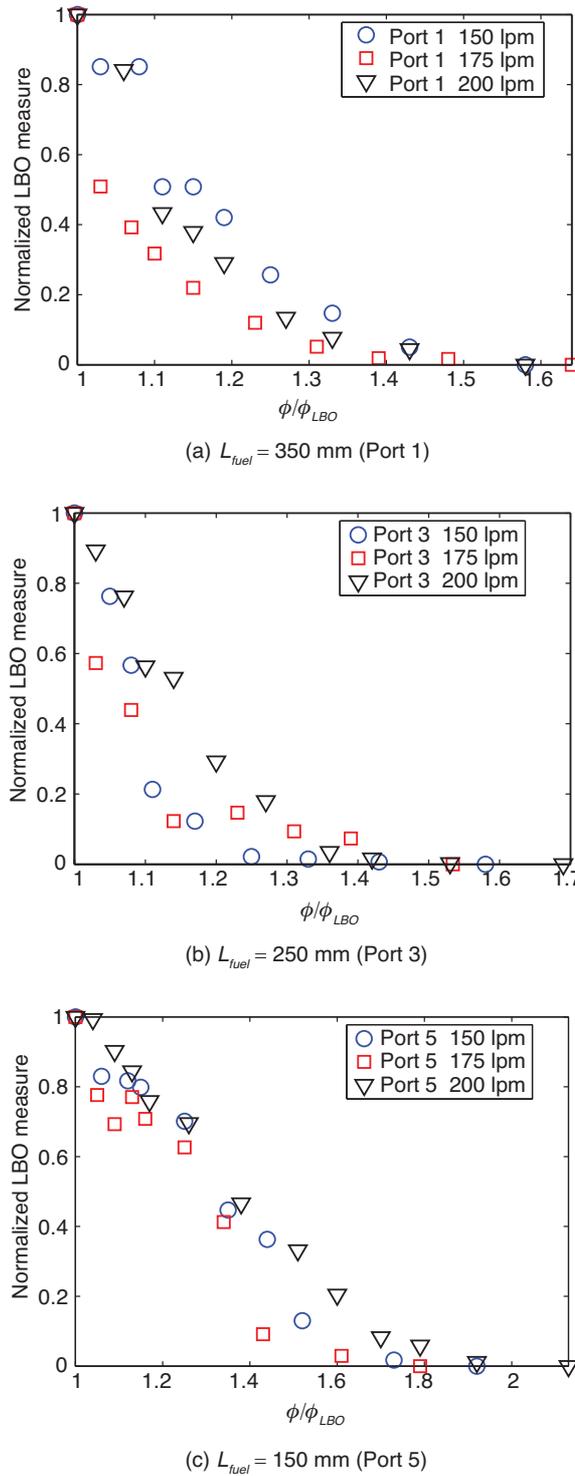


Figure 6: Dependence of Normalized LBO measure on ϕ/ϕ_{LBO} for three airflow rates of 150, 175 and 200 lpm.

Table 1: Confusion matrix before lean blow out for differet premixing (port 1, port 3 and port 5)

		PREDICTED								
A	Premixing level	Port 1 ($L_{fuel} = 350$ mm)			Port 3 ($L_{fuel} = 250$ mm)			Port 5 ($L_{fuel} = 150$ mm)		
C		Alarm			Alarm			Alarm		
T	Class	ILBO	PLBO	Nom	ILBO	PLBO	Nom	ILBO	PLBO	Nom
U	ILBO	9	0	0	9	0	0	6	2	0
A	Alarm PLBO	0	7	0	0	6	0	0	6	0
L	Nom	0	1	12	0	1	12	0	3	16

of well-premixed and partially-premixed conditions. The rationale for this result can be attributed to the absence of dominant precursor events prior to LBO at a poorly-premixed condition.

Subsequently, the ensemble of time series data for these three air-flow rates are mixed to build a robust classification scheme, where it is ensured that such data sets were independently collected. Since the number of samples in each class is not large (e.g., ~ 10), leave-one-out cross-validation approach is adopted for LBO prediction and support vector machines (SVM) with linear kernels [34] are used at both first and second levels of classification (see Figure 4). The results of this classification method are presented in Table I using confusion matrices for three different premixing lengths, where the rows are the actual classes (i.e., ground truth) and the columns are the predicted classes. [Note: the notions of two classes of alarm, namely, *Impending* LBO (ILBO) and *Progressive* LBO (PLBO) have been introduced in Subsection 3-B5]. Table I shows high accuracy in detecting ILBO, PLBO, and nominal conditions with larger air-fuel premixing (i.e., higher values of L_{fuel}).

The confusion matrices in Table I show that the *D*-Markov machine is capable of classifying different LBO situations fairly accurately even in the absence of intense visual flame-precursor events. The efficacy of the proposed method to detect LBO even at low levels of premixing is important as other methods reported in literature mostly deal with lean premixed flames and do not generally work very satisfactorily for partially premixed configurations [12]. This classification scheme, in general, predicts the proximity of LBO fairly accurately for a wide range of air-flow rate,

The *D*-Markov machine parameters of the pattern classifier are now presented for LBO prediction for well-premixed, partially-premixed, and poorly premixed fuel-air flow.

1) *LBO Prediction for Premixed Flame*: For combustion flames with fuel flow from Port 1 (i.e., $L_{fuel} = 350$ mm) and Port 3 (i.e., $L_{fuel} = 250$ mm) are at well-premixed and partially-premixed conditions, respectively. The *D*-Markov machine parameters are found to be the same for these cases.

The symbol alphabet is $\Sigma = \{1, 2, 3, 4, 5, 6, 7\}$, i.e., the alphabet size is $|\Sigma| = 7$. After state merging, the states of the *D*-Markov machine are calculated for the

stoichiometric fuel/air ratio, i.e., $\phi = 1$, are found to be eleven (i.e., $|Q| = 11$) for the data from Port 1 and Port 3. The corresponding sets of states are: $\{1, 2, 3, \{14, 24\}, 34, 44, 54, \{64, 74\}, 5, 6, 7\}$ and $\{1, 2, 3, 4, \{15, 75\}, \{25, 35\}, 45, 55, 65, 6, 7\}$ for port 1 and port 3, respectively.

2) *LBO Prediction for Non-Premixed Flame*: Combustion flame with fuel flow from port 5 ($L_{fuel} = 150$ mm) is close to non-premixed condition. The alphabet size is kept at $|\Sigma| = 7$. After state merging, the number of states for the D -Markov machine is calculated (based on $\phi = 1$) to be eleven. Figure 6(c) shows the performance of the proposed LBO measure for different flow rates at a (nearly) non-premixed condition.

D. Performance of D -Markov Machine: $D = 1$ and $D > 1$

D -Markov machines with $D = 1$ was reported by Chaudhuri [23] and Mukhopadhyay et al. [12] to perform equivalently or sometime superior to other time series based online LBO prediction tools [6], [7], [8], [9], [10]. This subsection makes a comparison of the predictive performance of D -Markov machines for $D > 1$ with that for $D = 1$, which was reported earlier for LBO prediction [12]. This comparison is performed based on the following two metrics.

- 1) Total mis-prediction = (Actual ILBO, predicted PLBO) + (Actual ILBO, predicted Nominal) + (Actual PLBO, predicted Nominal); and
- 2) Total false alarm = (Actual Nominal, predicted PLBO) + (Actual Nominal, predicted ILBO) + (Actual PLBO, predicted ILBO).

Figure 7 shows how the classification performance is improved for a typical case of combustion with fuel inlet at port 5 (see Figure 1), as the number of PFSA states $|Q|$ in the D -Markov machine is increased via state splitting and state merging for an alphabet

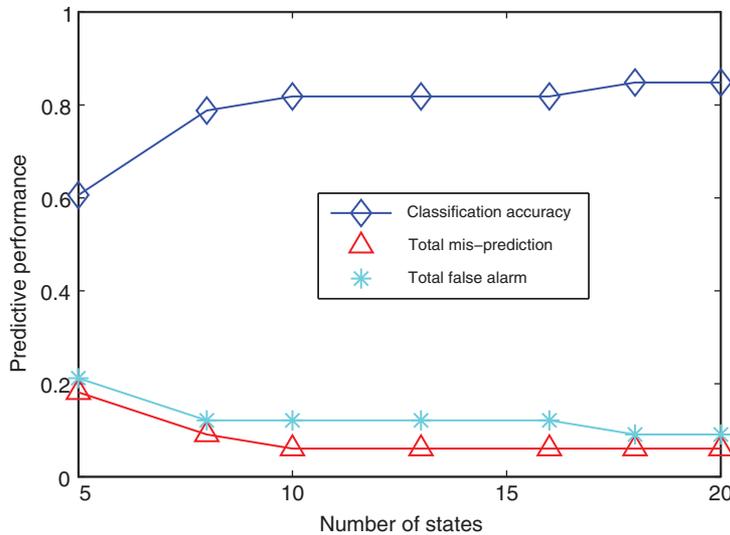


Figure 7: Dependence of predictive performance of the D -Markov machine on the number of states ($|Q|$) from alphabet size $|\Sigma| = 5$.

Table 2: Performance comparison of LBO prediction by D -Markov machine classification with $D > 1$ and $D = 1$

Port # (# of test cases)		Total mis- predictions	Total false alarms
Port 1 (29)	$D > 1$	0	1
	$D = 1$	2	3
Port 3 (28)	$D > 1$	0	1
	$D = 1$	4	4
Port 5 (33)	$D > 1$	2	3
	$D = 1$	6	9

size $|\Sigma| = 5$. It appears that the predictive performance of the D -Markov machine saturates beyond a certain value of number of states (e.g., $|Q| \approx 10$), as seen in Figure 7. Investigation of the fact, whether the phase space dimension of the combustion dynamics consistently converges in the vicinity of LBO, is a topic of future research as indicated in Section 5.

Table II shows that the classification performance of the D -Markov machine with $D > 1$ becomes increasing better relative to that of its predecessor with $D = 1$ as the quality of premixing is degraded. For higher premixing, the performances of Port 1 and Port 3 are comparable, which is intuitively supported by the presence of intense and clearly visible precursor events before the onset of LBO. However, as the quality of fuel-air premixing is reduced (e.g., Port 5), the visibility of precursor events becomes rather rare. This physical phenomenon makes the task of LBO prediction more difficult. The last row of Table II shows that the the D -Markov machine with $D > 1$ performs well for all ports including Port 5 too, whereas the D -Markov machine with $D = 1$ fails to predict LBO as it yields unacceptable levels of total mis-predictions and total false alarms. The technique adopted in [12] (that uses $D = 1$) may not perform satisfactorily when the bias due to the mean value of the CH^* chemiluminescence is removed from the analyzed signal. Although the usage of $D = 1$ may work in presence of the bias, mean-based implementations could lead to higher emission as discussed earlier. Thus, the development and validation of the D -Markov machine with $D > 1$ for LBO prediction is one of the major contributions of the present work.

E. Estimate of the Computational Cost

The computational cost of the proposed algorithm is estimated for calculating the LBO measure from the time series of raw chemiluminescence data on a Dell Precision T3400 platform with Intel(R) Core(TM) 2 Quad CPU Q9550 @ 2.83 GHz. The state space of the PFSA is usually constructed off-line via state splitting and state merging. Figure 8 shows the profiles of required computational time (in the MATLAB 7.10.0 (R2010a) environment) to obtain the LBO measure from a one-second-duration time series of raw chemiluminescence data, collected at a sample frequency of 2 kHz. The

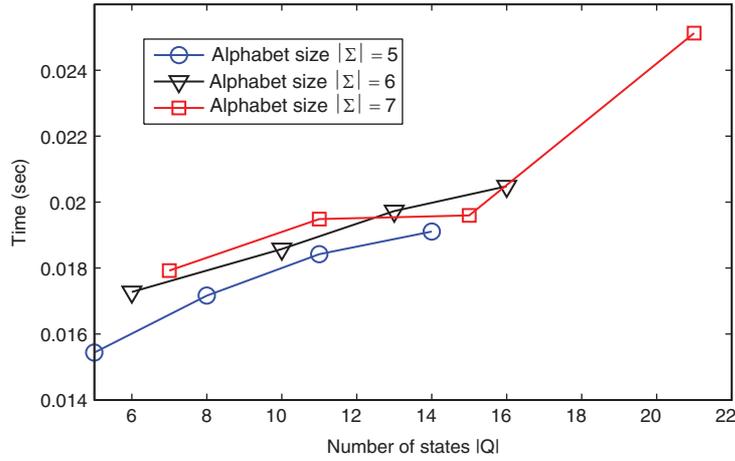


Figure 8: Computation times for calculating LBO measure from 1 second of raw chemiluminescence time series (sampling frequency 2 kHz) for different number of states $|Q|$ and different alphabet size $|\Sigma|$.

equality of alphabet size $|\Sigma|$ and number of PFSA states $|Q| >$ at the beginning of each plot in Figure 8 implies $D = 1$; for subsequent points in each plot, $|Q| > |\Sigma|$, which implies that state splitting and merging with $D > 1$. It is seen that the computational time has an increasing trend as the depth D of the underlying PFSA (see Definition 3.1) is increased from one. A typical value of computational time is 20 ms for the alphabet size $|\Sigma| = 7$ and the number of PFSA states $|Q| = 11$. Apparently, the proposed method of LBO prediction is well-suited for real-time LBO prediction, where the typical frequency response of combustion dynamics is ~ 10 Hz [18].

5. SUMMARY, CONCLUSIONS AND FUTURE WORK

This paper addresses data-driven pattern classification for prediction of lean blowout (LBO) phenomena in combustion processes. The proposed LBO prediction method is built upon low-dimensional feature vectors that are extracted from time series of optical sensor data of chemiluminescence. The feature vectors are realized as (statistically stationary) state probability vectors of a special class of finite-history probabilistic finite state automata (PFSA). These PFSA, called D -Markov machines, have a deterministic algebraic structure and their states are represented by symbol blocks of length D or less, where D is a positive integer. The states of a D -Markov machine are constructed via splitting the symbol blocks of different lengths based on their information contents and merging two or more split states without any significant loss of the embedded information.

An anomaly measure, based on Kullback-Leibler divergence, is constructed to successfully predict the onset of LBO. This anomaly measure becomes increasingly sensitive to small changes in the equivalence ratio as the combustion process

approaches LBO. An architecture of the pattern classification problem has been formulated based on different ranges of the equivalence ratio to reliably predict the LBO optimally ahead of its onset. The proposed pattern classification method has been validated on experimental data collected from a laboratory-scale swirl-stabilized combustor. It is observed that the modeling complexity (i.e., number of states of D -Markov machine) of the PFSA, constructed from optical sensor data reduces drastically as the system approaches LBO.

It is demonstrated over a wide range of fuel-air premixing and air flow rates that the D -Markov machine with $D > 1$ performs significantly better than the D -Markov machine with $D = 1$ for prediction of LBO [12]. The results, reported in this paper, suggest the potential capability of D -Markov machines to fairly precisely predict regions close to LBO in a laboratory-scale combustor. In this way, the operating condition in a combustor could be extended to a leaner equivalence ratio without significantly risking LBO. While there are many other areas yet to be addressed in this context, a few topics of future research are delineated below.

- 1) Theoretical research on identification of the optimum threshold for maximum number of states (state splitting) and the distance metric (state merging) for enhancement of the predictive performance of the D -Markov machine.
- 2) Prediction of combustion instability in both premixed and non-premixed combustors.
- 3) Prediction of LBO under thermo-acoustic instability for different flow conditions.
- 4) Validation of the proposed data-driven approach with respect to model-driven tools.
- 5) Identification of the statistical ranges of uncertainty in the LBO measure (e.g., relative to different confidence levels) in typical combustors.
- 6) Investigation of the physical significance of the reduction in number of D -Markov machine states in the vicinity of LBO relative to the corresponding changes in the dynamic behavior of the combustion system under nominal operating conditions.
- 7) Theoretical and experimental research on prediction of thermo-acoustic instabilities under operating conditions, other than LBO (e.g., screech phenomena in the afterburner of gas turbine engines in tactical aircraft [41]).

ACKNOWLEDGEMENTS

The work reported in this paper has been supported in part by the U.S. Air Force Office of Scientific Research under Grant No. FA9550-12-1-0270, by the U.S. Office of Naval Research under Grant No. N00014-14-1-0545, and by the U.S. Army Research Laboratory under Grant No. W911NF-14-2-0068. 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. The authors thankfully acknowledge the benefits of scientific and technical discussions with their colleagues in the following areas:

- Dr. K. Mukherjee and Dr. S. Phoha of Pennsylvania State University for development of the theory of the D -Markov machine.
- Professor S. Chakravarthy and Professor R.I. Sujith of Indian Institute of Technology, Madras for interpretation of the physical phenomena of combustion.

APPENDIX A

CONCEPT OF TWO TIME SCALES

This appendix presents the notion of two time scales in dynamical systems, as needed for pattern classification based on time series of sensor data.

Definition A.1 (Fast Scale) *The fast scale is defined to be a time scale over which the statistical properties of the process dynamics are assumed to remain invariant, i.e., the process is assumed to have statistically stationary dynamics at the fast scale.*

Definition A.2 (Slow Scale) *The slow scale is defined to be a time scale over which the statistical properties of the process dynamics may gradually evolve, i.e., the process may exhibit statistically non-stationary dynamics at the slow scale.*

In view of Definition A.1, statistical variations in the internal dynamics of the process are assumed to be negligible at the fast scale. Thus, sensor time series data are acquired based on the assumption of statistical stationarity at the fast scale. In view of Definition A.2, an observable non-stationary behavior could be associated with the gradual evolution of anomalies (i.e., deviations from the nominal behavior) in the process at the slow scale. In general, a long time span at the fast scale is a tiny (i.e., several orders of magnitude smaller) interval at the slow scale. A pictorial view of the two-time-scales operation in Figure 9 illustrates the concept.

APPENDIX B

MATHEMATICAL PRELIMINARIES

This appendix presents pertinent information regarding construction of D -Markov machines and other mathematical tools (e.g., entropy rate and metric to quantify the distance between two PFSA). The following standard definitions are recalled [14].

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

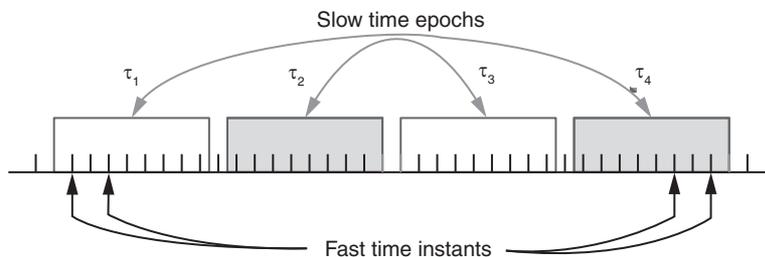


Figure 9: Underlying concept of fast and slow time scales.

- Σ is a (nonempty) finite alphabet with cardinality $|\Sigma|$;
- Q is a (nonempty) finite set of states with cardinality $|Q|$;
- $\delta : Q \times \Sigma \rightarrow Q$ is a state transition map.

Definition B.2 (Symbol Block) 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 $w \triangleq s_1 s_2 \dots s_\ell$ with $s_i \in \Sigma$ 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 , including the empty word ϵ , is denoted as Σ^* ,
- The set of all words, whose suffix (respectively, prefix) is the word w , is denoted as $\Sigma^* w$ (respectively, $w \Sigma^*$).
- The set of all words of (finite) length ℓ , where $\ell > 0$, is denoted as Σ^ℓ .

Definition B.3 (Extended Map) The extended state transition map $\delta^* : Q \times \Sigma^* \rightarrow Q$ transfers one state to another through finitely many transitions such that, for all $q \in Q$, $\sigma \in \Sigma$ and $w \in \Sigma^*$,

$$\delta^*(q, \epsilon) = q \text{ and } \delta^*(q, w\sigma) = \delta(\delta^*(q, w), \sigma)$$

where $w\sigma$ is the suffixing of the word w by the symbol σ .

Definition B.4 (Irreducible FSA) An FSA G is said to be irreducible if, for all $q_1, q_2 \in Q$, there exists a word $w_{1,2} \in \Sigma^*$ such that $q_1 = \delta^*(q_2, w_{1,2})$.

In the process of symbol generation, the space of time series is partitioned into finitely many mutually exclusive and exhaustive cells, each corresponding to a symbol belonging to a (finite) alphabet. As a trajectory of the dynamical system passes through or touches various cells of the partition, the symbol assigned to the cell is inserted in the symbol string. In this way, a time series corresponding to a trajectory is converted into a symbol string. Figure 10 illustrates the concept of constructing finite state automata (FSA) from time series, which provides the algebraic structure of probabilistic finite state automata (PFSA).

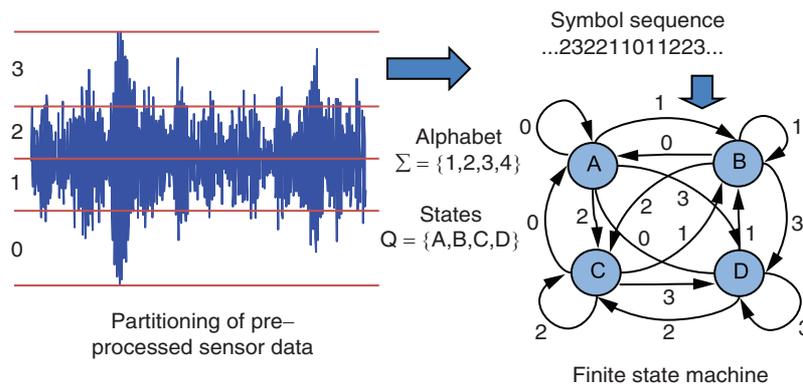


Figure 10: Construction of finite state automata (FSA)

Definition B.5 (PFSA) A probabilistic finite state automaton (PFSA) K is a pair (G, π) , where:

- The deterministic FSA G is called the underlying FSA of the PFSA K ;
- The probability map $\pi : Q \times \Sigma \rightarrow [0, 1]$ 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 $q \in Q$.

Equivalently, a PFSA is a quadruple $K = (G, Q, \delta, \pi)$, where

- The alphabet Σ of symbols is a (nonempty) finite set, i.e., $0 < |\Sigma| < \infty$, where $|\Sigma|$ is the cardinality of Σ ;
- The set Q of automaton states is (nonempty) finite, i.e., $0 < |Q| < \infty$, where $|Q|$ is the cardinality of Q ;
- The state transition function $\delta : Q \times \Sigma \rightarrow Q$;
- The morph function $\pi : Q \times \Sigma \rightarrow [0, 1]$, where $\sum_{\sigma \in \Sigma} \pi(q, \sigma) = 1$ for all $q \in Q$.

The morph function π generates the $(|Q| \times |\Sigma|)$ morph matrix Π .

Definition B.6 (Extended Morph Function) The morph function $\pi : Q \times \Sigma \rightarrow [0, 1]$ of PFSA is extended as $\pi^* : Q \times \Sigma^* \rightarrow [0, 1]$ such that, for all $q \in Q$, $\sigma \in \Sigma$ and $w \in \Sigma^*$,

$$\pi^*(q, \epsilon) = 1 \quad \text{and} \quad \pi^*(q, w\sigma) = \pi^*(q, w) \times \pi(\delta^*(q, w), \sigma)$$

where $w\sigma$ is the suffixing of the word w by the symbol σ . The above equations represent how the PFSA responds to occurrence of a certain block of symbols, i.e., a word $w \in \Sigma^*$ of finite length $|w|$.

A. Symbolization of Time Series

This step requires partitioning (also known as quantization) of the time series data of the measured signal. The signal space is partitioned into a finite number of cells that are labeled as symbols, i.e., the number of cells is identically equal to the cardinality $|\Sigma|$ of the (symbol) alphabet Σ . As an example for the one-dimensional time series in Figure 10, the alphabet $\Sigma = \{\alpha, \beta, \gamma, \delta\}$, i.e., $|\Sigma| = 4$, and three partitioning lines divide the ordinate (i.e., y-axis) of the time series profile into four mutually exclusive and exhaustive regions. These disjoint regions form a partition, where each region is labeled with one symbol from the alphabet Σ . If the value of time series at a given instant is located in a particular cell, then it is coded with the symbol associated with that cell. As such, a symbol from the alphabet Σ is assigned to each (signal) value corresponding to the cell where it belongs. (Details are reported in [31].) Thus, a (finite) array of symbols, called a symbol string (or symbol block), is generated from the (finite-length) time series data.

The ensemble of time series data are partitioned by using a partitioning tool (e.g., maximum entropy partitioning (MEP) or uniform partitioning (UP) methods [31]). In UP, the partitioning lines are separated by equal-sized cells. On the other hand, MEP maximizes the entropy of the generated symbols and therefore, the information-rich

cells of a data set are partitioned finer and those with sparse information are partitioned coarser, i.e., each cell contains (approximately) equal number of data points under MEP. In both UP and MEP, the choice of alphabet size $|\Sigma|$ largely depends on the specific data set and the allowable loss of information (e.g., leading to error of detection and classification).

1) *Selection of Alphabet Size*: Considerations for the choice of alphabet size $|\Sigma|$ include the maximum discrimination capability of a symbol sequence and the associated computational complexity. The maximum discrimination capability is characterized by the entropy of the sequence that should be maximized to the extent it is possible, or alternatively by minimizing the information loss that is denoted as the negative of the entropy. As the alphabet size is increased, there is both an increase in computational complexity and a possible reduction in loss of information; in addition, the effects of a large alphabet may become more pronounced for an insufficiently long time series [22].

In partitioning of a (one-dimensional) time series for symbolization, the alphabet size $|\Sigma|$ must be appropriately chosen in order to transform the real-valued finite-length data set S into a symbol string. The data set S is partitioned into a (finite) number of (mutually exclusive and exhaustive) segments to construct a mapping between S and the alphabet of symbols $\{\sigma | \sigma \in \Sigma\}$. To do so, a choice must be made as to the number of symbols, i.e., the cardinality $|\Sigma|$ of the symbol alphabet Σ . Presented below is a brief discussion on how to make the tradeoff between information loss and computational complexity.

Let the alphabet size be $k = |\Sigma|$ and the method of partitioning the time series be maximum entropy partitioning [31], i.e., a uniform probability distribution on the symbols with $P(\sigma) = \frac{1}{k} \forall \sigma \in \Sigma$. Then, the information loss, represented by the negative of the entropy [21] of the symbol sequence, is given as

$$I = -H = \sum_{\sigma \in \Sigma} P(\sigma) \ln P(\sigma) = -\ln k \quad (11)$$

By representing the computational complexity as a function $g(k)$ of the alphabet size k and choosing an appropriate scalar tradeoff weighting parameter $\alpha \in (0, 1)$, the cost functional to be optimized becomes:

$$J(k) = -\alpha \ln k + (1 - \alpha) g(k) \quad (12)$$

The optimal alphabet size $|\Sigma|$ is obtained by solving for k in the equation $J(k + 1) - J(k) = 0$ along with additional constraints that may have to be imposed in the optimization procedure to realize the effects of critical issues such as any bounds on the alphabet size.

B. Entropy rate

This subsection introduces the notion of entropy rate that, given the current state, represents the predictability of PFSA.

Definition B.7 (Conditional Entropy and Entropy Rate [21]) The entropy of a PFSA (Σ, Q, δ, π) conditioned on the current state $q \in Q$ is defined as follows.

$$H(\Sigma|q) \triangleq \sum_{\sigma \in \Sigma} P(\sigma|q) \log P(\sigma|q) \quad (13)$$

The entropy rate of a PFSA (Σ, Q, δ, π) is defined in terms of the conditional entropy as follows.

$$\begin{aligned} H(\Sigma|Q) &\triangleq \sum_{q \in Q} P(q) H(\Sigma|q) \\ &= - \sum_{q \in Q} \sum_{\sigma \in \Sigma} P(q) P(\sigma|q) \log P(\sigma|q) \end{aligned} \quad (14)$$

where $P(q)$ is the (unconditional) probability of a PFSA state $q \in Q$; and $P(\sigma|q)$ is the (conditional) probability of a symbol $\sigma \in \Sigma$ emanating from the PFSA state $q \in Q$.

C. Metric for the distance between two PFSA

This subsection introduces the notion of a metric to quantify the distance between two PFSA.

Definition B.8 (Metric) Let $K_1 = (\Sigma, Q_1, \delta_1, \pi_1)$ and $K_2 = (\Sigma, Q_2, \delta_2, \pi_2)$ be two PFSA with a common alphabet Σ . Let $P_1(\Sigma^j)$ and $P_2(\Sigma^j)$ be the steady state probability vectors of generating words of length j from the PFSA K_1 and K_2 , respectively, i.e., $P_1(\Sigma^j) \triangleq [P(w)]_{w \in \Sigma^j}$ for K_1 and $P_2(\Sigma^j) \triangleq [P(w)]_{w \in \Sigma^j}$ for K_2 . Then, the metric for the distance between the PFSA K_1 and K_2 is defined as

$$\Phi(K_1, K_2) \triangleq \lim_{n \rightarrow \infty} \sum_{j=1}^n \frac{\|P_1(\Sigma^j) - P_2(\Sigma^j)\|_{\ell_1}}{2^{j+1}} \quad (15)$$

where the norm $\|\cdot\|_{\ell_1}$ indicates the sum of absolute values of the elements in the vector \cdot .

The norm on the right side of Eq. (15) yields

$$\|P_1(\Sigma^j) - P_2(\Sigma^j)\|_{\ell_1} \leq \|P_1(\Sigma^j)\|_{\ell_1} + \|P_2(\Sigma^j)\|_{\ell_2} = 2$$

because each of the probability vectors $P_1(\Sigma^j)$ and $P_2(\Sigma^j)$ has non-negative entries that sum to 1. Furthermore, convergence of the infinite sum on the right side of Eq. (15) is guaranteed due to the weight $\frac{1}{2^{j+1}}$ and satisfies the relation $0 \leq \Phi(\cdot, \cdot) \leq 1$. It is noted that alternative forms of norms could also be used, because of norm equivalence in finite-dimensional vector spaces.

Since the metric in Definition B.8 assigns more weight to words of smaller length, the infinite sum could be truncated to a relatively small order D (e.g., typically in the range of 5 to 20) [31] for a given tolerance $\epsilon \ll 1$. That is, the distance $\Phi(\cdot, \cdot)$ in Eq. (15) effectively compares the probabilities of generating words of length D in two

PFSA and is especially adaptable to D -Markov machines whose dynamical behavior is characterized by words of a specified maximal depth [13].

The metric $\Phi(\cdot, \cdot)$ can also be used to calculate the distance between a PFSA and a symbol string, in which case the probabilities are expressed in terms of relative frequency of occurrence of a word.

APPENDIX C

STATE SPLITTING & STATE MERGING ALGORITHMS

This appendix lists the algorithms of state splitting and state merging for construction probabilistic finite state state automata (PFSA) from symbol strings, where state splitting is addressed by Algorithm 1 and state merging by Algorithms 2, 3 and 4.

Algorithm 1 State splitting

Input: Symbol sequence $s_1 s_2 s_3 \dots$, where each s_i belongs to the symbol alphabet Σ

User defined Parameters: Maximum number of states N_{max} and threshold η_{spl} for state splitting

Output: PFSA $K = \{\Sigma, Q, \delta, \pi\}$

Initialize: Create a 1-Markov machine $\tilde{Q} := \Sigma$

repeat

$Q := \tilde{Q}$;

$\tilde{Q} = \arg \min_{Q'} H(\Sigma | Q')$;

where $Q' = (Q \setminus \{q\}) \cup (\{\sigma q : \sigma \in \Sigma\})$ and $q \in Q$

until $|\tilde{Q}| < N_{max}$ or $H(\Sigma | Q) - H(\Sigma | \tilde{Q}) < \eta_{spl}$

for all $q \in \tilde{Q}$ and $\sigma \in \Sigma$ **do**

if $\delta(q, \sigma)$ is not unique **then**

$\tilde{Q} := (\tilde{Q} \setminus \{q\}) \cup (\{\sigma q : \sigma \in \Sigma\})$;

end if

end for

return $K = \{\Sigma, Q, \delta, \pi\}$

Algorithm 2 Minimal equivalence relation given $q \sim q'$

Input: δ, q, q' , Initial equivalence relation \sim

Output: Updated equivalence relation \sim

NOTE: Recursive function $(\sim) := \text{Merge}(\delta, q, q', \sim)$

Set $q \sim q'$;

for all $\sigma \in \Sigma$ **do**

if $\delta(q, \sigma) \approx \delta(q', \sigma)$ **then**

Set $\sim := \text{Merge}(\delta, \delta(q, \sigma), \delta(q', \sigma), \sim)$;

end if

end for

return \sim

Algorithm 3 Minimal PFSA K_2 after merging of two states

Input: $K_1 = \{\Sigma, Q_1, \delta_1, \pi_1\}, q, q'$
Output: Merged PFSA $K_2 = \{\Sigma, Q_2, \delta_2, \pi_2\}$
 Compute the equivalence relation \sim using Algorithm 2;
 Set $Q_2 := Q_1 / \sim$; % Q_2 is the quotient set of Q_1 under \sim
 Compute the stationary-probability vector \hat{P}_1 of the PFSA K_1 ;
for all $[q] \in Q_2$ **do**
 for all $\sigma \in \Sigma$ **do**
 Set $\delta_2([q], \sigma) := [\delta_1(q, \sigma)]$;
 Compute $\pi_2([q], \sigma)$ using Eq. (7);
 end for
end for
return $K_2 = \{\Sigma, Q_2, \delta_2, \pi_2\}$

Algorithm 4 State merging in PFSA

Input: PFSA $K = \{\Sigma, Q, \delta, \pi\}$ and symbol sequence $\{s_i\}$
User defined Parameter: Threshold η_{mrg} for state merging
Output: Merged PFSA $K_m = \{\Sigma, Q_m, \delta_m, \pi_m\}$
 Set $K_m := K$;
for all $q, q' \in Q_m$ **do**
 if $q \uparrow q'$ **then**
 Set LIST_STATES $(q, q') = \mathcal{M}(q, q')$ using Eq. (9);
 else
 Set LIST_STATES $(q, q') = 2$;
 end if
end for
sort (LIST_STATES); % Place the pair (q, q') with the smallest $\mathcal{M}(q, q')$ on top of the list
 Set $(q, q') := \mathbf{pop}$ (LIST_STATES); % Select the pair (q, q') that is on top of the sorted list
loop
 Compute K_1 from K_m by merging the states q and q' via Algorithm 3;
 if $d[K_1, \{s_i\}] < \eta_{mrg}$ **then**
 Set $K_m := K_1$;
 Recompute LIST_STATES;
 Set $(q, q') := \mathbf{pop}$ (LIST_STATES);
 else
 Set $(q, q') := \mathbf{pop}$ (LIST_STATES);
 if $q = q'$ **then**
 Break loop;
 end if
 end if
end loop
return $K_m = \{\Sigma, Q_m, \delta_m, \pi_m\}$

REFERENCES

- [1] S. D. Zilwa, J. H. Uhm, and J. H. Whitelaw, "Combustion oscillations close to the lean flammability limit," *Combustion Science and Technology*, 2000, 160, 231–258.
- [2] S. Chaudhuri and B. M. Cetegen, "Blowoff characteristics of bluff body stabilized conical premixed flames under upstream mixture gradients and velocity oscillations," *Combustion and Flame*, 2008, 153, 616–633.
- [3] S. Chaudhuri and B. M. Cetegen, "Blowoff characteristics of bluff body stabilized conical premixed flames in a duct with upstream mixture gradients and velocity oscillations," *Combustion Science and Technology*, 2009, 181, 555–569.
- [4] S. Chaudhuri, S. Kostska, M. W. Renfro, and B. M. Cetegen, "Blowoff dynamics of bluff body stabilized turbulent premixed flames," *Combustion and Flame*, 2010, 157, 790–802.
- [5] M. Stohr, I. Boxx, C. Carter, and W. Meier, "Dynamics of lean blowout of a swirl-stabilized flame in a gas turbine model combustor," *Proceedings of the Combustion Institute*, 2011, 33, 2953–2960.
- [6] S. Nair, T. M. Muruganandam, R. Olsen, A. Meyers, J. Seitzman, B. T. Zinn, T. Lieuwen, T. Held, and H. Mongia, "Lean blowout detection in a single nozzle swirl cup combustor," in *42nd AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 2004*, AIAA 2004–0138.
- [7] T. M. Muruganandam, S. Nair, D. Scarborough, Y. Neumeier, J. Jagoda, T. Lieuwen, J. Seitzman, and B. T. Zinn, "Active control of lean blowout for turbine engine combustors," *Journal of Propulsion and Power*, 2005, 21, 807–815.
- [8] S. Nair and T. Lieuwen, "Acoustic detection of blowout in premixed flames," *Journal of Propulsion and Power*, 2005, 21, 32–39.
- [9] S. Prakash, S. Nair, T. M. Muruganandam, Y. Neumeier, T. Lieuwen, J. M. Seitzman, and B. T. Zinn, "Acoustic based rapid blowout mitigation in a swirl stabilized combustor," in *ASME Turbo Expo, Reno, USA, 2005*.
- [10] T. Yi and E. J. Gutmark, "Real-time prediction of incipient lean blowout in gas turbine combustors," *AIAA Journal*, 2007, 45, 1734–1739.
- [11] R. R. Chaudhari, R. Sahu, S. Ghosh, A. Mukhopadhyay, and S. Sen, "Flame color as a lean blowout predictor," *International Journal of Spray and Combustion Dynamics*, 2013, 5, 49–66.
- [12] A. Mukhopadhyay, R. Chaudhuri, T. Paul, S. Sen, and A. Ray, "Prediction of lean blow-out in gas turbine combustors using symbolic time series analysis," *Journal of Propulsion and Power (AIAA)*, 2013, 29, 950–960.
- [13] A. Ray, "Symbolic dynamic analysis of complex systems for anomaly detection," *Signal Processing*, July 2004, 84, 1115–1130.
- [14] K. Mukherjee and A. Ray, "State splitting and state merging in probabilistic finite state automata for signal representation and analysis," *Signal Processing*, November 2014, 104, 105–119.

- [15] L. Kabiraj, R. I. Sujith, and P. Wahi, "Investigating dynamics of combustion driven oscillations leading to lean blowout," *Fluid Dynamics Research*, 2012, 44.
- [16] L. Kabiraj and R. I. Sujith, "Nonlinear self-excited thermoacoustic oscillations: intermittency and flame blowout," *J. Fluid Mech.*, 2012, 713, 376–397.
- [17] H. Gotoda, M. Amano, T. Miyano, T. Ikawa, K. Maki, and S. Tachibana, "Characterization of complexities in combustion instability in a lean premixed gas-turbine model combustor," *Chaos*, 2012, 22, 043128.
- [18] H. Gotoda, Y. Shinoda, M. Kobayashi, Y. Okuno, and S. Tachibana, "Detection and control of combustion instability based on the concept of dynamical system theory," *Phys. Rev. E*, 2014, 89, 022910.
- [19] V. Nair, G. Thampi, S. Karuppusamy, S. Gopalan, and R. I. Sujith, "Loss of chaos in combustion noise as a precursor of impending combustion instability," *International Journal of Spray and Combustion Dynamics*, 2013, 5, 273–290.
- [20] F. Darema, "Dynamic data driven applications systems: new capabilities for application simulations and measurements". In: *5th International conference on computational science – ICCS 2005*, Atlanta, GA (United States); May 2005, 610–615.
- [21] T. Cover and J. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken, NJ, USA: Wiley, 2006.
- [22] Y. Wen, K. Mukherjee, and A. Ray, "Adaptive pattern classification for symbolic dynamic systems," *Signal Processing*, 2013, 93, 252–260.
- [23] R. Chaudhari, *Investigation of Thermoacoustic Instabilities and Lean Blowout in a Model Gas Turbine Combustor*. PhD thesis, Jadavpur University, India, 2011. Ph.D. Dissertation, Department of Mechanical Engineering.
- [24] T. Williams, R. Schefer, J. Oefelein, and C. Shaddix, "Idealized gas turbine combustor for performance research and validation of large eddy simulations," *Review of Scientific Instruments*, 2007, 78, 035114–1–9.
- [25] W. Meier, P. Weigand, X. Duan, and R. Giezendanner-Thobben, "Detailed characterization of the dynamics of thermoacoustic pulsations in a lean premixed swirl flame," *Combustion and Flame*, 2007, 150, 2–26.
- [26] S. Turns, *An Introduction to Combustion*. NYC, NY, USA: McGraw-Hill, 2 ed., 2000.
- [27] C. Daw, M. Kennel, C. Finney, and F. Connolly, "Observing and modeling nonlinear dynamics in an internal combustion engine," *Physical Review. E*, 1998, 57, 2811–2819.
- [28] D. Lind and B. Marcus, *Symbolic Dynamics and Coding*. Cambridge University Press, Cambridge, UK, 1995.
- [29] P. Dupont, F. Denis, and Y. Esposito, "Links between probabilistic automata and hidden Markov models: probability distributions, learning models and induction algorithms," *Pattern Recognition*, 2005, 38, 1349–1371.

- [30] E. Vidal, F. Thollard, C. de la Higuera, F. Casacuberta, and R. Carrasco, “Probabilistic finite-state machines - Part I and Part II,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27, 1013–1039.
- [31] V. Rajagopalan and A. Ray, “Symbolic time series analysis via wavelet-based partitioning,” *Signal Processing*, 2006, 86, 3309–3320.
- [32] C. Rao, A. Ray, S. Sarkar, and M. Yasar, “Review and comparative evaluation of symbolic dynamic filtering for detection of anomaly patterns,” *Signal, Image and Video Processing*, 2009, 3, 101–114.
- [33] S. Bahrapour, A. Ray, S. Sarkar, T. Damarla, and N. Nasrabadi, “Performance comparison of feature extraction algorithms for target detection and classification,” *Pattern Recognition Letters*, December 2013, 34, 2126–2134.
- [34] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, New York, NY, USA, 2006.
- [35] T. Liao, “Clustering of time series data – A survey,” *Pattern Recognition*, 2005, 38, 1857–1874.
- [36] C. Shalizi and K. Shalizi, “Blind construction of optimal nonlinear recursive predictors for discrete sequences,” in *AUAI '04: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, (Arlington, VA, USA), AUAI Press, 2004, 504–511.
- [37] M. Sipser, *Introduction to the Theory of Computation, 3rd ed.* Boston, MA, USA: Cengage Publishing, 2013.
- [38] G. Mallapragada, A. Ray, and X. Jin, “Symbolic dynamic filtering and language measure for behavior identification of mobile robots,” *IEEE Transactions on System, Man, and Cybernetics Part B: Cybernetics*, June 2012, 42, 647–659.
- [39] C. Beck and F. Schlögl, *Thermodynamics of chaotic systems: an introduction*. Cambridge, U.K: Cambridge University Press, 1993.
- [40] P. Halmos, *Naive Set Theory*, 26–29. Princeton, NJ: Van Nostrand, 1960.
- [41] H. Ebrahimi, “Overview of gas turbine augmentor design, operation and combustion oscillation,” in *ILASS Americas, 19th Annual Conference on Liquid Atomization and Spray Systems, Toronto, Canada*, 2006.