

Symbolic Dynamic Analysis of Transient Time Series for Fault Detection in Gas Turbine Engines

Soumalya Sarkar

e-mail: sv5464@psu.edu

Kushal Mukherjee¹

Mem. ASME

e-mail: kushal.mukherjee@gmail.com

Soumik Sarkar²

Mem. ASME

e-mail: sarkars@utrc.utc.com

Asok Ray

Fellow ASME

e-mail: axr2@psu.edu

Department of Mechanical Engineering,
The Pennsylvania State University,
University Park, PA 16802

This brief paper presents a symbolic dynamics-based method for detection of incipient faults in gas turbine engines. The underlying algorithms for fault detection and classification are built upon the recently reported work on symbolic dynamic filtering. In particular, Markov model-based analysis of quasi-stationary steady-state time series is extended to analysis of transient time series during takeoff. The algorithms have been validated by simulation on the NASA Commercial Modular Aero Propulsion System Simulation (C-MAPSS) transient test-case generator. [DOI: 10.1115/1.4007699]

Keywords: transient time-series analysis, symbolic dynamics, fault detection, aircraft gas turbine engines

1 Introduction

Performance monitoring of aircraft gas turbine engines is typically conducted on quasi-stationary steady-state data collected during cruise conditions. This issue has been addressed by several researchers by using a variety of analytical tools. For example, Lipowsky et al. [1] made use of Bayesian forecasting for change detection and performance analysis of gas turbine engines. Guruprakash and Ganguli [2] reported optimally weighted recursive neural filtering for gas turbine diagnostics from noisy signals. However, transient operational data (e.g., from takeoff, climb, or landing) can be gainfully utilized for early detection of incipient faults. Usually engines operate at much higher stress and temperature conditions under transient operations compared to those at steady-state conditions. Signatures of certain incipient engine faults (e.g., bearing faults, controller miss-scheduling, and starter system faults) tend to magnify during the transient conditions [3], and appropriate usage of transient data could enhance the probability of fault detection [4]. Along this line, several model-based and data-driven fault diagnosis methods have been proposed by

making use of transient data. For example, Wang et al. [5] developed a neural network-based fault diagnosis method for automotive transient operations, and Surender and Ganguli [6] used an adaptive myriad filter to improve the quality of transient operations for gas turbine engines. However, model-based diagnostics may suffer from inaccuracy and loss of robustness due to low reliability of the transient models. This problem is partially circumvented through usage of data-driven diagnostics; for example, Menon et al. [7] used hidden Markov models (HMMs) for transient analysis of gas turbine engines.

A recently developed data-driven technique, called the symbolic dynamic filtering (SDF) [8], has been shown to yield superior performance in terms of early detection of anomalies and robustness to measurement noise in comparison to other techniques such as principal component analysis (PCA), neural networks (NN), and Bayesian techniques [9]. Recently, in a two-part paper [10,11], an SDF-based algorithm for detection and isolation of engine subsystem faults (specifically, faults that cause efficiency degradation in engine components) has been reported and an extension of that work to estimate simultaneously occurring multiple component-level faults has been presented in Ref. [12]. Furthermore, an optimized feature extraction technique has been developed under the same semantic framework in Ref. [13]. However, all of the above studies were conducted on steady-state cruise flight data that conform with the assumption of quasi-stationarity made in SDF. Due to this assumption, SDF may not be able to adequately handle transient data that are usually of limited length.

The goal of this brief paper is to extend SDF's capability to handle (short-length) transient data, beyond what is currently done for quasi-stationary data sets of sufficient length. This paper uses Dirichlet and multinomial distributions to construct the a priori and a posteriori models of uncertainties, respectively. The algorithms are formulated by quantitatively incorporating the effects of finite-length symbol strings in both training and testing phases of fault detection. The resulting algorithm is validated using the *transient test-case generator* [14] of the C-MAPSS test bed [15], developed by NASA.

2 Symbolic Analysis of Time-Series Data

A symbol string is obtained from the output of gas turbine engine dynamics by partitioning, also called quantization, of the time-series data. Thereafter, a probabilistic finite state automaton (PFSA) model is developed from the (finite-length) symbol sequence [8]. This section constructs a Bayesian classifier for identification of the probability morph matrices of PFSA based on the transient data in both training and testing phases. The Dirichlet and multinomial distributions have been used to construct the a priori and a posteriori models of uncertainties, respectively. This formulation quantitatively incorporates the effects of finite-length symbol strings in both training and testing phases of pattern classification.

2.1 Partitioning of Time-Series Data. The sensor time series is encoded by data partitioning in the range of the signal [13,16], where the conversion to symbol strings is achieved by substituting each (real-valued) data point in the time series by a symbol corresponding to the region (i.e., interval) within which the data point lies. This step enables transformation of the sensory information from the continuous domain to the symbolic domain; in other words, the sensor data at each sampling instant is replaced by a symbol. Sarkar et al. [12,13] have reported details of data partitioning and its usage for fault detection in gas turbine engines.

2.2 Modeling via Probabilistic Finite State Automata. The symbolic sequence is modeled as a PFSA that is constructed as a tuple $G \triangleq (Q, \Sigma, \delta, \Pi)$, where the alphabet Σ is a nonempty finite set of symbols and the set of states Q is constrained to be

¹Present address: United Technologies Research Center, Cork, Ireland.

²Present address: United Technologies Research Center, East Hartford, USA.

Contributed by the Dynamic Systems Division of ASME for publication in the JOURNAL OF DYNAMIC SYSTEMS, MEASUREMENT, AND CONTROL. Manuscript received October 11, 2011; final manuscript received July 27, 2012; published online November 7, 2012. Assoc. Editor: Eugenio Schuster.

nonempty and finite. To simplify the PFSA model construction, this paper considers only a class of PFSA, known as D-Markov machines [8], where the states are strings of the D past symbols; the positive integer D is called the depth of the machine and the number of states $|Q| \leq |\Sigma|^D$. Given the previous state and an observed symbol, the state transition function $\delta : Q \times \Sigma \rightarrow Q$ yields the new state. In addition, the morph function $\pi : Q \times \Sigma \rightarrow [0, 1]$ is an output mapping that satisfies the condition: $\sum_{\sigma \in \Sigma} \pi(q, \sigma) = 1$ for all $q \in Q$. The morph function π has a matrix representation Π , called the (probability) morph matrix of dimension $(|Q| \times |\Sigma|)$. Each row sum of Π is equal to 1 and each matrix element Π_{ij} is strictly positive due to the finite-length constraint of time series from which PFSA models are constructed [17,18].

Remark 2.1. The PFSA $G \triangleq (Q, \Sigma, \delta, \Pi)$ is not dependent on the initial state $q_0 \in Q$ due to the assumption of quasi-stationarity of the observed sensor data.

2.3 The Online Classification Problem. Let there be K symbolic systems (i.e., classes) of interest, denoted by C_1, C_2, \dots, C_K , over the same alphabet Σ . Each class C_i is modeled by an ergodic (equivalently, irreducible) PFSA $G^i = (Q^i, \Sigma, \delta^i, \Pi^i)$, where $i = 1, 2, \dots, K$.

During the training phase, a symbol string $S^i \triangleq s_1^i s_2^i \dots s_{N_i}^i$ is generated from each class C_i . The state transition function δ and the set of states Q of the D-Markov machine are fixed by choosing an appropriate depth D . Thus, Π^i 's become the only unknowns and could be selected as the feature vectors for the purpose of classification. The distribution of the morph matrix Π^i is computed in the training phase from the finite-length symbol sequences for each class.

In the testing phase, let another symbol string \tilde{S} be obtained from a sensor time-series data. Then, the task is to determine which class this observed symbol string \tilde{S} belongs to. While the previous work [8,16] has aimed at identification of a PFSA from a given symbol string, the objective of this paper is to imbed the uncertainties due to the finite length of the symbol string in the identification algorithm that would influence the final classification decision.

In the training phase, each row of Π^i is treated as a random vector. Let the m th row of Π^i be denoted as Π_m^i and the n th element of the m th row as $\Pi_{mn}^i > 0$ and $\sum_{n=1}^{|\Sigma|} \Pi_{mn}^i = 1$. The a priori probability density function $f_{\Pi_m^i|S^i}$ of the random row-vector Π_m^i , conditioned on a symbol string S^i , follows the Dirichlet distribution [19,20] as described below

$$f_{\Pi_m^i|S^i}(\theta_m^i|S^i) = \frac{1}{B(\alpha_m^i)} \prod_{n=1}^{|\Sigma|} (\theta_{mn}^i)^{\alpha_{mn}^i - 1} \quad (1)$$

where θ_m^i is a realization of the random vector Π_m^i , namely

$$\theta_m^i = [\theta_{m1}^i \quad \theta_{m2}^i \quad \dots \quad \theta_{m|\Sigma|}^i]$$

and the normalizing constant is

$$B(\alpha_m^i) \triangleq \frac{\prod_{n=1}^{|\Sigma|} \Gamma(\alpha_{mn}^i)}{\Gamma\left(\sum_{n=1}^{|\Sigma|} \alpha_{mn}^i\right)} \quad (2)$$

where $\alpha_m^i \triangleq [\alpha_{m1}^i \alpha_{m2}^i \dots \alpha_{m|\Sigma|}^i]$ with $\alpha_{mn}^i = N_{mn}^i + 1$ and N_{mn}^i is the number of times the symbol σ_n in S^i is emanated from the state q_m , i.e.,

$$N_{mn}^i \triangleq |\{(s_k^i, v_k^i) : s_k^i = \sigma_n, v_k^i = q_m\}| \quad (3)$$

where s_k^i is the k th symbol in S^i and v_k^i is the k th state as derived from the symbolic sequence S^i . Recall that a state is defined as a string of D past symbols. Then, the number of occurrence of the state q_m in the state sequence is given by $N_m^i \triangleq \sum_{n=1}^{|\Sigma|} N_{mn}^i$. It follows from Eq. (2) that

$$B(\alpha_m^i) = \frac{\prod_{n=1}^{|\Sigma|} \Gamma(N_{mn}^i + 1)}{\Gamma\left(\sum_{n=1}^{|\Sigma|} N_{mn}^i + |\Sigma|\right)} = \frac{\prod_{n=1}^{|\Sigma|} (N_{mn}^i)!}{(N_m^i + |\Sigma| - 1)!} \quad (4)$$

by use of the relation $\Gamma(n) = (n-1)! \quad \forall n \in \mathbb{N}_1$.

By the Markov property of the PFSA G^i , the $(1 \times |\Sigma|)$ row-vectors, $\{\Pi_m^i\}, m = 1, \dots, |Q|$, are statistically independent of each other. Therefore, it follows from Eqs. (1) and (4) that the a priori joint density $f_{\Pi^i|S^i}$ of the probability morph matrix Π^i , conditioned on the symbol string S^i , is given as

$$\begin{aligned} f_{\Pi^i|S^i}(\theta^i|S^i) &= \prod_{m=1}^{|Q|} f_{\Pi_m^i|S^i}(\theta_m^i|S^i) \\ &= \prod_{m=1}^{|Q|} (N_m^i + |\Sigma| - 1)! \prod_{n=1}^{|\Sigma|} \frac{(\theta_{mn}^i)^{N_{mn}^i}}{(N_{mn}^i)!} \end{aligned} \quad (5)$$

where $\theta^i = [(\theta_1^i)^T (\theta_2^i)^T \dots (\theta_{|Q|}^i)^T] \in [0, 1]^{|Q| \times |\Sigma|}$

In the testing phase, the probability of observing a symbol string \tilde{S} belonging to a particular class of PFSA $(Q, \Sigma, \delta, \Pi^i)$ is a product of independent multinomial distribution [21] given that the exact morph matrix Π^i is known.

$$\begin{aligned} \Pr(\tilde{S}|Q, \delta, \Pi^i) &= \prod_{m=1}^{|Q|} (\tilde{N}_m)! \prod_{n=1}^{|\Sigma|} \frac{(\Pi_{mn}^i)^{\tilde{N}_{mn}}}{(\tilde{N}_{mn})!} \end{aligned} \quad (6)$$

$$\triangleq \Pr(\tilde{S}|\Pi^i) \text{ as } Q \text{ and } \delta \text{ are kept invariant} \quad (7)$$

Similar to N_{mn}^i defined earlier for S^i , \tilde{N}_{mn} is the number of times the symbol σ_n is emanated from the state $q_m \in Q$ in the symbol string \tilde{S} in the testing phase, i.e.,

$$\tilde{N}_{mn} \triangleq |\{(\tilde{s}_k, \tilde{v}_k) : \tilde{s}_k = \sigma_n, \tilde{v}_k = q_m\}| \quad (8)$$

where \tilde{s}_k is the k th symbol in the string \tilde{S} and \tilde{v}_k is the k th state derived from \tilde{S} . It is noted that $\tilde{N}_m \triangleq \sum_{n=1}^{|\Sigma|} \tilde{N}_{mn}$.

The results, derived in the training and testing phases, are now combined. Given a symbol string S^i in the training phase, the probability of observing a symbol string \tilde{S} in the testing phase is obtained as follows:

$$\begin{aligned} \Pr(\tilde{S}|S^i) &= \int \dots \int \Pr(\tilde{S}|\Pi^i = \theta^i) f_{\Pi^i|S^i}(\theta^i|S^i) d\theta^i \\ &= \int \dots \int \left[\prod_{m=1}^{|Q|} (\tilde{N}_m)! \prod_{n=1}^{|\Sigma|} \frac{(\theta_{mn}^i)^{\tilde{N}_{mn}}}{(\tilde{N}_{mn})!} \right] \\ &\quad \times \prod_{m=1}^{|Q|} \left[(N_m^i + |\Sigma| - 1)! \prod_{n=1}^{|\Sigma|} \frac{(\theta_{mn}^i)^{N_{mn}^i}}{(N_{mn}^i)!} d\theta_{mn}^i \right] \\ &= \prod_{m=1}^{|Q|} (\tilde{N}_m)! (N_m^i + |\Sigma| - 1)! \\ &\quad \times \frac{\int \dots \int \prod_{n=1}^{|\Sigma|} (\theta_{mn}^i)^{\tilde{N}_{mn} + N_{mn}^i} d\theta_{mn}^i}{\prod_{n=1}^{|\Sigma|} (\tilde{N}_{mn})! (N_{mn}^i)!} \end{aligned} \quad (9)$$

The integrand in Eq. (9) is the density function for the Dirichlet distribution up to the multiplication of a constant. Hence, it follows from Eq. (4) that

$$\int \dots \int \prod_{n=1}^{|\Sigma|} (\theta_{mn}^{i})^{\tilde{N}_{mn} + N_{mn}^i} d(\theta_{mn}^i) = \frac{\prod_{n=1}^{|\Sigma|} (\tilde{N}_{mn} + N_{mn}^i)!}{(\tilde{N}_m + N_m^i + |\Sigma| - 1)!}$$

Then, it follows from Eq. (9) that

$$\Pr(\tilde{S}|S^i) = \prod_{m=1}^{|\mathcal{Q}|} \frac{(\tilde{N}_m)! (N_m^i + |\Sigma| - 1)!}{(\tilde{N}_m + N_m^i + |\Sigma| - 1)!} \prod_{n=1}^{|\Sigma|} \frac{(\tilde{N}_{mn} + N_{mn}^i)!}{(\tilde{N}_{mn})! (N_{mn}^i)!} \quad (10)$$

It might be easier to compute the logarithm of $\Pr(\tilde{S}|S^i)$ by using Stirling's approximation formula $\log(n!) \approx n \log(n) - n$ [22] because both N^i and \tilde{N} would be large numbers. The posterior probability of a symbol string S belonging to the class C_i is denoted as $\Pr(C_i|\tilde{S})$ and is given as

$$\Pr(C_i|\tilde{S}) = \frac{\Pr(\tilde{S}|S^i) \Pr(C_i)}{\sum_{j=1}^K \Pr(\tilde{S}|S^j) \Pr(C_j)}, \quad i = 1, 2, \dots, K \quad (11)$$

where $\Pr(C_i)$ is the known prior distribution of the class C_i . Then, the classification decision is made as follows:

$$\begin{aligned} D_{\text{class}} &= \arg \max_i \Pr(C_i|\tilde{S}) \\ &= \arg \max_i (\Pr(\tilde{S}|S^i) \Pr(C_i)) \end{aligned} \quad (12)$$

Algorithms 1 and 2 respectively summarize the training and testing phases for classification of time-series data.

Algorithm 1 Training

Input: Time-series data, one for each of the K classes.

User defined Input: Symbol alphabet Σ , depth D of PFSA, where the symbols are defined $\sigma_1, \sigma_2, \dots, \sigma_{|\Sigma|}$ and the states are defined as $q_1, q_2, \dots, q_{|\mathcal{Q}|}$ with $|\mathcal{Q}| \leq |\Sigma|^D$.

Output: N_{mn}^i , where $i = 1, \dots, K$, $n = 1, \dots, |\Sigma|$ and $m = 1, \dots, |\mathcal{Q}|$.

Initialize: All $N_{mn}^i = 0$.

for all $i \in \{1, 2, \dots, K\}$ **do**

 Symbolize time-series data for class i to obtain a symbol sequence S^i .

 Obtain N_{mn}^i , the number of times the symbol σ_n in S^i is observed immediately after the state q_m .

end for

return N_{mn}^i , where $i = 1, \dots, K$, $n = 1, \dots, |\Sigma|$ and $m = 1, \dots, |\mathcal{Q}|$.

Algorithm 2 Testing

Input: Test time-series data, and output of training N_{mn}^i .

User defined Input: Prior probability of each class $\Pr(C_i)$, symbol alphabet Σ , and depth D of PFSA, the symbols are defined as $\sigma_1, \sigma_2, \dots, \sigma_{|\Sigma|}$, and states are defined as $q_1, q_2, \dots, q_{|\mathcal{Q}|}$, where $|\mathcal{Q}| \leq |\Sigma|^D$.

Output: Posterior probability of each class.

Initialize: All $\tilde{N}_{mn}^i = 0$

Symbolize test time-series to generate symbol sequence \tilde{S}^i .

Obtain \tilde{N}_{mn}^i , the number of times the symbol σ_n in \tilde{S}^i is observed immediately after the state q_m is reached.

for all $i \in \{1, 2, \dots, K\}$ **do**

 Evaluate $\Pr(\tilde{S}^i|S^i)$ using Eq. (10).

end for

for all $i \in \{1, 2, \dots, K\}$ **do**

 Evaluate $\Pr(C_i|\tilde{S}^i)$ using Eq. (11)

end for

return $\Pr(C_i|\tilde{S}^i)$, $i \in \{1, 2, \dots, K\}$.

3 Fault Injection in C-MAPSS Test Bed

The NASA C-MAPSS simulation test bed [15] has been developed for a typical commercial-scale two-spool turbofan engine and its control system. The engine under consideration produces a thrust of approximately 400,000 N and is designed for operation at altitude (A) from the sea level (i.e., 0 m) up to 12,200 m, Mach number (M) from 0 to 0.90, and temperatures from approximately -50°C to 50°C . The throttle resolving angle (TRA) can be set to any value in the range between 0 deg at the minimum power level and 100 deg at the maximum power level. The gas turbine engine system consists of five major rotating components, namely, fan (F), low pressure compressor (LPC), high pressure compressor (HPC), high pressure turbine (HPT), and low pressure turbine (LPT).

Given the inputs of TRA, A and M , the interactively controlled component models compute nonlinear dynamics of real-time turbofan engine operation. A gain-scheduled control system is incorporated in the engine system, which consists of speed controllers and limit regulators for engine components.

Out of the different types of sensors (e.g., pressure, temperature, and shaft speed) used in the C-MAPSS simulation test bed, Table 1 lists those sensors that are commonly adopted in the instrumentation and control system of commercial aircraft engines, as seen in Fig. 1.

In the current configuration of the C-MAPSS simulation test bed, there are 13 component-level health parameter inputs, namely, efficiency parameters (ψ), flow parameters (ζ), and pressure ratio modifiers that simulate the effects of faults and/or degradation in the engine components. Ten, out of the 13 health parameters [23], are selected to modify efficiency (η) and flow (ϕ) that are defined as

- $\eta \triangleq$ ratio of actual enthalpy and ideal enthalpy changes
- $\phi \triangleq$ ratio of rotor tip and axial fluid flow velocities

For each of the engine's five rotating components X , where X represents F, LPC, HPC, LPT, or HPT, there are two health parameters, namely, efficiency parameter (ψ_X) and flow parameter (ζ_X). If an engine component X is in the nominal condition, then both ψ_X and ζ_X are set to 1, and a fault can be injected in a component X by independently reducing the values of the parameters ψ_X and/or ζ_X . For example, $\psi_F = 0.98$ signifies a 2% relative loss in the fan efficiency.

An experimental-data-based stochastic damage model, called C-MAPSS *transient test-case generator* [14], has been developed at NASA for simulating natural deterioration and fault injection in the engine components. For all five rotating components, an injected fault exhibits a random magnitude F_m and a random health parameter ratio HPR. While F_m and HPR directly determine perturbations $\delta\psi_X$ in the efficiency health parameter ψ_X of a component X , perturbations $\delta\zeta_X$ in the flow health parameter ζ_X are determined by HPR for given perturbations $\delta\zeta_X$ based on the following relations.

$$\delta\psi_X \triangleq -\frac{F_m}{\sqrt{1 + \text{HPR}^2}} \quad \text{and} \quad \delta\zeta_X \triangleq \delta\psi_X \cdot \text{HPR} \quad (13)$$

Table 1 Sensor suite for the engine system

Sensors	Description
T_{24}	LPC exit/HPC inlet temperature
P_{S30}	HPC exit static pressure
T_{48}	HPT exit temperature
P_{50}	LPT exit pressure
N_f	Fan spool speed
N_c	Core spool speed

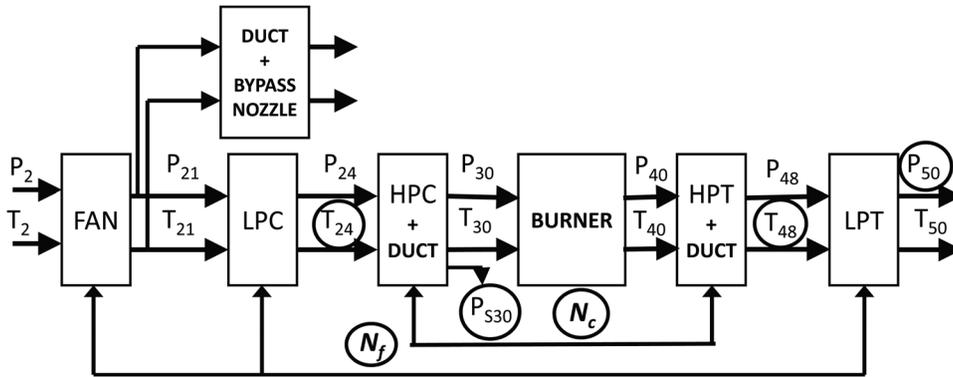


Fig. 1 Schematic diagram of the C-MAPSS engine model with sensors

In the case studies that follow in Sec. 4, (random) fault magnitudes F_m are uniformly distributed between 1.0 and 7.0. HPR for fan, LPC, and HPC are uniformly distributed between 1.0 and 2.0, whereas HPRs for HPT and LPT are uniformly distributed between 0.5 and 1.0. Perturbations in the efficiency (ψ_X) and flow (ζ_X) health parameters occur from their respective base values. The relative losses in the health parameters are expressed as follows: nominal health is represented by 100% of ψ_X and ζ_X ; for faulty fan, LPC and HPC, ψ_X is varied between 95% and 99.5% and ζ_X is varied between 90% and 99.5%; for faulty HPT and LPT, ψ_X is varied between 94% and 99% and ζ_X is varied between 93.5% and 99.5%.

4 Results and Discussion

This section presents the simulation results of symbolic analysis of transient time series and incipient fault detection on the C-MAPSS test bed. The goal here is to demonstrate how the binary faults (i.e., $K=2$ in Algorithms 1 and 2) are detected and identified at an early stage in time-critical operations like engine health management. To establish the relationship between detection time and detection accuracy, numerous simulation experiments have been performed during the “takeoff” operation, where the Mach number is varied from 0 to 0.24 in 60 s keeping the altitude at zero (i.e., sea level) and TRA at 80%. Faulty operations in three different components, fan, LPC and HPT, are simulated along with the ideal engine condition. The HPT exit temperature (T_{48}) sensor (that captures the above failure signatures in the gas path) is chosen to provide the transient response. The rationale behind choosing the T_{48} sensor is attributed to the following physical facts: (i) T_{48} is placed between HPT and LPT, and (ii) LPT is mechanically connected to the fan and LPC via the outer shaft. The sampling frequency of the T_{48} sensor data is 66.7 Hz. For the purpose of training, the duration for each simulation run is chosen to be 60 s (i.e., a time series of length ~ 4000) and the transient data set for each fault class is constructed by concatenating 50 such blocks of time-series data that were individually generated on the simulation test bed under similar transient operating conditions. It is noted that these blocks of data are statistically similar but they are not identical.

The next step is to partition the data sets to generate respective symbol strings. The range of the time series is partitioned into five intervals (i.e., the alphabet size $|\Sigma| = 5$), where each segment of the partitioning corresponds to a distinct symbol in the alphabet Σ . The training phase commences after the symbol strings are obtained for each of the four classes, i.e., one nominal and three faulty conditions. In this application, it suffices to choose the depth $D = 1$ [8], which implies that the probability of generation of a future symbol depends only on the last symbol, and hence the set of states is isomorphic to the symbol alphabet (i.e., $Q \equiv \Sigma$). For each class C_1 and C_2 , the parameters N_{mm}^i are obtained

by counting the number of times the symbol σ_n is emitted from state q_m .

The testing phase starts with a new time series from one of the classes, where the symbol sequence is generated by using the same alphabet and partitioning as in the training phase. Following Eq. (11), the posterior probability of each class is calculated as a function of the length of the testing data set. Figure 2 shows the posterior probability of each class as a function of the length of the observed test data. It is seen that the observed sequence is correctly identified to belong to the class of LPC faults as the posterior probability of the corresponding class approaches one, while it approaches zero for each of the remaining classes (i.e., the other two component faults and nominal condition). By repeating the same classification technique on 50 new test runs of LPC fault, it is observed that a data length of 500 is sufficient to detect the fault with a reasonable confidence. For other two component faults, the posterior probability for the correct fault class approaches unity within the data length of 50, as seen in Fig. 3. The rationale is that the fault signatures for fan and HPT in T_{48} response are dominant, even if they are incipient.

In a hierarchical fault detection and isolation strategy, at a low resolution level, the goal is to identify the faulty component (e.g., fan, LPC, or HPT). Once the fault is located, the next step is to quantify the severity of the located fault. The symbolic transient fault detection technique can be extended to classify different levels of fault in single components of a gas turbine engine. For verification, samples of nominal data are injected with a low-level fan fault, where the fault magnitude (F_m) follows a random uniform distribution ranging from 1 to 3. The remaining samples of nominal data are injected with a high-level fan fault of magnitude (F_m) within the range of 5–7. In this case, the alphabet size is chosen to be $|\Sigma| = 6$ to obtain better class separability. Figure 3 shows that the posterior probability for a high fan fault saturates

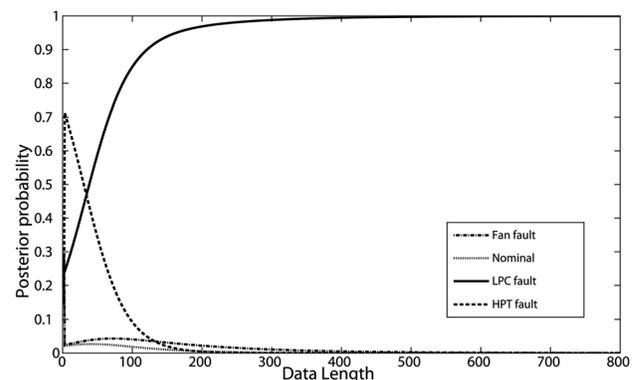


Fig. 2 Detection in a multifault framework (ground truth: a faulty LPC)

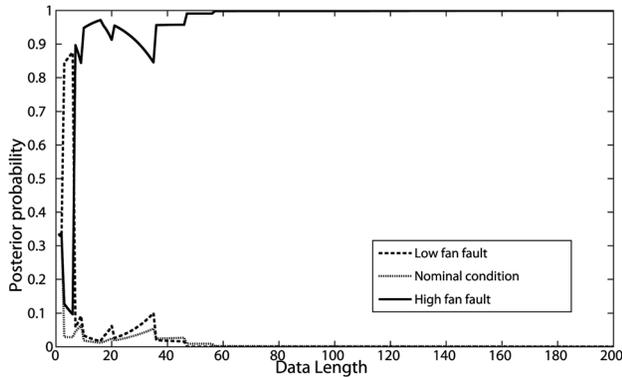


Fig. 3 High-level fan fault detection (ground truth: a high-level fan fault)

to the value of 1 within a data length of 70, which is equivalent to ~ 1 s. (Note: The oscillations in posterior probabilities in Fig. 3 are due to randomness of the estimated parameters for data lengths smaller than 50. But when the test case is a low-fan fault (see Fig. 4), the posterior probability of a high-level fan fault remains dominant until the data length reaches ~ 400 . This would result in a false classification as the posterior probability for the true class reaches 1 monotonically at around the data length of 1000 at, about 15 s, as seen in Fig. 4.) This result agrees well with the intuition that the detection time increases with smaller the fault levels.

To examine the performance of symbolic transient fault detection, a family of receiver operating characteristics (ROC) is constructed for different lengths of test data. A binary classification scenario is constructed, which consists of two classes: the nominal engine condition belonging to the class C_1 , and a faulty fan condition belonging to the class C_2 . The training data length is the same as described in the previous simulation runs. The general classification rule [24] in a symbol string \tilde{S} is given by

$$\frac{\Pr(\tilde{S}|C_1)}{\Pr(\tilde{S}|C_2)} \underset{C_2}{\overset{C_1}{\gtrless}} \lambda \quad (14)$$

where the threshold parameter λ is varied to generate the ROC curves. For the binary classification problem at hand, an ROC curve provides the trade-off between the probability of detection $P_D = \Pr\{\text{decide } C_2|C_2 \text{ is true}\}$ and the probability of false alarms $P_F = \Pr\{\text{decide } C_2|C_1 \text{ is true}\}$. Figure 5 exhibits a family of ROC curves for the proposed fault detection technique with varying lengths of test data. For each ROC curve, λ is varied between 0.01 and 2 with steps of 0.01. It is observed that the ROC curves yield improved performance (i.e., movement toward the top left corner) progressively as the test data length is increased from

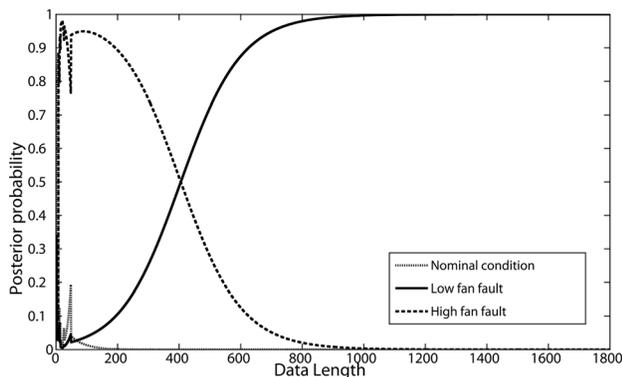


Fig. 4 Low-level fan fault detection (ground truth: a low-level fan fault)

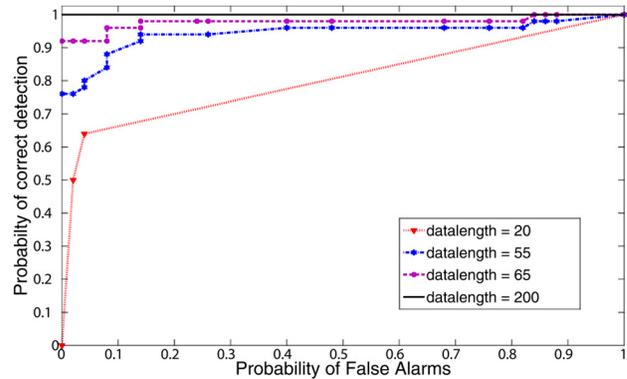


Fig. 5 ROC curves for fan fault identification with different test data lengths (λ varying between 0.01 and 2 with steps of 0.01)

$N_{\text{test}} = 20$ to $N_{\text{test}} = 200$. Thus, based on a family of such ROC curves, a good combination of P_D and N_{test} is obtained for a given P_F .

5 Summary, Conclusions, and Future Work

This paper addresses detection of incipient faults in gas turbine engines by analysis of pertinent transient time series. PFSA models of the engine's behavior have been constructed for fault identification in the context of SDF [8,16]. The proposed fault detection and classification method has been validated on the NASA C-MAPSS simulation test bed [15] for various operating conditions, and dependence of detection accuracy on the data length is reported. The major conclusions of the paper are as follows.

- (1) Extension of SDF's capability to handle (short-length) transient data, beyond what has been done for quasi-stationary data sets of sufficient length.
- (2) Usage of Dirichlet distribution and multinomial distribution for a priori and a posteriori models, respectively, to represent the uncertainties resulting from the finite length of symbol strings in both training and testing phases.

The next step is validation of the underlying algorithms with actual engine operation data, followed by on-board testing. Some of the topics of future research are listed below:

- *Sensor fusion in the SDF framework:* A single sensor may not capture small faults in components or actuators. Consequently, the sensor data may have to be observed for a longer period of time to assure accuracy of fault detection and classification. Sensor fusion using codependence among sensors in the SDF framework [25] is likely to increase accuracy and reduce delay.
- *Comparison with other model-based and data-driven methods of fault detection and classification.*
- *Extension of the fault detection and classification method to accommodate different types of sensor degradation.*
- *Testing and validation of the fault detection and classification method on a variety of test facilities.*

Acknowledgment

The authors acknowledge the benefits of technical discussions with Donald L. Simon and Jeffrey Armstrong of NASA Glenn Research Center, Cleveland, OH, USA. This work has been supported in part by NASA under Cooperative Agreement No. NNX07AK49A and the U.S. Army Research Laboratory and the U.S. Army Research Office under Grant No. W911NF-07-1-0376.

References

- [1] Lipowsky, H., Staudacher, S., Bauer, M., and Schmidt, K.-J., 2010, "Application of Bayesian Forecasting to Change Detection and Prognosis of Gas Turbine Performance," *ASME J. Eng. Gas Turbines Power*, **132**(3), p. 031602.
- [2] Guruprakash, V., and Ganguli, R., 2008, "Three-and Seven-Point Optimally Weighted Recursive Median Filters for Gas Turbine Diagnostics," *Proc. Inst. Mech. Eng., Part G: J. Aerosp. Eng.*, **222**(3), pp. 307–318.
- [3] Li, Y. G., 2003, "A Gas Turbine Diagnostic Approach With Transient Measurements," *Proc. Inst. Mech. Eng., Part A*, **217**(2), pp. 169–177.
- [4] Merrington, G., Kwon, O., Goodwin, G., and Carlsson, B., 1991, "Fault Detection and Diagnosis in Gas Turbines," *ASME J. Eng. Gas Turbines Power*, **113**, pp. 276–282.
- [5] Wang, X., McDowell, N., Kruger, U., McCullough, G., and Irwin, G. W., 2008, "Semi-Physical Neural Network Model in Detecting Engine Transient Faults Using the Local Approach," Proceedings of the 17th World Congress of the International Federation of Automatic Control (IFAC'08), July 6–11.
- [6] Surender, V. P., and Ganguli, R., 2005, "Adaptive Myriad Filter for Improved Gas Turbine Condition Monitoring Using Transient Data," *ASME J. Eng. Gas Turbines Power*, **127**, pp. 329–339.
- [7] Menon, S., Uluyol, O., Kim, K., and Nwadiogbu, E. O., 2003, "Incipient Fault Detection and Diagnosis in Turbine Engines Using Hidden Markov Models," *ASME Turbo Expo*, Paper No. GT2003-38589, pp. 493–500.
- [8] Ray, A., 2004, "Symbolic Dynamic Analysis of Complex Systems for Anomaly Detection," *Signal Process.*, **84**(7), pp. 1115–1130.
- [9] Rao, C., Ray, A., Sarkar, S., and Yasar, M., 2009, "Review and Comparative Evaluation of Symbolic Dynamic Filtering for Detection of Anomaly Patterns," *Signal, Image Video Process.*, **3**(2), pp. 101–114.
- [10] Gupta, S., Ray, A., Sarkar, S., and Yasar, M., 2008, "Fault Detection and Isolation in Aircraft Gas Turbine Engines. Part 1: Underlying Concept," *Proc. Inst. Mech. Eng., Part G: J. Aerosp. Eng.*, **222**(3), pp. 307–318.
- [11] Sarkar, S., Yasar, M., Gupta, S., Ray, A., and Mukherjee, K., 2008, "Fault Detection and Isolation in Aircraft Gas Turbine Engines. Part 2: Validation on a Simulation Test Bed," *Proc. Inst. Mech. Eng., Part G: J. Aerosp. Eng.*, **222**(3), pp. 319–330.
- [12] Sarkar, S., Rao, C., and Ray, A., 2009, "Statistical Estimation of Multiple Faults in Aircraft Gas Turbine Engines," *Proc. Inst. Mech. Eng., Part G: J. Aerosp. Eng.*, **223**(4), pp. 415–424.
- [13] Sarkar, S., Jin, X., and Ray, A., 2011, "Data-Driven Fault Detection in Aircraft Engines With Noisy Sensor Measurements," *ASME J. Eng. Gas Turbines Power*, **133**(8), p. 081602.
- [14] Armstrong, J., 2009, "User's Guide for the Transient Test Case Generator," NASA GRC Internal Report.
- [15] Frederick, D. K., DeCastro, J. A., and Litt, J. S., 2007, "User's Guide for the Commercial Modular Aero-Propulsion System Simulation (C-MAPSS)," Report No. NASA/TM2007-215026.
- [16] Rajagopalan, V., and Ray, A., 2006, "Symbolic Time Series Analysis via Wavelet-Based Partitioning," *Signal Process.*, **86**(11), pp. 3309–3320.
- [17] Wen, Y., and Ray, A., 2012, "Vector Space Formulation of Probabilistic Finite State Automata," *J. Comput. Syst. Sci.*, **78**, pp. 1127–1141.
- [18] Adenis, P., Wen, Y., and Ray, A., 2012, "An Inner Product Space on Irreducible and Synchronizable Probabilistic Finite State Automata," *Math. Control, Signals, Syst.*, **23**(1), pp. 281–310.
- [19] Ferguson, T., 1973, "Bayesian Analysis of Some Nonparametric Problems," *Ann. Stat.*, **1**(2), pp. 209–230.
- [20] Sethuraman, J., 1994, "A Constructive Definition of Dirichlet Priors," *Stat. Sin.*, **4**, pp. 639–650.
- [21] Wilks, S., 1963, *Mathematical Statistics*, John Wiley, New York.
- [22] Pathria, R., 1996, *Statistical Mechanics*, 2nd ed., Butterworth-Heinemann, Oxford, UK.
- [23] Kobayashi, T., and Simon, D., 2005, "Hybrid Neural-Network Genetic-Algorithm Technique for Aircraft Engine Performance Diagnostics," *J. Propulsion Power*, **21**(4), pp. 751–758.
- [24] Poor, V., 1988, *An Introduction to Signal Detection and Estimation*, 2nd ed., Springer-Verlag, New York.
- [25] Sarkar, S., Singh, D. S., Srivastav, A., and Ray, A., 2011, "Semantic Sensor Fusion for Fault Diagnosis in Aircraft Gas Turbine Engines," Proceedings of American Control Conference, San Francisco, CA.