Statistical estimation of multiple faults in aircraft gas turbine engines

S Sarkar, C Rao, and **A Ray**[*]
Department of Mechanical Engineering, The Pennsylvania State University, University Park, Pennsylvania, USA

Abstract: This article presents estimation of multiple faults in aircraft gas-turbine engines, based on a statistical pattern recognition tool called symbolic dynamic filtering. The underlying concept is formulated by statistical analysis of evidences to estimate anomalies (i.e. deviations from the nominal values) in multiple critical parameters of the engine system; it also presents a framework for sensor information fusion. The fault estimation algorithm is validated on a numerical simulation test-bed that is built upon the NASA C-MAPSS model of a generic commercial aircraft engine.

Keywords: aircraft propulsion, gas turbine engines, multiple fault estimation, statistical pattern recognition

1 INTRODUCTION

Aircraft propulsion system health monitoring is one of the key issues regarding aviation safety. Current state of the art of health monitoring involves a regular and fixed schedule of inspection and maintenance, which are essentially conservative in nature and hence may not be cost effective. On the other hand, in unusual circumstances, the normal ground inspection schedule may not be able to detect aggravation of hidden faults, which may result in a permanent damage of the engine or a potentially catastrophic accident. Hence, there is a need for new technologies of health monitoring for aircraft gas turbine engines, which can be pursued either onboard during a flight or on the ground but without the need for installation of additional sensors and/or mounting the engine on a maintenance test facility.

Health monitoring of aircraft engines is traditionally conducted in three major steps, namely, detection, diagnosis, and prognosis. Detection involves identification of an existing fault in the system. Upon detection of a fault, the role of diagnosis is to isolate the fault type and location and to quantify the fault in a statistical sense. Given possible statistics of future operating conditions and an expected component deterioration profile, the role of prognosis is to estimate the remaining life of the engine from the information generated in the diagnosis step. Nevertheless, an aircraft engine is a complex large-scale dynamical system whose subsystems are interconnected physically as well as through feedback control loops. It is a challenging task to detect, isolate, and estimate the severity of the fault(s), which has been addressed by many investigators over the last several decades.

Hoffman and van der Merwe [1] have shown that traditional frequency-based methods may not be readily applicable to estimate the evolution of multiple faults in gas turbine engines; fault dictionaries have been used to alleviate this problem, but they are often infeasible to store in onboard applications because of their large sizes [2]. Recent literature has reported several other methods, such as those based on joint state estimation [3], parity equations [4], and similarity measures [5], for multiple parameter estimation.

Several model-based and data-driven techniques have been reported in literature for fault detection, diagnosis, and prognosis, which include statistical linearization [6], Kalman filtering [7], unscented Kalman filtering [8, 9], particle filtering (PF) [10], Markov chain Monte Carlo [11], Bayesian networks [12], neural networks [1], maximum likelihood estimation [13], wavelet-based tools [14], and genetic algorithms [15].

[*]*Corresponding author: Department of Mechanical Engineering, Pennsylvania State University, 329 Reber Building, University Park, PA 16802, USA.*
email: axr2@psu.edu

In general, faults in an aircraft engine are classified into three major categories.

1. Loss of efficiency in subsystems (e.g. fan, compressor, and turbines).
2. Sensor degradation (e.g. drift and fluctuations).
3. Actuator faults (e.g. sluggish response and excessive dead band and hysteresis).

Specifically, this article addresses estimation of those faults that cause efficiency degradation in engine components.

For data-driven estimation of multiple faults, every representative fault condition that is manifested by a combination of different types of faults could be unique in the multi-dimensional parameter space. Hence, individual training patterns need to be stored in a database. This requires intelligent information compression for low memory requirement and low computational complexity. On the other hand, in a complex system such as an aircraft gas-turbine engine, the patterns generated from a single sensor may not carry sufficient information to identify multiple parameters/faults because different combinations of component faults may generate similar signatures in a particular sensor observation.

Anomaly (i.e. deviation from the nominal condition) detection algorithms, built upon on a statistical pattern recognition tool called symbolic dynamic filtering (SDF) [16], have been developed and experimentally validated for real-time execution in different applications (e.g. electronic circuits [17] and fatigue damage monitoring in polycrystalline alloys [18]). For such applications, a framework for probabilistic identification of a single system parameter is reported in reference [17]. The anomaly detection algorithms, constructed in the SDF setting, have shown superior performance in terms of early detection of anomalies and robustness to measurement noise by comparison with other existing techniques such as principal component analysis, artificial neural networks, and Bayesian techniques [19]. Very recently, the theory of SDF-based single-parameter estimation has been extended to multiple parameters along with experimental validation in nonlinear electronic systems [20]. A key feature of this extension is fusion of statistical information to obtain a probabilistic estimate of multiple parameters of the underlying system.

In a two-part paper [21, 22], an SDF-based algorithm for detection and isolation of faults in aircraft engine subsystems has been reported. As described earlier, the next step is to estimate the severity of detected fault(s) for diagnosis. In this context, quantification of simultaneously evolving faults in critical subsystems pose the challenging task of multiple parameter estimation in a dynamical system. This issue is addressed in the present article that develops and validates a statistical methodology, based on SDF, for estimation of multiple faults in a gas turbine engine. The major contributions of this article beyond our recent work [21, 22] in the field of engine health monitoring are listed below.

1. Estimation of (simultaneously occurring) multiple faults in aircraft gas-turbine engines.
2. Formulation of a framework for sensor data fusion as needed for estimation of multiple faults.
3. Data compression as pattern vectors of low-dimension for feature-level sensor fusion as needed for onboard vehicle health monitoring and resilient control.

The article is organized in four sections including the present section. Section 2 describes the multiple fault estimation methodology along with a framework of sensor information fusion in aircraft gas-turbine engines. Section 3 validates the concept of multiple fault estimation on the NASA simulation test-bed of commercial aircraft engines (C-MAPSS) [23]. The article is summarized and concluded in section 4 along with recommendations for future work.

2 MULTIPLE FAULT ESTIMATION METHODOLOGY

This section describes a statistical methodology for multiple fault estimation in aircraft gas-turbine engines along with a sensor fusion framework.

2.1 Problem statement

In general, the fault scenarios in major engine components can be categorized into three different types based on their mode of occurrence. These types are: (a) gradual deterioration, (b) intermittent faults, and (c) abrupt large faults. However, no matter what type of fault occurs in a particular component, the problem can be reduced to a parameter identification problem from the point of view of fault estimation as presented below.

Let $\mathcal{S}$ denote a collection of (finitely many) data points in the n-dimensional parameter space, where the positive integer n is the number of parameters that are to be estimated, i.e. $\mathcal{S} = \{s^0, s^1, \ldots, s^{|\mathcal{S}|-1}\}$, on which the training process is executed. In the context of gas-turbine engines, s^k signifies a particular faulty condition in the set of fault conditions $\mathcal{S}$ under consideration. Let s^0 denote the nominal condition of the engine and $\mathcal{Y}$ be the set of sensors for the engine system consisting of sensors y_j for $j = 1, 2, \ldots, |\mathcal{Y}|$. Let Ω be the convex hull of $\mathcal{S}$, which represents the range over which the parameters take values. It is noted that Ω is a convex and compact subset of the separable space $\mathbb{R}^n$. The problem at hand is to statistically estimate fault condition $s \in \Omega$, given an experimental data

set Υ, i.e. to identify the conditional probability density $f(s|\Upsilon)$. It is noted that $s \in \Omega$ may not be one of the points in set $\mathcal{S}$.

The multiple-fault estimation procedure is divided into two steps, which are: (a) forward problem/training and (b) inverse problem/testing, as shown in Fig. 1. The following sections 2.2 and 2.3 describe the two steps in detail.

2.2 Forward problem/training

In the forward problem, a database of patterns is created at parameter values, $s^k, \forall k = 0, 1, \ldots, (|\mathcal{S}| - 1)$, by collecting time-series data from sensors $y_j \in \mathcal{Y}$, as shown in Fig. 1. Generation of statistical patterns from time series data is posed as a two-scale problem [16, 21]. The *fast scale* is related to the response time of the process dynamics, over the span of which the process is assumed to be quasi-stationary. The *slow scale* is related to the time span over which deviations (e.g. parametric or non-parametric changes) may occur and exhibit non-stationary dynamics. In the present context, time-series data are collected with the system being quasi-stationary at a particular slow-scale epoch s^k. The procedural steps of the forward problem are presented below.

1. *Time series data acquisition on the fast scale from the available sensors*: time series data sets from each sensor $y_j \in \mathcal{Y}$ are collected for each epoch $s^k \in \mathcal{S}$.
2. *Wavelet/Hilbert transform preprocessing of the time-series data*: the wavelet or Hilbert transforms largely alleviate the difficulties of phase-space partitioning and are particularly effective with noisy data from high-dimensional dynamical systems [24, 25].
3. *Maximum entropy partitioning of the transformed space at the reference condition of epoch s^0*: this step enables transformation of the preprocessed time series data from the continuous domain to the symbol domain [16] by partitioning the transformed phase space, where the data set from each sensor $y_j, j = 1, \ldots, |\mathcal{Y}|$, has its own alphabet; for each sensor, a specific symbol is assigned to each partition segment from the respective alphabet. Maximum entropy partitioning [24, 25] is constructed separately for different sensor data sets at epoch s^0. These partitions are kept invariant for

Fig. 1 Outline of the fault estimation procedure

analysis at subsequent epochs $s^1, s^2, \ldots, s^{|\mathcal{S}|-1}$ of respective sensor data.

4. *Construction of a probabilistic finite state automaton at the reference condition s^0 and computation of state probabilities*: probabilistic finite state automaton (PFSA) are constructed for every sensor data at epoch s^0 and their structures remain invariant for subsequent epochs of each sensor data. Let the N_j be the number of states in the PFSA corresponding to the sensor $y_j, j = 1, \ldots, |\mathcal{Y}|$. The sum of the probabilities of all states is equal to unity, i.e. $\sum_{i=1}^{N_j} p_i^j(s^k) = 1 \; \forall j \in \{1, \ldots, |\mathcal{Y}|\} \forall k \in \{0, \ldots, |\mathcal{S}| - 1\}$, where $p_i^j(s^k)$ denotes the probability of the ith state of the PFSA constructed fron time series of jth sensor at epoch s^k; at most $N_j - 1$, out of the N_j elements of the state probability vector can be independent. Therefore, the pattern for each sensor labeled by $j = 1, \ldots, |\mathcal{Y}|$ is represented by a $(N_j - 1)$-dimensional row vector $\boldsymbol{p}^j \triangleq [p_1^j \cdots p_{N_j-1}^j] \; \forall j \in \{1, \ldots, |\mathcal{Y}|\}$; this notation holds for all epochs $s^k \forall k \in \{0, \ldots, |\mathcal{S}| - 1\}$.

5. *Construction of the pattern database*: a reference pattern array $\mathcal{P}(s^0)$ is constructed by vertical stacking of the reference row vectors, $\boldsymbol{p}^j(s^0)$, $j \in \{1, \ldots, |\mathcal{Y}|\}$, as shown below

$$\mathcal{P}(s^0) \triangleq \begin{Bmatrix} \boldsymbol{p}^1(s^0) \\ \boldsymbol{p}^2(s^0) \\ \cdots \\ \cdots \\ \boldsymbol{p}^{|\mathcal{Y}|}(s^0) \end{Bmatrix} = \begin{Bmatrix} p_1^1(s^0) \cdots p_{N_1-1}^1(s^0) \\ p_1^2(s^0) \cdots p_{N_2-1}^2(s^0) \\ \cdots \\ \cdots \\ p_1^{|\mathcal{Y}|}(s^0) \cdots p_{N_{|\mathcal{Y}|}-1}^{|\mathcal{Y}|}(s^0) \end{Bmatrix}$$

Note that the individual rows in the array $\mathcal{P}$ may have different lengths because the PFSA corresponding to different sensors may have different state cardinalities; hence, $\mathcal{P}$ should not be viewed as a matrix but it is a two-dimensional array of positive fractions, where the total number of elements is $(N_1 + \cdots + N_{|\mathcal{Y}|} - |\mathcal{Y}|)$. Similarly, $\mathcal{P}(s^1), \mathcal{P}(s^2), \ldots, \mathcal{P}(s^{|\mathcal{S}|-1})$ are computed at epochs $s^1, s^2, \ldots, s^{|\mathcal{S}|-1}$ from the respective patterns. Note that the structure of the PFSA at all epochs for a particular sensor is identical while the pattern arrays $\mathcal{P}(s^k)$ are possibly different at different $s^k \in \mathcal{S}$ because of parametric or non-parametric changes in the process.

6. *Computation of pattern statistics*: different units of identically manufactured engines are different in behaviour or performance; this inevitable uncertainty is modelled as the process noise. Therefore, several runs are performed for each fault condition, with a certain value of process noise along with an *a priori* determined sensor noise (e.g. calculated from instrumentation manufacturer's specifications) to obtain the pattern vector statistics.
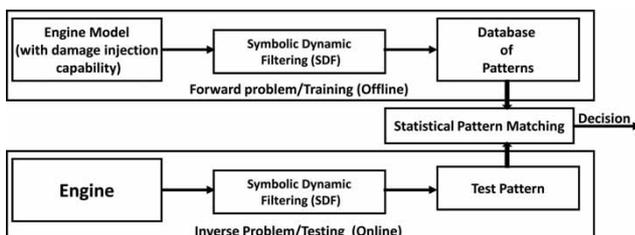
Let the pattern array $\mathcal{P}(\boldsymbol{s}^k)$ be modelled as a random array $\mathcal{Q}(\boldsymbol{s}^k)$, whose elements are $q_i^l(\boldsymbol{s}^k)$ that is constructed from the ensemble of realizations $p_i^l(\boldsymbol{s}^k)$. Considering up to second-order statistics, elements of the random array $\mathcal{Q}(\boldsymbol{s}^k)$ are modelled to have multivariate structures from the perspectives of state machine construction in the SDF setting, as explained later in Remark 1. Thus, for each epoch $\boldsymbol{s}^k$, a mean pattern vector $\boldsymbol{\mu}(\boldsymbol{s}^k)$ and a corresponding covariance matrix $\boldsymbol{\Gamma}(\boldsymbol{s}^k)$ of the pattern are calculated from the elements of $\mathcal{Q}(\boldsymbol{s}^k)$. An element of $\boldsymbol{\mu}(\boldsymbol{s}^k)$ is expressed as $m_i^l(\boldsymbol{s}^k)$, $\forall l \in \{1, 2, \ldots, |\mathcal{Y}|\}$ and $\forall i \in \{1, 2, \ldots, N_l - 1\}$, which signifies the mean values of $p_i^l(\boldsymbol{s}^k)$ generated from the data sets of different runs. Similarly, an element of the covariance matrix $\boldsymbol{\Gamma}(\boldsymbol{s}^k)$ is expressed as $\gamma_{ij}^{l\ell}(\boldsymbol{s}^k)$, $\forall l, \ell \in \{1, 2, \ldots, |\mathcal{Y}|\}$ and $\forall i \in \{1, 2, \ldots, N_l - 1\}$, and $\forall j \in \{1, 2, \ldots, N_\ell - 1\}$, which signifies the value of cross-covariance between $p_i^l(\boldsymbol{s}^k)$ and $p_j^\ell(\boldsymbol{s}^k)$, which is also generated from the data sets of different runs. Note that, for $l = \ell$, the covariance matrix terms yield correlation among the states i and j of the PFSA generated from the same sensor data and, for $l \neq \ell$, the covariance matrix terms yield correlation among the states i and j of different *PFSA* corresponding to different sensors.

For the purpose of book-keeping in statistical calculations, each of the (two-dimensional) arrays $\mathcal{P}(\boldsymbol{s}^k)$ is rearranged as a single row vector $\boldsymbol{p}(\boldsymbol{s}^k)$ by horizontally concatenating the row vectors $\boldsymbol{p}^j(\boldsymbol{s}^k)$, $j \in \{1, \ldots, |\mathcal{Y}|\}$, i.e. the random pattern array $\mathcal{Q}(\boldsymbol{s}^k)$ is rearranged as the random pattern vector $\boldsymbol{q}(\boldsymbol{s}^k)$. The mean pattern vector $\boldsymbol{\mu}(\boldsymbol{s}^k)$ and covariance matrix $\boldsymbol{\Gamma}(\boldsymbol{s}^k)$ are constructed correspondingly. The covariance matrix $\boldsymbol{\Gamma}(\boldsymbol{s}^k)$ is comprised of several blocks of elements. The square diagonal blocks correspond to the covariance among states of same sensor data, where as the off-diagonal possibly non-square (due to possible different alphabet size for different sensor data) blocks correspond to the covariance among states of different sensor data.

Remark 1

The underlying dynamical system is modelled as an irreducible Markov process through SDF, where the state probability vector is the sum-normalized eigenvector of the state transition matrix corresponding to the unique unity eigenvalue. Hence, no element in the state probability vector is either 0 or equal to 1. However, due to process noise and sensor noise, the random vector $\boldsymbol{q}(\boldsymbol{s}^k)$ fluctuates around its mean $\boldsymbol{\mu}(\boldsymbol{s}^k)$. Analysing the experimental data, the terms of the covariance matrices of the random vectors $\boldsymbol{q}(\boldsymbol{s}^k)$ were found to be very small compared to the mean. Therefore, a parametric or non-parametric two-sided uni-modal distribution should be adequate to model the random vector $\boldsymbol{q}(\boldsymbol{s}^k)$. The choice of Gaussian

distribution for $\boldsymbol{q}$ would facilitate estimation of the statistical parameters and involve only second-order statistics. Also, the elements of $\boldsymbol{q}(\boldsymbol{s}^k)$ have to be positive, which is made possible by truncating the far end of the Gaussian distribution tail on the left side. The goodness of fit of the distribution as Gaussian still remains valid at a very high significance level. For a particular sensor y_j summation of the elements $q_i^j(\boldsymbol{s}^k)$, $\forall i \in \{1, 2, \ldots, N_j\}$ has to be unity, which is achieved by sumnormalization.

The (jointly Gaussian) conditional probability distribution of a random pattern vector $\boldsymbol{q}$ is given as

$$f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\boldsymbol{s}^k) = \frac{1}{(2\pi)^{N/2}|\boldsymbol{\Gamma}(\boldsymbol{s}^k)|^{1/2}} \exp\left\{-\frac{1}{2}[\boldsymbol{p} - \boldsymbol{\mu}(\boldsymbol{s}^k)]\right.$$
$$\left. \times [\boldsymbol{\Gamma}(\boldsymbol{s}^k)]^{-1}[\boldsymbol{p} - \boldsymbol{\mu}(\boldsymbol{s}^k)]^T\right\} \qquad (1)$$

where $N = N_1 + \cdots + N_{|\mathcal{Y}|} - |\mathcal{Y}|$.

2.3   Inverse problem/testing

The objective here is to identify the probabilistic location of the fault in the multi-dimensional parameter space, i.e. identification of the unknown parameter vector $\boldsymbol{s} \in \Omega$; however, it is possible that $\boldsymbol{s} \notin \mathcal{S}$. Therefore, for a particular test case, time series data are collected from different sensors. The data are analysed using the same symbolic dynamic filter constructed in the forward problem/training (see section 2.2), and the resulting row vector $\boldsymbol{p}$ is a realization of a random pattern vector $\boldsymbol{q}$. The density function $f_{\Omega|\boldsymbol{q}}(\boldsymbol{s}|\boldsymbol{p})$ is obtained as

$$f_{\Omega|\boldsymbol{q}}(\boldsymbol{s}|\boldsymbol{p}) = \frac{f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\boldsymbol{s})f_\Omega(\boldsymbol{s})}{f_{\boldsymbol{q}}(\boldsymbol{p})}$$
$$= \frac{f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\boldsymbol{s})f_\Omega(\boldsymbol{s})}{\int_\Omega f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\tilde{\boldsymbol{s}})f_\Omega(\tilde{\boldsymbol{s}})\mathrm{d}\tilde{\boldsymbol{s}}} \qquad (2)$$

In the absence of *a priori* information, an assumption is made that all operating conditions are equally likely, i.e. $f_\Omega(\boldsymbol{s}) = f_\Omega(\tilde{\boldsymbol{s}})\ \forall \tilde{\boldsymbol{s}},\ \tilde{\boldsymbol{s}} \in \Omega$. With this assumption of uniform probability, equation (2) reduces to

$$f_{\Omega|\boldsymbol{q}}(\boldsymbol{s}|\boldsymbol{p}) = \frac{f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\boldsymbol{s})}{\int_\Omega f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\tilde{\boldsymbol{s}})\mathrm{d}\tilde{\boldsymbol{s}}} \qquad (3)$$

It is noted that accuracy of the above distribution would be improved if the actual prior mapping, i.e. $f_\Omega(\boldsymbol{s})$ is known. The integral in the denominator of equation (3) is approximated by a Reimann sum as

$$f_{\Omega|\boldsymbol{q}}(\boldsymbol{s}|\boldsymbol{p}) \approx \kappa \frac{f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\boldsymbol{s})}{\sum_\mathcal{S} f_{\boldsymbol{q}|\Omega}(\boldsymbol{p}|\tilde{\boldsymbol{s}})} \qquad (4)$$

where κ is a constant. This approximation converges to the exact solution as the training set $\mathcal{S}$ approaches a countable dense subset of $\Omega \subset \mathbb{R}^n$.

The density function in equation (4) is now sampled at the points s^k in the training set $\mathcal{S}$ and the following sampled density is constructed as to yield

$$f_{\Omega|q}(\boldsymbol{s}|\boldsymbol{p})\big|_{s=s^k} \approx \kappa \frac{f_{q|\Omega}(\boldsymbol{p}|\boldsymbol{s}^k)}{\sum_{\tilde{s}\in\mathcal{S}} f_{q|\Omega}(\boldsymbol{p}|\tilde{\boldsymbol{s}})} \; \forall \boldsymbol{s}^k \in \mathcal{S} \tag{5}$$

The density functions in the numerator and denominator of equation (5) are obtained from equation (1), which were determined in the training phase. It is noted that the nature of the density function $f_{\Omega|q}(\boldsymbol{s}^k|\boldsymbol{p})$ does not depend on the constant κ.

The probability mass functions are obtained by evaluating the probability density function in equation (5) at points $\boldsymbol{s}^k \in \mathcal{S}$.

$$\begin{aligned} P(\boldsymbol{s}^k|\boldsymbol{p}) &\triangleq \frac{f_{\Omega|q}(\boldsymbol{s}^k|\boldsymbol{p})}{\sum_{j=1}^{|\mathcal{S}|} f_{\Omega|q}(\boldsymbol{s}^j|\boldsymbol{p})} \\ &\approx \frac{f_{q|\Omega}(\boldsymbol{p}|\boldsymbol{s}^k)}{\sum_{j=1}^{|\mathcal{S}|} f_{q|\Omega}(\boldsymbol{p}|\boldsymbol{s}^j)} \end{aligned} \tag{6}$$

which is expressed in terms of equation (1) as

$$P(\boldsymbol{s}^k|\boldsymbol{p}) \approx \frac{[1/((2\pi)^{N/2}|\Gamma(\boldsymbol{s}^k)|^{1/2})]\exp(X(\boldsymbol{s}^k))}{\sum_{l=1}^{|\mathcal{S}|}[1/((2\pi)^{N/2}|\Gamma(\boldsymbol{s}^l)|^{1/2})]\exp(X(\boldsymbol{s}^l))} \tag{7}$$

where $X(\bullet) = -(\frac{1}{2})[\boldsymbol{p}-\boldsymbol{\mu}(\bullet)][\Gamma(\bullet)]^{-1}[\boldsymbol{p}-\boldsymbol{\mu}(\bullet)]^{\mathrm{T}}$.

The above equation signifies a statistical pattern matching by calculating the Mahalnobis distance [26] between the test and the training patterns; therefore, smaller the Mahalnobis distance, better is the match between these two patterns.

It has been observed from experimental data that fluctuations of the pattern vectors are very weakly correlated among different symbols and different sensors. Therefore, the jointly Gaussian distribution of all $f_{q|\Omega}(\boldsymbol{p}|\boldsymbol{s}^k)$'s can be reduced to the product of individual Gaussian distributions $f_{q|\Omega}(p_i^j|\boldsymbol{s}^k)$ of different symbols $\forall j \in \{1, 2, \ldots, |\mathcal{Y}|\}$ and $\forall i \in \{1, 2, \ldots, N_j - 1\}$. Therefore, instead of using the multi-variate jointly Gaussian distribution, univariate Gaussian distribution is used for each symbol, (the variance being the corresponding diagonal element of the covariance matrix) to calculate $P(\boldsymbol{s}^k|\boldsymbol{p})$. Thus, equation (7) is expressed as

$$\forall j \in \{1, 2, \ldots, |\mathcal{Y}|\} \text{ and } \forall i \in \{1, 2, \ldots, N_j - 1\}$$

$$P(\boldsymbol{s}^k|\boldsymbol{p})$$

$$\approx \frac{\prod_j \prod_i \{1/[(2\pi)^{1/2}(\gamma_{ii}^{jj}(\boldsymbol{s}^k))^{1/2}]\}\exp[X_i^j(\boldsymbol{s}^k)]}{\sum_{l=1}^{|\mathcal{S}|}\prod_j \prod_i \{1/[(2\pi)^{1/2}(\gamma_{ii}^{jj}(\boldsymbol{s}^l))^{1/2}]\}\exp[X_i^j(\boldsymbol{s}^l)]} \tag{8}$$

where $X_i^j(\bullet) \triangleq \{-(\frac{1}{2})[p_i^j - m_i^j(\bullet)][\gamma_{ii}^{jj}(\bullet)]^{-1}[p_i^j - m_i^j(\bullet)]\}$.

Once the probability mass function $P(\boldsymbol{s}^k|\boldsymbol{p})$ is obtained, there can be different estimates $\hat{\boldsymbol{s}} \in \Omega$

depending upon the cost function of estimation. For example, the median of the distribution yields the estimated value by minimizing the root mean square value of the deviations. Again, most likely parameter value can be obtained from the mode of the distribution. In this article, estimated mean is considered that minimizes the average of the square of the absolute deviations around the estimated point. Estimated mean $\hat{\boldsymbol{s}}$ and estimated covariance matrix $\hat{\mathbf{C}}_s$ of the parameter (column) vector $\boldsymbol{s}$ are obtained directly from $P(\boldsymbol{s}^k|\boldsymbol{p})$ as

$$\hat{\boldsymbol{s}}(\boldsymbol{p}) \triangleq \sum_{k=1}^{|\mathcal{S}|} \boldsymbol{s}^k \, P(\boldsymbol{s}^k|\boldsymbol{p}) \tag{9}$$

$$\hat{\mathbf{C}}_s(\boldsymbol{p}) \triangleq \sum_{k=1}^{|\mathcal{S}|} \left[\boldsymbol{s}^k - \hat{\boldsymbol{s}}(\boldsymbol{p})\right] P(\boldsymbol{s}^k|\boldsymbol{p}) \left[\boldsymbol{s}^k - \hat{\boldsymbol{s}}(\boldsymbol{p})\right]^{\mathrm{T}} \tag{10}$$

Since the statistical information is available in the form of probability mass functions, the third and higher moments of the parameter vector can be estimated in a similar way; however, third and higher moments are redundant because the inherent distribution is assumed to have a Gaussian structure that carries full statistical information in the first two moments.

2.4 Discussion on sensor fusion

The above formulation of the inverse problem uses information from all the sensors y_j for $j = 1, 2, \ldots, |\mathcal{Y}|$, i.e. the information from all sensors are fused together to estimate the fault level in the engine test case. However, this fusion technique allows the user to choose the number and the combination of sensors to be used for the multiple fault estimation. For example, let us assume that the user just wants to use only one sensor, say, sensor y_1. Then, only blocks pertaining to y_1 are to be selected from the forward problem pattern database. Thus, elements $m_i^j(\boldsymbol{s}^k)$ for $j = 1$ and $\forall i \in \{1, 2, \ldots, N_1 - 1\}$ are selected from $\boldsymbol{\mu}(\boldsymbol{s}^k)$ and elements $\gamma_{xy}^{mn}(\boldsymbol{s}^k)$ for $m, n = 1$ and $\forall x, y \in \{1, 2, \ldots, N_1 - 1\}$ are selected from $\Gamma(\boldsymbol{s}^k) \; \forall \boldsymbol{s}^k \in \mathcal{S}$. In another example, both sensors s_1 and s_3 are to be used. Then, elements $m_i^j(\boldsymbol{s}^k)$, $\forall j \in \{1, 3\}$, and $\forall i \in \{1, 2, \ldots, N_j - 1\}$ are selected from $\boldsymbol{\mu}(\boldsymbol{s}^k)$ and elements $\gamma_{xy}^{mn}(\boldsymbol{s}^k)$, $\forall m, n \in \{1, 3\}$, and $\forall x \in \{1, 2, \ldots, N_m - 1\}$, and $\forall y \in \{1, 2, \ldots, N_n - 1\}$ are selected from $\Gamma(\boldsymbol{s}^k) \; \forall \boldsymbol{s}^k \in \mathcal{S}$. It follows from the above two examples that the elements of the test patterns need to be selected corresponding to the sensors under consideration.

Remark 2

The current framework attempts to fuse information from different sensors at *feature level* as opposed to the frameworks of *data level* or *decision level* fusion. The advantages of the present sensor information fusion framework are delineated below.

1. Data level fusion techniques often encounter scaling problem while fusing information from sensors of different modality. However, the present technique fuses the probability vector patterns, which does not have any scaling issue.
2. Decision level fusion generally provides too coarse diagnosis of faults and also requires in depth understanding of the physical system.
3. Multi-dimensionality of the parameter space has been taken care of without the use of product automata which leads to state explosion.

3 VALIDATION ON THE C-MAPSS TEST-BED

The C-MAPSS simulation test-bed, developed at NASA, is built upon the model of a commercial-scale two-spool turbofan engine and its control system. While the details of the model are available in reference [23], a brief outline of C-MAPSS is provided here for completeness of the article. The engine under consideration produces a thrust of ∼400 000 N and is designed for operation at (a) altitudes from sea level up to 12 200 m, (b) Mach numbers from 0 to 0.90, and (c) sea-level temperatures from approximately −50 °C to 50 °C. The throttle resolving angle (TRA) can be set to any value in the range between 0° (minimum power) and 100° (maximum power).

As seen in Figs 2(a) and (b), the simulation test-bed of the gas turbine engine system consists of high-pressure compressor (HPC), combustor, and high-pressure turbine (HPT), which form the core of the engine model; this subsystem is also referred to as the gas generator. In the turbofan engine, the engine core is surrounded by the fan and low-pressure compressure (LPC) in the front and an additional low-pressure turbine (LPT) at the rear; and fan, LPC and LPT are mechanically connected by an additional shaft. The fan shaft passes through the core shaft and because of this type of arrangement, the engine is called a two spool engine. In contrast to gas turbine engines for military aircraft [21, 22], a relatively small part of the incoming air at the engine inlet passes through the fan and continues on into the core compressor and then into the combustor, where it is mixed with fuel and combustion occurs; therefore, this type of engine is known as a high-bypass engine. The hot exhaust gas, called the core airflow, passes through the core and LPT and then exits through the nozzle; and the rest of the incoming air passes through the fan and bypasses, or flows around the engine. A gain-scheduled control system is incorporated in the engine system, which consists of (a) a fan-speed controller for a specified throttle-resolver angle (TRA), (b) three high-limit regulators that prevent the engine from exceeding its design limits for core-spool speed, engine-pressure ratio, and HPT exit temperature, (c) the fourth limit regulator
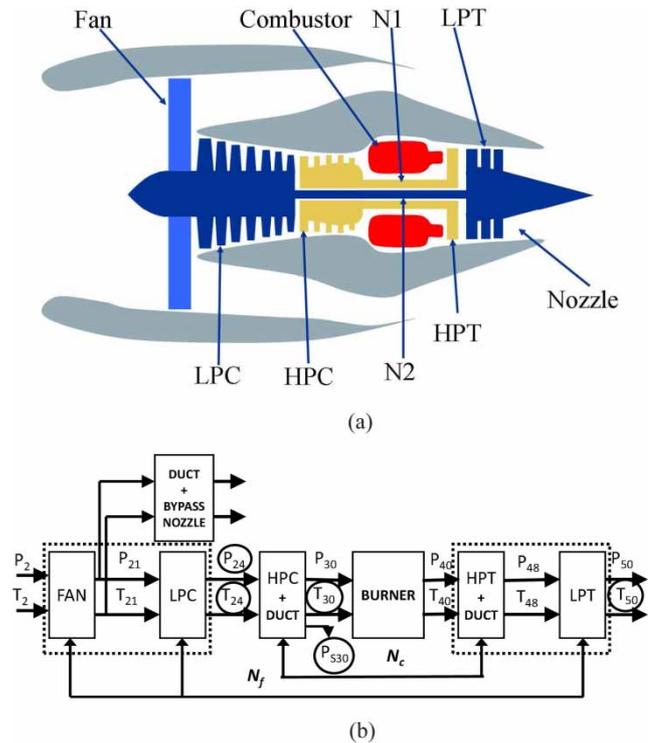

Fig. 2 C-MAPSS engine simulation test-bed: (a) gas turbine engine schematic and (b) gas turbine engine model configuration

that attempts to prevent the static pressure at the HPC exit from dropping too low, (d) acceleration and deceleration limiters for the core-spool speed, and (e) a comprehensive logic structure that integrates these control-system components in a manner similar to that used in real engine controllers such that integrator-windup problems are avoided. To achieve fast execution of simulation runs, the sensors and actuators are approximated to have instantaneous response, no computational time delays, and no drift and or bias. Given the inputs of TRA, altitude (*a*) and Mach number (*M*), the interactively controlled component models at the simulation test-bed compute non-linear dynamics of real-time turbofan engine operation. Both steady-state and transient operations are simulated in the continuous-time setting. Performance maps are used to provide steady-state representations of the engine's rotating components. Fluid momentum in the bypass duct and the augmentor, mass and energy storage within control volumes, and rotor inertias are also included to model transient operations. The entire test-bed code is written on Matlab and Simulink platform.

As indicated earlier, this article addresses estimation of those faults that cause efficiency degradation in engine components. In the current configuration of the C-MAPSS simulation test-bed, there are 13 health parameter inputs, namely, efficiency health parameters (ψ), flow health parameters (ζ), and pressure ratio

modifiers, that simulates the effects of faults and/or degradation in the engine components. Ten, out of these 13 health parameters, are selected to modify efficiency (η) and flow (ϕ) which are defined [27] as:

(a) $\eta \triangleq$ the ratio of actual enthalpy change and ideal enthalpy change;
(b) $\phi \triangleq$ the ratio of tip rotor velocity and axial fluid flow velocity.

For the engine's five rotating components (i.e. Fan, LPC, HPC, HPT, and LPT), the ten health parameters are: (a) fan (ψ_F, ζ_F), (b) LPC (ψ_{LPC}, ζ_{LPC}), (c) HPC (ψ_{HPC}, ζ_{HPC}), (d) HPT (ψ_{HPT}, ζ_{HPT}), and (e) LPT (ψ_{LPT}, ζ_{LPT}). Table 1 lists the (commercially available) sensors and their locations (see Fig. 2(b)) that have been used for multiple fault estimation in C-MAPSS engine test-bed.

3.1 Results and discussions

Time series data have been collected for different sensors under persistent excitation of TRA inputs that have truncated triangular profiles with the mean value of 80°, fluctuations within ±2° and frequency of 0.056 Hz as shown in Fig. 3. The ambient conditions are chosen to be at the sea level (i.e. altitude $a = 0.0$ and Mach number $M = 0.0$) when the engine is on the ground for fault monitoring and maintenance by the engineering personnel. The engine simulation is conducted at a frequency of 66.67 Hz (i.e. inter-sample time of 15 ms) and the length of the simulation time window is 150 s, which generate 10 000 data points for each training or test case.

Table 1 Required engine system sensors

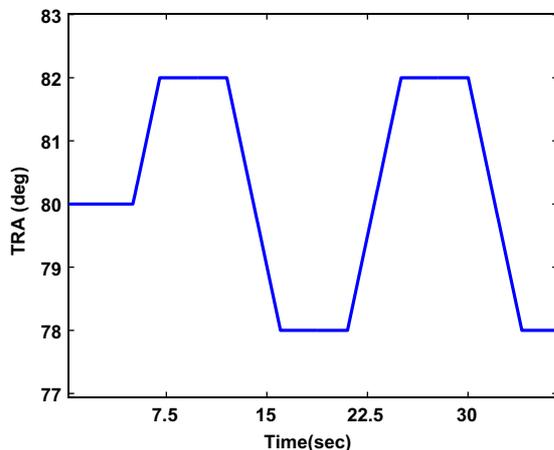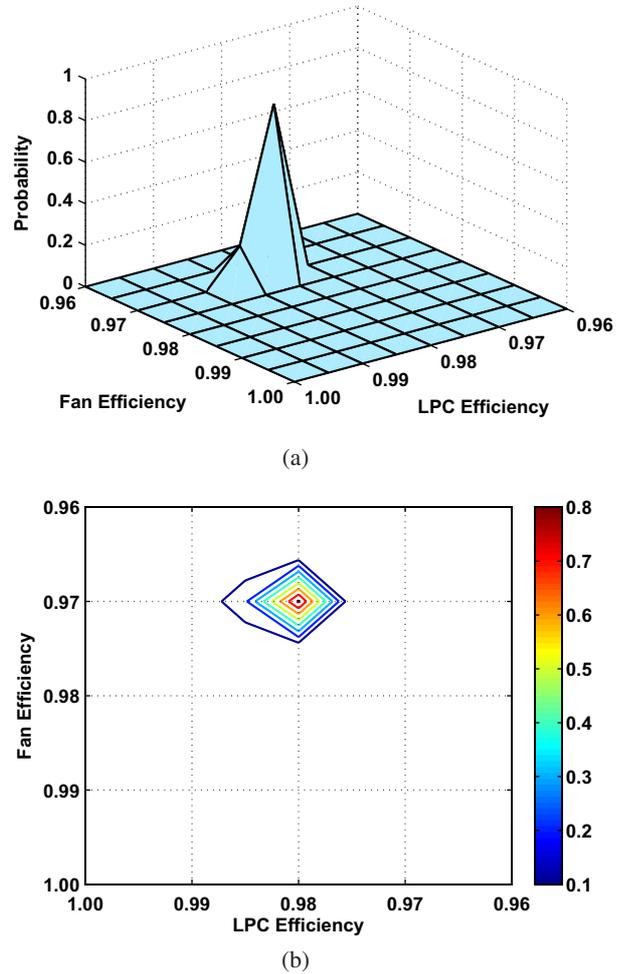| Sensors	Description
P_{24}	LPC exit/HPC inlet pressure
T_{24}	LPC exit/HPC inlet temperature
Ps_{30}	HPC exit static pressure
T_{30}	HPC exit/burner inlet temperature
T_{50}	LPT exit temperature

Fig. 3 TRA profile

Fig. 4 Fault estimation in fan-LPC based on Ps_{30} sensor: (a) surface plot of fault estimation and (b) contour plot of fault estimation

An engine component C is considered in nominal condition when both ψ_C and ζ_C are equal to 1. Fault is injected in the component C by simultaneously reducing both ψ_C and ζ_C by same amount in the results reported in this article. Although the algorithm described above, does not have any restriction on the dimension of the parameter space, the result presented here considers simultaneous degradation of two different components. Section 3.1.1 describes a fault condition, where Fan and LPC are degraded simultaneously, whereas section 3.1.2 analyses simultaneous degradation in HPT and LPT. For both training (i.e. forward problem) and testing (i.e. inverse problem), time series data from all sensors, listed in Table 1, are generated with ψ and ζ ranging from 1.0 to 0.96 (i.e. 4 per cent relative loss in efficiency) in steps of 0.005 for the engine components under consideration. For SDF analysis, the number of states in the PFSA is selected to be 15 for each sensor after preprocessing the time series data by Hilbert transform and pattern vectors are generated for each of the possible fault conditions. Fifty repetitions of each
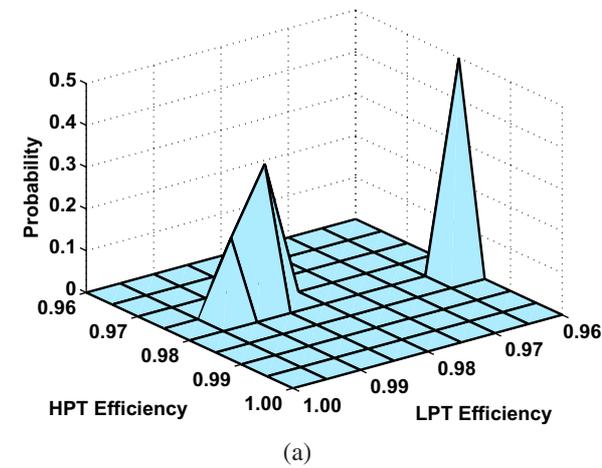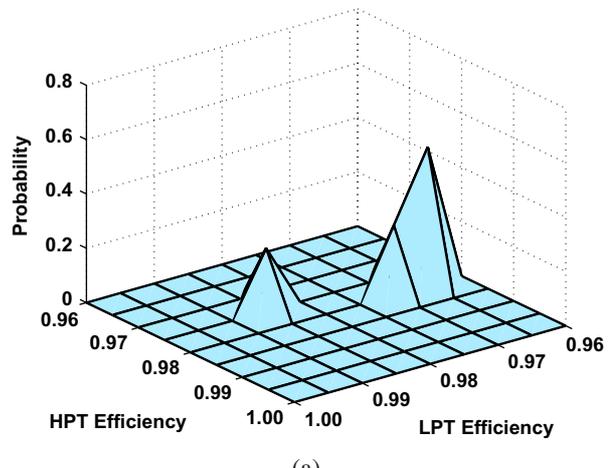
(a)
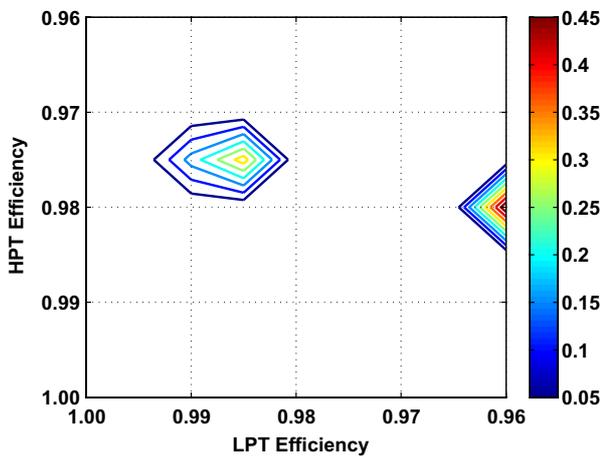
(b)

Fig. 5 Fault estimation in HPT–LPT based on Ps_{30} sensor: (a) surface plot of fault estimation and (b) contour plot of fault estimation

(a)

(b)

Fig. 6 Fault estimation in HPT–LPT based on T_{24} sensor: (a) surface plot of fault estimation and (b) contour plot of fault estimation
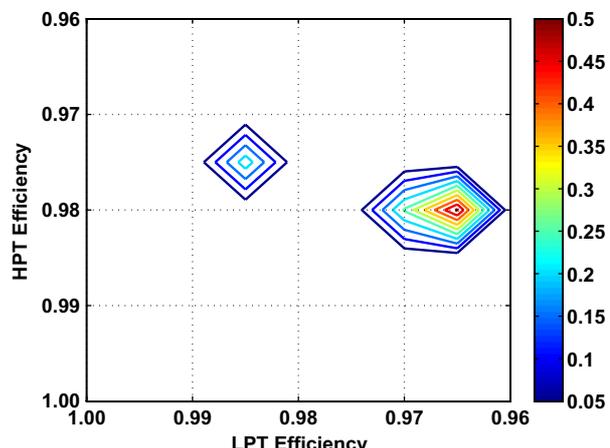
simulation have been conducted to generate pattern vector statistics with injected process and sensor noise. For testing (i.e. inverse problem), fault conditions are chosen within the range of training data such that they do not coincide with the training grid points. The results of multiple-fault estimation are presented in the following two sections along with discussions on sensor fusion.

3.1.1 Fault estimation in Fan and LPC

A test pattern is generated for a given fault condition, $\psi_F = \zeta_F = 0.973$ and $\psi_{LPC} = \zeta_{LPC} = 0.981$. The three-dimensional plot in Fig. 4(a) shows the bivariate probability distribution of the estimated fault condition, followed by a close-up view of the contour plots in Fig. 4(b), where the results are generated from time series of a single sensor, namely, Ps_{30}. The estimates lie within the $\pm 3\sigma$ bound around the estimated mean (see equation (9)), where the variance σ^2 is obtained as a diagonal element of the estimated covariance matrix $\hat{\mathbf{C}}_s$ (see equation (10)). In this case, the estimates range

from 0.9606 to 0.9704 for ψ_F and ζ_F, and from 0.9805 to 0.9813 for ψ_{LPC} and ζ_{LPC}, respectively. This indicates that the correct region is located in the parameter space, which assigns highest probability to the nearest training grid point.

3.1.2 Fault estimation in HPT-LPT

This example shows the result for a fault condition, $\psi_{HPT} = \zeta_{HPT} = 0.977$ and $\psi_{LPT} = \zeta_{LPT} = 0.985$. In contrast to the previous example of fan and LPC, the plots in Figs 5(a) and (b) show that there is an ambiguity in estimation when using information from only one sensor, namely Ps_{30}. Although, it identifies the correct region with significant probability, another fault condition is seen to be identified with higher probability. Similar is the result if sensor T_{24} is used, as seen in Figs 6(a) and (b). To resolve this ambiguity, the sensor information fusion framework makes use of both Ps_{30} and T_{24} to correctly identify the fault in the parameter space without any ambiguity, as seen in Figs 7(a) and (b). The estimates lie in the ranges ($\pm 3\sigma$ bound)
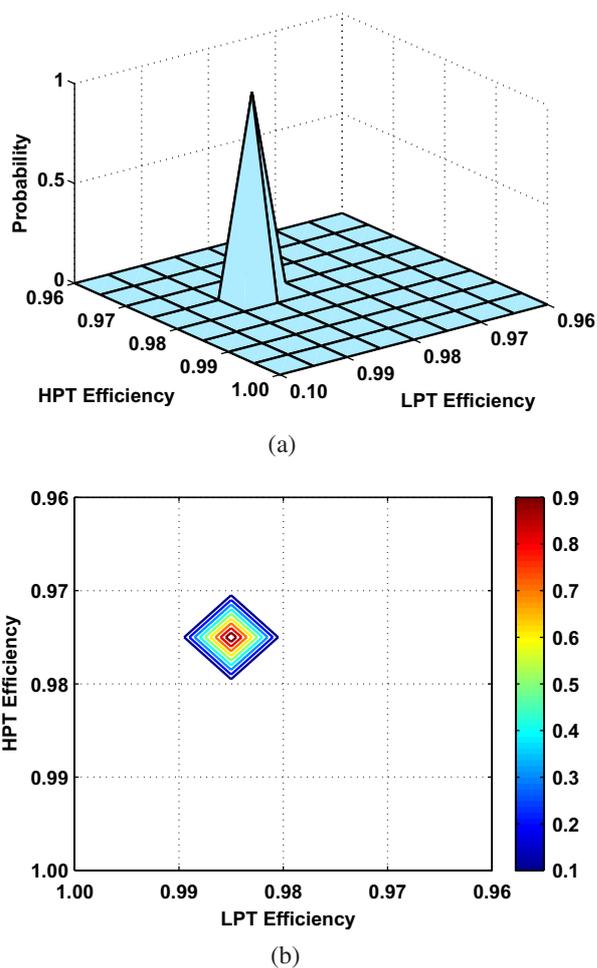
(a)

(b)

Fig. 7 Fault estimation in HPT–LPT based on Ps_{30} and T_{24} sensors: (a) surface plot of fault estimation and (b) contour plot of fault estimation

of 0.9747 to 0.9753 for ψ_{HPT} and ζ_{HPT} and 0.9847 to 0.9853 for ψ_{LPT} and ζ_{LPT}, respectively; in this case, highest probability is assigned to the training grid point that is nearest to the test point.

4 SUMMARY, CONCLUSIONS, AND FUTURE WORK

This article presents a SDF-based methodology for multiple-fault estimation in aircraft gas turbine engines. It also proposes and validates a sensor-information-fusion framework to alleviate certain problems of data-level (i.e. related to scaling) and decision-level (i.e. resolution of detection) techniques for information fusion. The proposed fault estimation tool is sensor-data-driven and is apparently applicable for early detection of multiple faults for prognosis of catastrophic failures in aircraft gas turbine engines. The underlying algorithm can be implemented on small microprocessors as it enables compression of information into pattern vectors of low dimension for real-time execution on limited-memory platforms.

While there are many other issues that need to be addressed before the proposed estimation method can be considered for real-life applications in commercial aircraft, the following research topics are being currently pursued.

1. Validation of the proposed method for fault estimation of a larger number (i.e. more than two) of components in gas turbine engines.
2. Identification of the mode (e.g. gradual deterioration, intermittently occurring, and abrupt large) of multiple fault evolution as an extension of the earlier work on single faults [**28**].
3. Extension of the fault estimation problem under different types of non-linearities (e.g. dead band and hysteresis in actuators) with structured and unstructured uncertainties.
4. Usage of *a priori* information about the likelihood of the test fault condition.
5. Extension of *fault estimation* to *fault prognosis* by using usual deterioration profiles of the components with respect to cycles of operation.
6. Development of a better damage injection model.
7. Optimal sensor selection based on the estimation results of the whole parameter space.
8. Incorporation of other fault types, such as sensor faults and actuator faults.

REFERENCES

 1 Hoffman, A. J. and **van der Merwe, N. T.** The application of neural networks to vibrational diagnostics for multiple fault conditions. *Comput. Stand. Interfaces*, 2002, **24**(2), 139–149.
 2 Boppana, V., Hartanto, I., and **Fuchs, W. K.** Fault diagnosis using state information. In FTCS '96: Proceedings of the 26th Annual International Symposium on *Fault-tolerant computing* (*FTCS '96*), Washington, DC, 1996, p. 96 (IEE Computer Society, USA).
 3 Ding, Y., Wu, Z., and **Zhang, Y.** Multi-fault diagnosis method based on a joint estimation of states and fault parameters. *J. Tsinghua Univ.*, 2001, **41**(12), 92–94.
 4 Song, H. and **Zhang, H. Y.** Approach for multiple faults diagnosis based on parity equation and parameter estimation. *Control Decis.*, 2003, **18**(4), 413–417.
 5 Huang, H. P., Li, C. C., and **Jeng, J. C.** Multiple multiplicative fault diagnosis for dynamic processes via parameter

similarity measures. *Ind. Eng. Chem. Res.*, 2007, **46**(13), 4517–4530.

6 Broersen, P. Estimation of parameters of non-linear dynamical systems. *Int. J. Non-linear Mech.*, 1974, **9**, 355–361.

7 Van Lith, P. F., Witteveen, H., Betlem, B. H. L., and **Roffel, B.** Multiple nonlinear parameter estimation using PI feedback control. *Control Eng. Pract.*, 2001, **9**, 517–531.

8 Wan, E. and **Van der Merwe, R.** The unscented Kalman filter for nonlinear estimation. In Proceedings of the IEEE Symposium 2000 AS-SPCC, Lake Louise, Alberta, Canada, 2000, pp. 153–158.

9 Julier, S. and **Uhlmann, J.** A new extension of the Kalman filter to nonlinear systems. In Proceedings of the International Symposium on Aerospace/Defense Sensing, Simulation and Controls, Orlando, Florida, 1997.

10 Ching, J., Beck, J. L., and **PorterChin K. A.** Bayesian state and parameter estimation of uncertain dynamical systems. *Probab. Eng. Mech.*, 2006, **21**(1), 81–96.

11 Bremer, C. and **Kaplan, D.** Markov chain Monte Carlo estimation of nonlinear dynamics from time series. *Phys. D*, 2001, **160**, 116–126.

12 Wang, Y. and **Geng, L.** Bayesian network based fault section estimation in power systems. In Proceedings of the IEEE Region 10th Annual International Conference, TENCON, Hong Kong, People's Republic of China, 2006.

13 David, B. and **Bastin, G.** Parameter estimation in non-linear systems with auto and crosscorrelated noise. *Automatica*, 2002, **38**, 81–90.

14 Ghanem, R. and **Romeo, F.** A wavelet-based approach for model and parameter identification of non-linear systems. *Int. J. Non-linear Mech.*, 2001, **36**, 835–859.

15 Yao, L. and **Sethares, W.** Nonlinear parameter estimation via the genetic algorithm. *IEEE Trans. Signal Process.*, 1994, **42**(4), 927–935.

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

17 Rajagopalan, V., Chakraborty, S., and **Ray, A.** Estimation of slowly varying parameters in nonlinear systems via symbolic dynamic filtering. *Signal Process.*, 2008, **88**(2), 339–348.

18 Gupta, S., Ray, A., and **Keller, E.** Symbolic time series analysis of ultrasonic data for early detection of fatigue damage. *Mech. Syst. Signal Process.*, 2007, **21**(2), 866–884.

19 Rao, C., Ray, A., Sarkar, S., and **Yasar, M.** Review and comparative evaluation of symbolic dynamic filtering for detection of anomaly patterns. *Signal, Image Video Process.*, 2008, **3**(2), 1863–1703. DOI: 10.1007/s11760-008-0061-8.

20 Rao, C., Mukherjee, K., Sarkar, S., and **Ray, A.** Estimation of multiple parameters in dynamical systems. In Proceedings of the American Control Conference, Seattle, Washington, USA, June 2008.

21 Gupta, S., Ray, A., Sarkar, S., and **Yasar, M.** Fault detection and isolation in aircraft gas turbine engines: part I – underlying concept. *Proc. IMechE, Part G: J. Aerospace Engineering*, 2008, **222**(G3), 307–318. DOI: 10.1243/09544100JAERO311.

22 Sarkar, S., Yasar, M., Gupta, S., Ray, A., and **Mukherjee, K.** Fault detection and isolation in aircraft gas turbine engines: part II – validation on a simulation test bed. *Proc. IMechE, Part G: J. Aerospace Engineering*, 2008, **222**(G3), 319–330. DOI: 10.1243/09544100JAERO312.

23 Frederick, D. K., DeCastro Jonathan, A., and **Litt, J. S.** User's guide for the commercial modular aero-propulsion system simulation (C-MAPSS). NASA/TM2007-215026, 2007.

24 Rajagopalan, V. and **Ray, A.** Symbolic time series analysis via wavelet-based partitioning. *Signal Process.*, 2006, **86**(11), 3309–3320.

25 Subbu, A. and **Ray, A.** Space partitioning via Hilbert transform for symbolic time series analysis. *Appl. Phys. Lett.*, 2008, **92**(8), 084107–1–084107–3.

26 Duda, R., Hart, P., and **Stork, D.** *Pattern classification*, 2001 (John Wiley, New York).

27 Kobayashi, T. and **Simon, D. L.** A hybrid neural network-genetic algorithm technique for aircraft engine performance diagnostics. In Proceedings of the 37th Joint Propulsion Conference and Exhibit, cosponsored by the AIAA, ASME, SAE, and ASEE, Salt Lake City, Utah, 2001.

28 Tolani, D. K., Yasar, M., Ray, A., and **Yang, V.** Anomaly detection in aircraft gas turbine engines. *AIAA J. Aerosp. Comput. Inf. Commun.*, 2006, **3**(2), 44–51.