

# Semantic Sensor Fusion for Fault Diagnosis in Aircraft Gas Turbine Engines★

Soumik Sarkar   Dheeraj Sharan Singh   Abhishek Srivastav   Asok Ray  
szs200@psu.edu   dss240@psu.edu   axs964@psu.edu   axr2@psu.edu  
*Department of Mechanical Engineering*

*The Pennsylvania State University  
University Park, PA 16802, USA*

Keywords: *Gas Turbine Engines; Fault Diagnosis; Sensor Fusion; Statistical Pattern Recognition*

**Abstract**—Data-driven fault diagnosis of a complex system such as an aircraft gas turbine engine requires interpretation of multi-sensor information to assure enhanced performance. This paper proposes feature-level sensor information fusion in the framework of symbolic dynamic filtering. This hierarchical approach involves construction of composite patterns consisting of: (i) atomic patterns extracted from single sensor data and (ii) relational patterns that represent the cross-dependencies among different sensor data. The underlying theories are presented along with necessary assumptions and the proposed method is validated on the NASA C-MAPSS simulation model of aircraft gas turbine engines.

## 1. INTRODUCTION

Data-driven fault diagnosis in human-engineered complex systems is extremely crucial from the perspective of safety and reliability. However, in complex engineering system such as an aircraft gas turbine engine, the patterns generated from a single sensor observation may not carry sufficient information to correctly diagnose a fault in the system. Moreover, simultaneously occurring different faults in the same or different subsystems may generate similar signatures in a particular sensor observation. Hence, a data-driven fault detection tool for an aircraft engine system should have the capability to characterize, quantify and interpret multiple sensor outputs. However, sensor information fusion for a complex system like an aircraft is extremely challenging. First of all, an aircraft system has sensors with a large variation in modality (e.g., pressure, temperature, speed, and acceleration). Hence, a data-level information fusion is extremely difficult due to the inherent scaling problem of sensor data of different modality. On the other hand, decision level fusion generally requires an in depth understanding of the physical system and its failure signatures in different sensor

observations. In literature, Dempster-Shafer evidence theory has been applied for engine fault diagnosis [1]; similarly, the concept of Bayesian Belief Networks has been used for fault diagnosis in turbofan engines [2]. Both techniques belong to the class of decision fusion at an upper level.

Recently, in a two-part paper [3][4], 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 [5]. One of the main aspects of SDF-based method is semantic representation of sensor data, irrespective of modality and other sensor specific characteristics. This would facilitates feature-level fusion of non-homogeneous sensors. They are derived based on a *semantic framework* for pattern extraction and classification. To handle a large volume of data in real time, a hierarchical framework for information fusion is proposed that progressively leads from machine representations of observed data to fault classification. One of novelties of the proposed approach is identifying the cross-dependence among difference sensor observations to reduce loss of information.

## 2. DATA-DRIVEN FAULT DIAGNOSIS POSED AS A MULTI-CLASS PATTERN CLASSIFICATION PROBLEM

This section formulates the problem based on the C-MAPSS test-bed [6] presented in the sequel.

### A. Description of the C-MAPSS Test-bed

The C-MAPSS simulation test-bed is the model of a 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 sea level (i.e., 0 m) up to 12,200 m, Mach number ( $M$ ) from 0 to 0.90, and sea-level temperatures from approximately  $-50^{\circ}\text{C}$  to  $50^{\circ}\text{C}$ . The throttle resolving angle ( $TRA$ ) can be set to any value in the range between

★This work has been supported in part by NASA under Cooperative Agreement No. NNX07AK49A, by the U.S. Office of Naval Research under Grant No. N00014-09-1-0688, and by the U.S. Army Research Laboratory and the U.S. Army Research Office under Grant No. W911NF-07-1-0376. 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

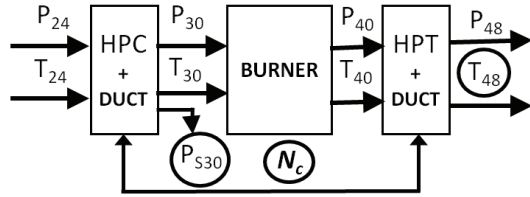


Fig. 1. Schematic diagram of the HPC-HPT subsystem

$0^\circ$  at the minimum power level and  $100^\circ$  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). In the current configuration of the C-MAPSS simulation test-bed, ten component level health parameter inputs, namely, efficiency parameters ( $\psi$ ), flow parameters ( $\zeta$ ) are used to inject faults. For the engine's five rotating components F, LPC, HPC, LPT, and HPT, such parameters are:  $(\psi_F, \zeta_F)$ ,  $(\psi_{LPC}, \zeta_{LPC})$ ,  $(\psi_{HPC}, \zeta_{HPC})$ ,  $(\psi_{HPT}, \zeta_{HPT})$ , and  $(\psi_{LPT}, \zeta_{LPT})$ . An engine component  $C$  is considered to be in nominal condition if both  $\psi_C$  and  $\zeta_C$  are equal to 1 and fault can be injected in the component  $C$  by reducing the values of  $\psi_C$  and/or  $\zeta_C$ . For example,  $\psi_{HPC} = 0.98$  signifies a 2% relative loss in efficiency of HPC.

### B. Problem Formulation

Component-level fault diagnosis in an aircraft gas turbine engine involves identification of the fault type, and location & quantification of the fault level. In the C-MAPSS test-bed setting, the physical fault scenarios (e.g., fouling, increased tip clearance, and seal wear) are assumed to manifest themselves in affecting the efficiency and flow of the associated engine component(s). In the present case study, a simultaneous fault scenario has been considered involving two major rotating components of the engine, namely, HPC and HPT. Choice of these components has important significance from the perspective of diagnosis of simultaneously occurring faults. Both HPC and HPT are mounted on the core shaft of the engine. Hence, they have strong mechanical interconnection between them. Besides, they also have electrical interconnection via control loop. Such strong electro-mechanical interconnection increases difficulty of the fault diagnosis problem. On the other hand, information from an HPC sensor will possibly have strong cross-dependency with an HPT sensor and it may not be reasonable to ignore that for the purpose of fault diagnosis. In the present study, three heterogeneous, non-collocated and commonly used sensors are selected that are placed in the HPC-HPT subsystem as shown in Figure 1. The three sensors are as follows:

Sensor noise standard deviation values are provided as percent of operating point trim values [7]. The three chosen sensors are of different modalities (pressure, temperature, speed) and while the pressure sensor is located at HPC exit,

TABLE I  
SENSORS FOR FAULT DIAGNOSIS IN THE HPC-HPT SUBSYSTEM

Sensors	Description	Noise Std. (%)
$P_{s30}$	HPC exit static pressure	0.50
$T_{48}$	HPT exit temperature	0.75
$N_c$	Core spool speed	0.25

the temperature sensor is located at HPT exit and the speed sensor is measuring the rotational speed of the core shaft on which both HPC and HPT are mounted. The challenge here is to identify the relational dependencies among these sensor data to enhance the fault diagnosis performance. Diagnosis involves both fault localization and fault level identification. The health parameters that defines the health status of HPC and HPT, are the efficiency and flow health parameters ( $\psi_{HPC}, \zeta_{HPC}$  and  $\psi_{HPT}, \zeta_{HPT}$ ). Assuming that both efficiency and flow parameters change simultaneously by the same amount, three fault levels are considered similarly for both HPC and HPT. Table II shows the approximate ranges of health parameters under different fault levels. Here, the low fault level indicate very minimal loss in efficiency and flow and hence also includes the absolute nominal health condition ( $\psi_{HPC} = \zeta_{HPC} = 1$  or  $\psi_{HPT} = \zeta_{HPT} = 1$ ).

TABLE II  
FAULT LEVELS IN HPC/HPT

Fault Level	Efficiency/Flow Range
Low Fault	1.0000 to 0.9867
Medium Fault	0.9867 to 0.9733
High Fault	0.9733 to 0.9600

In this study, classes are defined as Cartesian products of the ranges of HPC and HPT health parameters. Thus, there are 9 (i.e.,  $3 \times 3$ ) classes of data that can be obtained when a class is uniquely defined by an HPC fault level (a range of HPC health parameters) and an HPT fault level (a range of HPT health parameters). 100 simulation runs of the engine system have been conducted for each class to generate data set for analysis among which 50 samples are chosen as the training set and the remaining 50 samples are kept as testing set. HPC/HPT health parameters are chosen randomly from independent Uniform distributions for health parameters within the prescribed ranges given in above table. Figure 2 plots the samples generated using the above logic in the two dimensional parameter space. Different classes of samples are shown in different colors and as well as marked with the class numbers in the figure.

*Remark 2.1:* The health parameter distribution is artificially chosen here as uniform to cover the whole space in order to validate the proposed technique. However, extensive experimentation involving real faults (or realistic damage models) may result in a different distribution of health parameters, e.g., Gaussian distribution.

For each data sample, a time series was collected for all three sensors (given in Table I) under persistent excitation of

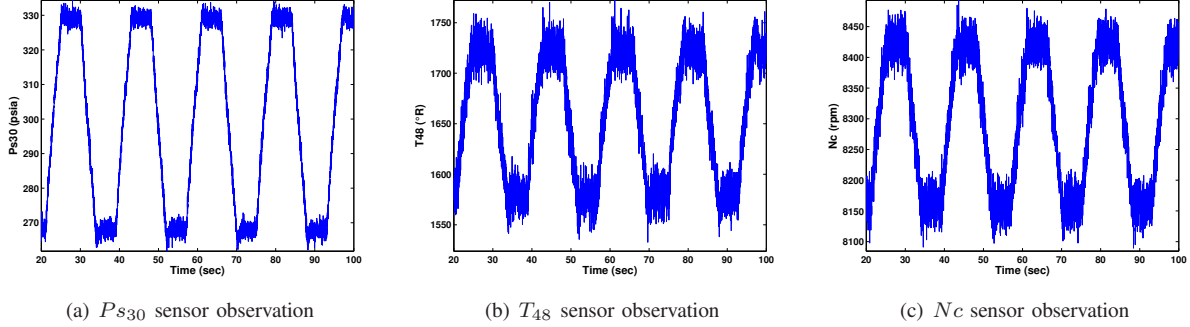


Fig. 3. Representative time series observations from different Sensors

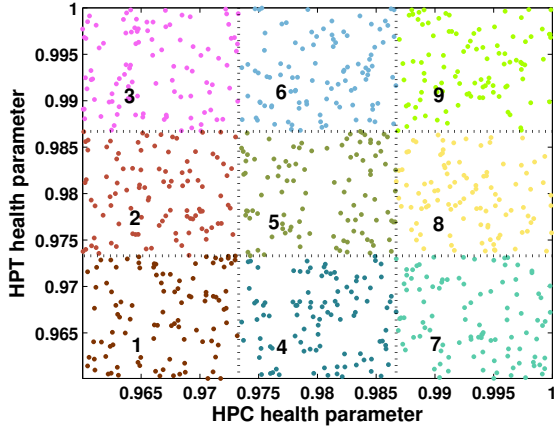


Fig. 2. Fault classes for data collection and classification

*TRA* inputs that have truncated triangular profiles with the mean value of  $40^\circ$ , fluctuations within  $\pm 8^\circ$  and frequency of 0.056 Hz. The ambient conditions are chosen to be at the sea level when the engine is on the ground (i.e. altitude  $A = 0.0$ , Mach number  $M = 0.0$ ) for fault monitoring and maintenance by the engineering personnel. For each experiment, the engine simulation is conducted at a frequency of 66.67 Hz (i.e., inter-sample time of 15ms) and the length of the simulation time window is 150 seconds, which generate 10,000 data points. Figure 3 shows representative examples of time series data from each of the three sensors.

### 3. SEMANTIC FRAMEWORK FOR MULTI-SENSOR DATA INTERPRETATION AND FUSION

A hierarchical (three-layered) semantic framework is proposed in this paper for the purpose of multi-sensor data interpretation and fusion. The basic structure of this architecture is inspired by the information fusion model proposed by the Data Fusion Information Group (DGIF) [8]. The lowest level of this hierarchy deals with signal conditioning, transformation and finally feature extraction for unimodal sensor data streams. In the present framework, patterns discovered from individual sensors are called atomic patterns and SDF is used to extract them. A brief review of SDF is

provided in Section 3-A for the completeness of the paper.

Let  $\mathbb{L} = \{\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_N\}$  be the universal set of *atomic* patterns. The *atomic* pattern library  $\mathbb{L}$  is set of modal footprints identified from individual sensing modalities for various fault classes. Given the atomic pattern library, a popular framework for addressing information fusion is what is called the *set-theoretic* approach. In this framework, higher level patterns or contexts are modeled as subsets of  $\mathbb{L}$ . Thus a composite pattern, resulting from fusion of atomic patterns, is a collection of elements from  $\mathbb{L}$  and the composite pattern library  $\mathbb{L}^* \subset 2^{\mathbb{L}}$ . The disadvantage of this approach is that it considers only modal footprints for constructing composite patterns as a *bag of atomic patterns*; relational dependencies, if any, between patterns are disregarded. However, the relational dependencies should not be ignored for many problems in practice, e.g., in the present problem of fault diagnosis of simultaneously degrading electro-mechanically connected aircraft engine components. Therefore, a hierarchical semantic framework for multi-sensor data interpretation and fusion is proposed that involves a common approach to information fusion going from one level to another and to include relational dependencies for composite pattern representation. Thus, the middle layer deals with the relational dependencies among atomic patterns, where relationships are modeled as the cross-dependencies among sensor data streams from different sensors. The cross-dependencies are discovered via relational PFSA that essentially capture the dynamics of state transition in one symbol sequence (obtained from one sensor) corresponding to a symbol appearance in the second symbol sequence (obtained from another sensor). Loose time-synchronization between sensor observations will be enough for this purpose. Symbol-level cross-dependencies between modalities are exploited to reduce information loss.

Finally, the top layer consists of higher level composite patterns that will be represented as digraphs where the atomic patterns (AP) are modeled as nodes and dependencies between nodes are modeled as relational patterns (RP). A formal definition is as follows:

*Definition 3.1 (Composite pattern representation):* Let  $\mathbb{L} = \{\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_N\}$  be the atomic pattern library. Let  $\mathbb{L}^* \subset 2^{\mathbb{L}}$  be the set of allowable primitives for a class.

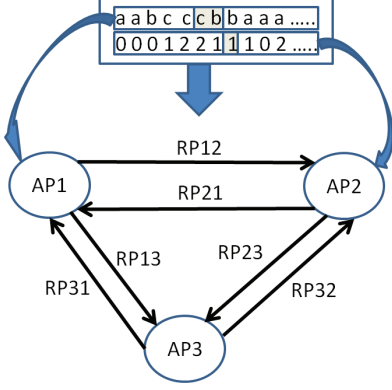


Fig. 4. Composite Pattern Digraph

Then a composite pattern library  $\mathbb{H}^r = \{\mathcal{H}_1^r, \mathcal{H}_2^r, \dots, \mathcal{H}_M^r\}$  where a composite pattern  $\mathcal{H}_i^r$  is digraph  $\mathcal{H}_i^r = (\mathcal{L}_{V_i}, \mathcal{E}_{V_i})$ ;  $\mathcal{L}_{V_i} \subset \mathbb{L}$  with the index set  $V_i \subset \{1, 2, \dots, N\}$  and  $\mathcal{E}_i = \{\mathcal{R}_{jk} | j, k \in V_i \times V_i\}$  is a set of *relational* PFSA. The digraph representation is illustrated in Figure 4.

The relational probabilistic finite state automata (PFSA) are discovered using xD-Markov machine construction to determine cross-dependence; the algorithm is described in Section 3-B.

#### A. Symbolic Dynamic Filtering for Feature Extraction

This section briefly describes the concepts of Symbolic Dynamic Filtering (SDF) for extracting atomic patterns from single sensor data. Symbolic feature extraction from time series data is posed as a two-time-scale problem. The *fast scale* is related to the response time of the process dynamics. Over the span of data acquisition, dynamic behavior of the system is assumed to remain invariant, i.e., the process is quasi-stationary at the fast scale. On the other hand, the *slow scale* is related to the time span over which non-stationary evolution of the system dynamics may occur. It is expected that the features extracted from the fast-scale data will depict statistical changes between two different slow-scale epochs if the underlying system has undergone a change. The method of extracting features from stationary time series data is comprised of the following steps.

- Sensor time series data, generated from a physical system or its dynamical model, are collected at a slow-scale epoch and let it be denoted as  $\mathbf{q}$ . A compact (i.e., closed and bounded) region  $\Omega \in \mathbb{R}^n$ , where  $n \in \mathbb{N}$ , within which the stationary time series is circumscribed, is identified. Let the space of time series data sets be represented as  $\mathcal{Q} \subseteq \mathbb{R}^{n \times N}$ , where  $N \in \mathbb{N}$  is sufficiently large for convergence of statistical properties within a specified threshold. While  $n$  represents the dimensionality of the time-series,  $N$  is the number of data points in the time series. Then,  $\{\mathbf{q}\} \in \mathcal{Q}$  denotes a time series at the slow-scale epoch of data collection.

- Encoding of  $\Omega$  is accomplished by introducing a partition  $\mathbb{B} \triangleq \{B_0, \dots, B_{(|\Sigma|-1)}\}$  consisting of  $|\Sigma|$  mutually exclusive (i.e.,  $B_j \cap B_k = \emptyset \forall j \neq k$ ), and exhaustive (i.e.,  $\cup_{j=0}^{|\Sigma|-1} B_j = \Omega$ ) cells, where each cell is labeled by symbols  $\sigma_j \in \Sigma$  and  $\Sigma = \{\sigma_0, \dots, \sigma_{|\Sigma|-1}\}$  is called the alphabet. This process of coarse graining can be executed by uniform, maximum entropy, or any other scheme of partitioning. Then, the time series data points that visit the cell  $B_j$  are denoted as  $\sigma_j \forall j = 0, 1, \dots, |\Sigma| - 1$ . This step enables transformation of the time series data  $\{\mathbf{q}\}$  to a symbol sequence  $\{\mathbf{s}\}$ , consisting of the symbols  $\sigma_j$  in the alphabet  $\Sigma$ .
- A probabilistic finite state machine (PFSA), is then constructed and the symbol sequence  $\{\mathbf{s}\}$  is run through the PFSA. The PFSA considered in this framework is known as D-Markov machine [9]. A formal definition is as follows:

*Definition 3.2 (D-Markov):* A D-Markov machine of depth  $D$  is defined as a 4-tuple  $\mathcal{M} \triangleq (\mathcal{Q}, \Sigma, \delta, \tilde{\Pi})$  such that:

- $\Sigma = \{\sigma_0, \dots, \sigma_{|\Sigma|-1}\}$  is the alphabet set of symbol sequence  $\{\mathbf{s}\}$
- $\mathcal{Q} = \{q_1, q_2, \dots, q_{|\Sigma|^D}\}$  is the state set corresponding to symbol sequence  $\{\mathbf{s}\}$ . States represent all possible words of length  $D$ , using the symbol alphabet
- $\delta : \mathcal{Q} \times \Sigma \rightarrow \mathcal{Q}$  is the state transition mapping that maps the transition in a symbol sequence from one state to another upon arrival of a symbol
- $\tilde{\Pi}$  is the symbol generation matrix of size  $|\mathcal{Q}| \times |\Sigma|$ ; an element  $\tilde{\Pi}_{ij}$  denotes the probability of finding  $j^{\text{th}}$  symbol from  $i^{\text{th}}$  state in the symbol sequence  $\{\mathbf{s}\}$

In practice,  $\tilde{\Pi}$  is reshaped into a vector of length  $|\mathcal{Q}| \times |\Sigma|$  and is treated as the extracted feature vector that is a low-dimensional representation of the dynamical system at the slow-scale epoch. This feature vector obtained from a single symbol sequence (sensor data) is called an Atomic Pattern (AP).

For pattern classification applications, a time-series from the reference class is partitioned using a partitioning scheme (e.g., Uniform Partitioning (UP) or Maximum Entropy partitioning (MEP)) [9][10][11]. Then, using the steps described before, a low dimensional feature is constructed for the reference class. Similar features can be extracted from time-series data of all classes using the same partitioning. Finally a classifier is trained using features of different classes extracted from training data and can be used to classify the features from test data set. In the present context, a non-parametric classifier, such as the k-NN classifier may a better candidate for this study [12][13]; however, any other suitable classifier, such as the Support Vector Machines (SVM) or the Gaussian Mixture Models (GMM), may also be used.

### B. Construction of Relational PFSA: xD-Markov machine

This section describes the construction of xD-Markov machines from two symbol sequences  $\{s_1\}$  and  $\{s_2\}$  obtained from two different sensors (possibly of different modalities) to capture the symbol level cross-dependence. A formal definition is as follows:

*Definition 3.3 (xD-Markov):* Let  $\mathcal{M}_1$  and  $\mathcal{M}_2$  be the PFSA's corresponding to symbol streams  $\{s_1\}$  and  $\{s_2\}$  respectively. Then a xD-Markov machine is defined as a 5-tuple  $\mathcal{M}_{1 \rightarrow 2} \triangleq (\mathcal{Q}_1, \Sigma_1, \Sigma_2, \delta_1, \tilde{\Pi}_{12})$  such that:

- $\Sigma_1 = \{\sigma_0, \dots, \sigma_{|\Sigma_1|-1}\}$  is the alphabet set of symbol sequence  $\{s_1\}$
- $\mathcal{Q}_1 = \{q_1, q_2, \dots, q_{|\Sigma_1|^{D_1}}\}$  is the state set corresponding to symbol sequence  $\{s_1\}$ , where  $D_1$  is the depth for  $\{s_1\}$
- $\Sigma_2 = \{\sigma_0, \dots, \sigma_{|\Sigma_2|-1}\}$  is the alphabet set of symbol sequence  $\{s_2\}$
- $\delta_1 : \mathcal{Q}_1 \times \Sigma_1 \rightarrow \mathcal{Q}_1$  is the state transition mapping that maps the transition in symbol sequence  $\{s_1\}$  from one state to another upon arrival of a symbol in  $\{s_1\}$
- $\tilde{\Pi}_{12}$  is the symbol generation matrix of size  $|\mathcal{Q}_1| \times |\Sigma_2|$ ; the  $i^{th}$  element of  $\tilde{\Pi}_{12}$  denotes the probability of finding  $j^{th}$  symbol in  $\{s_2\}$  while making a transition from  $i^{th}$  state in the symbol sequence  $\{s_1\}$

In practice,  $\tilde{\Pi}_{12}$  is reshaped into a vector of length  $|\mathcal{Q}_1| \times |\Sigma_2|$  and is treated as the extracted feature vector that is a low-dimensional representation of the relational dependence between  $\{s_1\}$  and  $\{s_2\}$ . This feature vector is called a Relational Pattern (RP). Figure 4 schematically describes the basic concept of the xD-Markov machine. Note, a Relational Pattern between two symbol sequences is not symmetric; therefore, RPs need to be identified for both directions. Also, when both symbol sequences are same, the relational patterns are essentially the atomic pattern corresponding to the symbol sequence; i.e., xD-Markov machine reduces to a simple D-Markov machine.

In the set-theoretic approach all relationships are excluded and any fusion is solely done in the decision-theoretic sense where the presence (or absence) of one or more footprints can be used to estimate the probability of the fault class under consideration. The other end of the spectrum is to fuse data at the lowest level and construct machines (PFSA's) working in the product space of all sensors. This approach would be able to extract modal dependencies before they are lost when constructing separate machines for individual sensor or modalities. But working in the product space has the danger of state space explosion especially when the sensors and sensing modalities can be numerous, as in a case of a modern aircraft engine equipped with high number of sensors. The proposed approach is a trade-off between the two ends of the spectrum and attempts to include relational dependencies between sensing modalities, while keeping it tractable for a practical application.

## 4. RESULTS AND DISCUSSION

This section presents pertinent fault diagnosis results for the HPC-HPT subsystem. For data partitioning, maximum entropy partitioning is used with alphabet size,  $|\Sigma| = 6$  for all three sensors (although alphabet size does not need to be same for different sensors). The depth for constructing PFSA states is taken to be,  $D = 1$  for both atomic and relational pattern construction and features are classified by a k-NN classifier (with  $k = 5$ ) using the Euclidean distance metric. For a particular health parametric condition, three atomic patterns are generated from three sensor observations and six relational patterns are generated by extracting pairwise directed cross-dependencies. Finally all the patterns are concatenated to construct the overall composite pattern. The classification error on the test data set using the composite pattern is found to be 11.56% and Table III provides the classification errors corresponding to all atomic and relational patterns. The cross-dependence direction is from Sensor 1 to Sensor 2 in the table. Hence, the diagonal elements represent the classification error percentages corresponding to the atomic patterns, where as the off-diagonal elements represent the classification error percentages corresponding to the relational patterns.

TABLE III  
COMPARISON OF CLASSIFICATION ERROR PERCENTAGES USING ATOMIC AND RELATIONAL PATTERNS ON TEST DATA SET ( $50 \times 9$  SAMPLES); CROSS-DEPENDENCE DIRECTION: *Sensor1*  $\rightarrow$  *Sensor2*

		Sensor 2		
		$P_{s_{30}}$	$T_{48}$	$N_c$
Sensor 1	$P_{s_{30}}$	56.44	15.11	41.56
	$T_{48}$	12.89	56.89	16.00
	$N_c$	25.11	11.78	52.22

It is observed from the above table that the relational patterns are able to extract useful information from the perspective of fault diagnosis. The problem in the present study has been posed in such a way that the sensor information from different sensors actually have cross-dependencies due to strong electro-mechanical interconnections between HPC and HPT. Therefore, ignoring these cross-dependencies should affect the fault diagnosis results. This result confirms the conjecture and shows that the xD-Markov machine construction can extract those cross-dependencies. It should be noted that the fault diagnosis algorithm is completely data-driven and has no model information. Therefore, the result is significantly encouraging. Representative classification confusion matrices corresponding to atomic patterns from Sensors  $T_{48}$ ,  $N_c$  and their relational patterns of both directions are provided below. In a confusion matrix  $C$ , element  $C_{ij}$  denotes the frequency of classifying test sample from class  $i$  as a sample from class  $j$ .

$$\mathbf{C}_{test}^{T_{48}} = \begin{pmatrix} 47 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 18 & 2 & 22 & 3 & 0 & 5 & 0 & 0 \\ 0 & 1 & 12 & 0 & 15 & 5 & 11 & 6 & 0 \\ 11 & 11 & 0 & 21 & 5 & 0 & 2 & 0 & 0 \\ 0 & 6 & 15 & 2 & 15 & 2 & 5 & 5 & 0 \\ 0 & 0 & 4 & 0 & 0 & 21 & 0 & 16 & 9 \\ 0 & 5 & 11 & 3 & 17 & 2 & 11 & 1 & 0 \\ 0 & 0 & 9 & 0 & 3 & 19 & 2 & 13 & 4 \\ 0 & 0 & 0 & 0 & 0 & 11 & 0 & 3 & 36 \end{pmatrix}$$

$$\mathbf{C}_{test}^{Nc} = \begin{pmatrix} 17 & 0 & 0 & 17 & 8 & 0 & 1 & 7 & 0 \\ 1 & 23 & 1 & 0 & 10 & 6 & 0 & 1 & 8 \\ 0 & 2 & 40 & 0 & 0 & 6 & 0 & 0 & 2 \\ 11 & 0 & 0 & 29 & 0 & 0 & 7 & 3 & 0 \\ 6 & 8 & 0 & 1 & 19 & 0 & 0 & 6 & 10 \\ 0 & 5 & 16 & 0 & 2 & 17 & 0 & 0 & 10 \\ 0 & 0 & 0 & 14 & 0 & 0 & 35 & 1 & 0 \\ 10 & 0 & 0 & 9 & 11 & 0 & 2 & 17 & 1 \\ 1 & 6 & 4 & 0 & 5 & 13 & 0 & 3 & 18 \end{pmatrix}$$

$$\mathbf{C}_{test}^{T_{48} \rightarrow Nc} = \begin{pmatrix} 47 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 46 & 1 & 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 43 & 0 & 0 & 7 & 0 & 0 & 0 \\ 1 & 0 & 0 & 44 & 4 & 0 & 1 & 0 & 0 \\ 0 & 3 & 1 & 0 & 40 & 1 & 0 & 5 & 0 \\ 0 & 0 & 2 & 0 & 0 & 39 & 0 & 0 & 9 \\ 0 & 0 & 0 & 4 & 1 & 0 & 43 & 2 & 0 \\ 0 & 0 & 0 & 0 & 7 & 2 & 1 & 36 & 4 \\ 0 & 0 & 0 & 0 & 0 & 5 & 0 & 5 & 40 \end{pmatrix}$$

$$\mathbf{C}_{test}^{Nc \rightarrow T_{48}} = \begin{pmatrix} 49 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 40 & 2 & 3 & 5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 48 & 0 & 0 & 2 & 0 & 0 & 0 \\ 6 & 1 & 0 & 40 & 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 2 & 43 & 0 & 0 & 2 & 0 \\ 0 & 0 & 3 & 0 & 0 & 45 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 & 0 & 0 & 46 & 3 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 & 40 & 7 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 46 \end{pmatrix}$$

A close observation reveals similarity of fault signatures on single sensor data for two completely different fault conditions, e.g., fault signatures of Class 4 (HPC medium fault, HPT low fault) has been confused with Class 1 (HPC low fault, HPT low fault) for both sensors  $T_{48}$  and  $Nc$  individually. However, this ambiguity can be removed by using relational pattern directed from  $T_{48}$  to  $Nc$ .

## 5. SUMMARY, CONCLUSIONS AND FUTURE WORK

This article presents a SDF-based methodology for data-driven diagnosis of component level faults in aircraft gas turbine engines via multi-sensor data interpretation and fusion. The abstract semantic representation of sensor data

in the proposed method, enables feature level fusion of heterogeneous, disparate sensors. Besides, it is shown that identification of cross-dependencies among different sensor prevents loss of significant information compared to set-theoretic information fusion methods. Also, the hierarchical architecture of this method reduces computational complexity, allowing real-time operability.

Although the method presented in the paper can perform in real time, various operating conditions need to be investigated for its on-board (in-flight) application. Apart from this important aspect, the following research areas are currently being pursued as well before testing the methodology on a real-life engine test bed.

- Development of algorithms to extract relational dependencies among three or more symbol sequences.
- Identifying the effects of lack of synchronization among sensor observations on xD-Markov machine construction.
- Comparative evaluation of semantic information fusion framework with other information fusion techniques, e.g., Dempster-Shafer or Bayesian network approaches.

## REFERENCES

- [1] O. Basir and X. Yuan, "Engine fault diagnosis based on multi-sensor information fusion using dempstershafer evidence theory," *Information Fusion*, vol. 8, pp. 379–386, 2007.
- [2] C. Romessis and K. Mathioudakis, "Bayesian network approach for gas path fault diagnosis," *J. Eng. Gas Turbines Power*, vol. 128, pp. 64–72, 2006.
- [3] S. Gupta, A. Ray, S. Sarkar, and M. Yasar, "Fault detection and isolation in aircraft gas turbine engines: Part i - underlying concept," *Proceedings of the I Mech E Part G: Journal of Aerospace Engineering*, vol. 222, no. 3, pp. 307–318, May 2008.
- [4] S. Sarkar, M. Yasar, S. Gupta, A. Ray, and K. Mukherjee, "Fault detection and isolation in aircraft gas turbine engines: Part ii - validation on a simulation test bed," *Proceedings of the I Mech E Part G: Journal of Aerospace Engineering*, vol. 222, no. 3, pp. 319–330, May 2008.
- [5] S. Sarkar, C. Rao, and A. Ray, "Statistical estimation of multiple faults in aircraft gas turbine engines," *Proceedings of the I Mech E Part G: Journal of Aerospace Engineering*, vol. 223, no. 4, pp. 415–424, 2009.
- [6] D. K. Frederick, J. A. DeCastro, and J. S. Litt, "Users guide for the commercial modular aero-propulsion system simulation (C-MAPSS)," October 2007. NASA/TM2007-215026.
- [7] D. L. Simon and S. Garg, "Optimal tuner selection for kalman filter-based aircraft engine performance estimation," *Journal of Engineering for Gas Turbines and Power*, vol. 132, no. 3, p. 031601, 2010.
- [8] E. Blasch, I. Kadar, K. Hintz, J. Biermann, C. Chong, and S. Das, "Resource Managemenet Coordination with Level 2/3 Fusion Issues and Challenges," *IEEE A & E Sytems Magazine*, pp. 32–46, March 2008.
- [9] A. Ray, "Symbolic dynamic analysis of complex systems for anomaly detection," *Signal Processing*, vol. 84, no. 7, pp. 1115–1130, 2004.
- [10] V. Rajagopalan and A. Ray, "Symbolic time series analysis via wavelet-based partitioning," *Signal Processing*, vol. 86, no. 11, pp. 3309–3320, 2006.
- [11] A. Subbu and A. Ray, "Space partitioning via hilbert transform for symbolic time series analysis," *Applied Physics Letters*, vol. 92, no. 8, pp. 084107–1 to 084107–3, February 2008.
- [12] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
- [13] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. John Wiley, New York, 2001.