# For review for presentation in ACC 2013, Washington DC Anomaly Detection in Flight Recorder Data: A Dynamic Data-driven Approach

Santanu Das<sup>†</sup>Soumalya Sarkar<sup>‡</sup>Asok Ray<sup>‡</sup>Ashok Srivastava<sup>§</sup>Donald L. Simon<sup>#</sup>santanu.das-l@nasa.govsvs5464@psu.eduaxr2@psu.eduashok.srivastava@nasa.govdonald.l.simon@nasa.gov

<sup>†</sup>UARC, NASA Ames Research Center, Moffett Field, CA 94035

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

§ NASA Ames Research Center, Moffett Field, CA 94035

# NASA Glenn, 21000 Brookpark Road Cleveland, OH 44135

Keywords: Anomaly detection; Symbolic Dynamics; Flight recorder data; Data-driven approach

Abstract—This paper presents a method of feature extraction in the context of aviation data analysis. The underlying algorithm utilizes a feature extraction algorithm called symbolic dynamic filtering (SDF) that was recently published. In SDF, time-series data are partitioned for generating symbol sequences that then construct probabilistic finite state automata (PFSA) to serve as features for pattern classification. The SDF-based algorithm of feature extraction, which enjoys both flexibility of implementation and computational efficiency, is directly applicable to detection, classification, and prediction of anomalies and faults. The results of analysis with realworld flight recorder data show that the SDF-based features can be derived at a desired level of abstraction from the information embedded in the time-series data. The performance of the proposed SDF-based feature extraction is compared with that of standard temporal feature extraction for anomaly detection. Our study on flight recorder data shows that SDFbased features can enable discovering unique anomalous flights and improve the performance of the detection algorithm. We also theoretically show that under certain conditions it may be possible to achive a better or comparable time complexity with SDF based features.

## I. INTRODUCTION

Over the last few decades, data sets have been growing at an unprecedented pace in terms of variability, velocity, and volume. Today, we are left with the challenge of dealing with these vast and heterogeneous data sources. Mining these heterogeneous resources is still a challenging task. Data mining is the art and science of analyzing a large collection of observations to extract previously known and actionable information from large data sets. The field of data mining is highly multidisciplinary and draws from fields like statistics, machine learning, pattern recognition, high-performance computing, and data visualization. The entire Data Mining and Knowledge Discovery (DMKD) framework may be customized by the requirements of the study undertaken by the user. Figure 1 shows the schematic diagram of a DMKD framework that has been designed for this research. Apart from input raw data, the four basic functionalities of this DMKD process are "data preparation module," "detection module," "knowledge discovery module," and "reporting module". The data preparation module transforms data into information and in order to do so it performs several tasks: data cleaning, normalization, feature selection, feature extraction, feature derivation, and data type segregation. The "detection module" is the heart of the DMKD process and constitutes statistical models that learn on the data. "Knowledge discovery module" discovers knowledge from the information and in many cases involve subject matter experts along with statistical indicators that provide quantitative evidences in order to characterize the performance of the entity subjected to test. The final module contains routines for graphical presentation of information in order to provide a qualitative understanding of the information contents and thus reveals the patterns, trends, relationships out of data sets.

The field of feature extraction is an important area of research in many fields including machine learning, data mining, and computer vision. This paper makes use of a feature extraction tool for anomaly detection, called symbolic dynamic filtering (SDF) [17]. Mallapragada et al. [12] used SDF as a feature extraction tool for behavior identification of mobile robots, where the performance of SDF-based feature extraction was shown to be significantly superior to that of principal component analysis (PCA) [6], based on the experimental data in a laboratory environment. In a contemporaneous paper submitted to 2013 American control Conference, Bahrampour et al. [3] have reported consistently superior performance of SDF-based feature extraction over cepstrum-based feature extraction in terms of successful detection, false alarm, and overall correct classification rates in an application of target detection and classification (e.g., monitoring of human intruders). This paper reports a novel application of symbolic dynamic filtering to extract key features from real-life Flight operations quality assurance (FOQA) data and investigates the impact of these features on the performance of anomaly detection.



Fig. 1: Data Mining and Knowledge Discovery (DMKD) Framework used in this study.

# A. Anomaly Detection Methods in FOQA Analysis

The theme of this paper is anomaly detection, also known as outlier detection. Outlier or anomaly detection refers to the task of identifying new or unknown patterns which, in many cases, are abnormal or inconsistent. The problem of outlier detection has been extensively studied using several approaches [13],[14], [15], [8].

Some algorithms that have been used extensively used for FOQA data analysis include Aviation Performance Measuring System, Morning Report, Inductive Monitoring System (IMS), SequenceMiner, Multiple Kernel Anomaly Detection (MKAD), Cluster based Anomaly Detection (ClusterAD) and iOrca.

- The Aviation Performance Measuring System (APMS) [1] was a NASA program aimed to analyze Flight Operational Quality Assurance (FOQA) data. The program identified three major goals—analyzing data beyond simply looking for exceedances of typical ranges of single parameters, focused analysis of higher-risk phases of flight, and looking for potential precursors to aviation safety incidents and accidents.
- Morning Report (MR) [21], [2] was designed for individual airlines to analyze their flight data in a manner much like APMS. The subsequent System Level Morning Report (SLMR) attempted to address the problem we described earlier of balancing between analyzing within each flight and across multiple flights. SLMR allowed users to analyze flight data in the context of individual airlines and the context of all the airlines.
- The Inductive Monitoring System (IMS) [10] is a distance based anomaly detection tool that uses an unsupervised anomaly detection algorithm that uses incremental clustering to build models of the expected operation of the system on a set of nominal data. The model can be used to test new data to determine whether an anomaly is present or not. The underlying concept states that if the system behaves similar to the normal operating modes that the data was trained on, the distance scores

will be lower than data that are generated from a system that is in an anomalous state.

- SequenceMiner [7] was developed to address the problem of detecting and describing anomalies in large sets of high dimensional symbol sequences. SequenceMiner is an unsupervised clustering algorithm that focuses on detecting sequential anomalies. SequenceMiner detects anomalies using the normalized Longest Common Subsequence (nLCS) based distance measure.
- Multiple Kernel Anomaly Detection (MKAD) [9] algorithm combine strengths of both vector space based techniques and sequential anomaly detectors like SequenceMiner into a single approach to allow for detection of a variety of anomalies from heterogeneous data sources. MKAD is a multiple kernel learning approach to incorporate more knowledge in the decision process so that one can achieve an improvement in detecting anomalies in complex heterogeneous systems that involve various data sources and data structures. MKAD is based on classical one-class SVMs [19] which is a unsupervised learning method that finds a set of outliers using a decision boundary.
- ClusterAD [11] is based on Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm that can automatically determine the number of clusters, progressively finds clusters based on a density criterion in the clouds of data and finds outliers in the feature space. ClusterAD has been tuned to find anomalies in FOQA type data sets.

Another important feature of SequenceMiner, MKAD and ClusterAD algorithm is their ability to detect anomalies across a fleet of aircraft.

In this study we use iOrca [5] as the baseline algorithm. However, for vector space-based methods like IMS or iOrca, point by point analysis of the data quickly starts to become computationally expensive. This paper combines the strengths of SDF-based feature extraction and iOrca algorithms to increase the performance of detecting a variety of anomalous conditions and to address the scalability issue under certain circumstances. The proposed DMKD framework, shown in Fig. 1, is designed with both types of analysis in mind, allowing for single flight diagnosis of subcomponents and identifying periods of anomalous flights, while also performing data compression for scalability to fleet-wide analysis. In the sequel we will demonstrate the proposed framework on aviation data using both temporal and SDF features, while maintaining the same model parameter settings.

1) Baseline Algorithm - iOrca: iOrca[5] is a scalable version of the Orca developed by Bay et al. [4]. Orca is a k-nearest neighbor based unsupervised anomaly-detection algorithm in conjunction with some efficient pruning rules. Orca has a nested loop structure to calculate pairwise distances between data points but uses a simple pruning rule to keep the algorithm's actual time complexity significantly less than the square of the number of data points. In fact, the pruning used in this algorithm helps to achieve near linear time performance with high dimensional data. This makes Orca suitable for analyzing large data sets. Orca takes a dataset and a user specified number  $T_{top}$  as input and returns the top  $T_{top}$  anomalies in the dataset by computing distances to the kth nearest neighbor or average distances to the knearest neighbors. Orca can process both continuous and binary data format. For continuous data, Orca uses Euclidean distance while Hamming distance is used for binary data points. Each data point is scored independently and therefore anomalies in the time domain are undetectable. The measure of anomalousness is the computed mean distance (or maximum distance) of individual  $x_i$  to its k-nearest neighbors. This is repeated for each point  $x_i$  in the dataset and the rankings are decided. In [5], the authors introduced a novel indexing strategy and an early termination criterion to make Orca scalable to extremely large data sets. The indexed version of Orca is known as iOrca.

### II. SYMBOLIC DYNAMIC FILTERING (SDF)

Statistical information embedded in time series of signals (e.g., avionic, flight, or structural vibration data) can be detected and identified by symbolic dynamic filtering (SDF) [17]. Symbolic dynamic filtering has been used for anomaly detection in diverse aerospace and electromechanical applications (e.g., [18]). This paper makes use of SDF for anomaly detection for FOQA data under different operating conditions. Although the details on the theory and construction of SDF have been reported in recent publications (e.g., [17][16][20]), the underlying concept of SDF is succinctly presented below.

Symbolic feature extraction from time series data is posed as a two-time-scale problem. Over the span of a given time series, dynamic behavior of the anomaly evolution is assumed to remain statistically invariant, i.e., the process is assumed to be quasi-stationary in the fast scale. The slow scale is related to the time span over which parametric or non-parametric changes may occur and exhibit nonstationary dynamics that can be associated with the evolving dynamics of the anomalous behavior. In general, a long time span in the fast scale is a tiny (i.e. several orders of magnitude smaller) interval in the slow scale. Figure 2 illustrates the concept of two time scales. It is expected that the features extracted from the fast-scale data will depict statistical changes, if any, between two different slow-scale epochs if the underlying system has undergone a change in its statistical behavior. The method of extracting features from stationary time series data is comprised of the following two major steps.



Fig. 2: Concept of Two Time Scales.

Step 1 Data Partitioning and Symbol Generation: Sensor time series data, generated from a physical system or its dynamical model, are collected at a slow-scale epoch denoted as **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  $Q \subseteq \mathbb{R}^{n \times N}$ , 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,  $\{\mathbf{q}\} \in Q$  denotes a time series at the slow-scale epoch of data collection.

Encoding of  $\Omega$  is accomplished by introducing a partition  $\mathbb{B} \triangleq \{B_0, ..., B_{(|\Sigma|-1)}\}$  consisting of  $|\Sigma|$  mutually exclusive (i.e.,  $B_j \cap B_k = \emptyset \ \forall j \neq k$ ), and exhaustive (i.e.,  $\bigcup_{j=0}^{|\Sigma|-1} B_j = \Omega$ ) cells, where each cell is labeled by symbols  $\sigma_j \in \Sigma$  and  $\Sigma = \{\sigma_0, ..., \sigma_{|\Sigma|-1}\}$  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  $\sigma_j \ \forall j = 0, 1, ..., |\Sigma| - 1$ . 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  $\Sigma$ .

Step 2 Construction of PFSA for Feature Extraction: A probabilistic finite state automaton (PFSA) is constructed from the *training* symbol sequence {**s**}. Subsequently, *test* or *operational* symbol sequences are run through the same (irreducible) PFSA structure to generate the respective features. In this paper, the PFSA is constructed in the framework of a D-Markov machine [17], where the morph

matrix  $\pi$  is obtained by frequency counting of symbols from each state of the D-Markov machine and an element  $\pi_{ij}$  of  $\pi$ represents the probability of generating the symbol  $\sigma_i$  at the state  $q_i$ . Using the morph matrix  $\pi$  and the state transition map  $\delta$  of the *PFSA* [17], a state transition probability matrix  $\Pi$  is constructed, where an element  $\Pi_{ij}$  of  $\Pi$  represents the probability of transition from the state  $q_i$  to the state  $q_i$  in a single transition. The unity-sum-normalized (and strictly positive) left eigenvector of the (irreducible)  $\Pi$  matrix corresponding to the unity eigenvalue is called the stationary state probability vector p. While both  $\pi$  and p are viable candidates for feature representation, the morph matrix  $\pi$ provides a larger amount of statistical information at the expense of increased computational complexity. In this paper, the morph matrix  $\pi$  is used as a feature to represent the statistical behavior embedded in time series data.

# III. EXPERIMENTAL RESULTS AND DISCUSSIONS

Aviation data have many aspects that create natural sources of heterogeneity like origin or destination airports, city pair routes, tail numbers, aircraft models, as well as seasonal aspects such as time of the year. A real-world data set was chosen from a commercial passenger jet airline landing at a single destination airport resulting in approximately 25519 flights. All aircraft analyzed are of the same fleet and type. Even within a flight there exist several phases such as takeoffs, landings and cruise. Flight characteristics vary significantly by phase of flight. As a pre-processing step, for each flight the airborne part of a flight starting from 6 nautical miles (nmi) from touch down to touch down is chosen in order to obtain segments with comparable characteristics. Each flight consists of 367 parameters sampled at 1 Hz with the average flight length approximately 2.5 hours. However, we used a subset of the flight parameters, as seen in Table I, based on a correlation study and domain expert's feedback in order to focus on detecting operational problems. The correlation study helped in identifying some of the redundant parameters. Flight recorded data are often contaminated with missing data, out of bounds variables, noisy recordings, amplitude spikes etc. caused by sensor malfunctions or recording medium errors. We used a set of data quality filters to clean the data.

In addition to the observed parameters we construct some derived parameters depending on our study and expert inputs. Derived parameters may help to understand the hidden state of the aircraft based on some set of observed parameters *e.g.* estimated aircraft speed margin above stall speed based on flap settings, gross weight, and velocity. Sometimes these parameters can take into account some of the physical laws of the application domain and can help tracking particular events. The flap position parameter was continuously recorded, however it is categorical. Using information from the domain expert in conjunction with the statistics from the data, the flap parameter, which is categorical in nature, was decomposed into 4 binary state variables. The mapping of

Attribute Type	Variable Names				
Diamata					
Discrete	Autopilot and all Autopilot related modes,				
	Auto-Flight Director, Glide Slope, Stall In-				
	dicator, throttle, Ground Proximity Warning				
	System, Altitude Mode, Flare Mode, Flap				
	Positions (derived parameter), Flight Path				
	Angle Mode etc.				
Continuous	Altitude, Target Air Speed, Computed Air				
	Speed, Engine-related Measures, Pitch An-				
	gle, Roll Angle, Rudder Position, Angle of				
	Attack, Aileron Position, Stabilizer Position,				
	Aircraft Gross Weight, Latitude, Longitude				
	and Normal Accelerations, Derived parame-				
	ters like Above Stall Speed, Vertical Speed				
	etc.				

TABLE II: Relation of Derived Flap Parameters with Flap Positions

	Flap0	Flap1	Flap2	FullFlaps
Flap Positions (in degree)	10	15	20	40

the original flap parameter to the binary state variables is shown below in Table II.

Since the flight data used in the analysis include the airborne part of the landing phase to a fixed destination airport, the lengths of each flight differ in length but with small variations. We chose to resample all flights so that each flight had a duration equivalent to the shortest duration flight. The information loss due to this compression is negligible. For SDF based feature extraction, we chose uniform partition scheme assuming a window size of 9 and generated the PFSA features. The binary variables were appropriately resampled to match the length of the continuous parameters.

In the first experiment, the z-score normalized temporal features of all the flights were concatenated resulting a 51 dimensional matrix with 3,011,242 tuples. In the second experiment, the SDF based features of all the flights were concatenated resulting matrix having 2,551,900 tuples with the same number of dimensions. The discrete inputs were in standard binary format. These matrices served as inputs to the iOrca algorithm.

### A. Summary of Analysis

When using iOrca to analyze the flights, the number of nearest neighbors was set to the default value (k=5). We asked iOrca to report top 1000 anomalous tuples ( $T_{top}$ ). These model parameter settings were kept consistent for both the experiments. After each experiment, a simple post-processing method was used to map the scores of anomalous tuples to respective flights, aggregate the scores for individual flights and finally rank the anomalous flights based on their scores. The flights without any anomalous tuples have zero scores. This resulted a total of 356 and 820 anomalous flights using temporal and SDF based features respectively. Since the number of anomalous flights detected by iOrca is controlled by  $T_{top}$ , one observation is that for a given  $T_{top}$ , iOrca with SDF reported more anomalies as compared to the run with temporal features. This may be due to the



Fig. 3: Distribution of Exceedances for the Anomalous Flights Detected by iOrca using temporal and SDF based Features.

fact that SDF based features are more unique in representing anomalous states of individual flights. In addition SDF based features extraction technique is sensitive to signal distortions and at the same time robust to measurement noise and spurious signals. It is also adaptable to low-resolution sensing due to the coarse graining in space partitions.

For each experiment, the anomalies were ranked based on their scores. It means that the flight on the top of this list is the most anomalous example compared to the rest of the flights and has the highest anomaly score. For the rest of the paper we will address the sorted list of anomalies obtained from iOrca using temporal features and SDF based features as  $List_{356}$  and  $List_{820}$  respectively. At this point we would like to compare the agreement between two experiments based on the rank information of the anomalous flights. Starting from the top of each list (say  $List_{356}$ ) we intended to find if the candidate flights are present in the other list (say,  $List_{820}$ ) and computed the percentages of overlapping flights. Based on this analysis, we observed both the features are sensitive to the first few top ranking anomalies. However the overall trend shows that the agreement between the two features monotonically decreases as the number of anomalous flight increases. This means that the two features have some unique properties and hence sensitive to different kinds of abnormalities.

In the airline industry, the most widespread method for detecting operationally significant anomalies in flight recorder data is "exceedance detection technique" where the threshold exceedances are defined by domain experts. This method has been in use for as long as the FOQA program has been in existence and has provided analysts with valuable results for "known anomalies." Normally, airlines use three levels to detect the "exceedances". Level 1 indicates minor variation

from performance target, while Level 3 indicates the severest deviation from the target value. Because Level 2 and 3 events are the issues and problems that interest the airlines the most, we exclude Level 1 events for this study. We have seen that almost every flight comes up with one or multiple Level 1 exceedances. Therefore we have used exceedance detection at Level 2 and 3 as the baseline to compare the findings of iOrca using both features. Table III shows the number of flights with exceedance Level 2 and 3 detected by iOrca with temporal features and the number of flights with exceedance Level 2 and 3 detected by iOrca with SDF features. It should be noted that iOrca with SDF features has detected more flights with exceedance Level 2 and 3. This result may be expected since iOrca with SDF features has reported more flights but the important point here is that SDF features has icreased the capability of iOrca to detect more flights with exceedance Level 2 and 3 in a single run, for a fixed  $T_{top}$ .

From the entire event list defined by the FOQA program, a total of 92 exceedance events correspond to the landing phase which have been considered in this study. From the unique anomalies reported by  $List_{356}$  and  $List_{820}$ , we were able to determine that iOrca identified 64 exceedance events out of 92. There could be several explanations on why some of the other exceedance events were not detected. For example the ability of the algorithm to process information extracted from different data formats or the sensitivities of the features to those event types or even the absence of some information (like physical laws) related to specific events, in FOQA data. Figure 3 shows the distribution of the exceedance events (Level 2 and 3) detected by iOrca. The exceedance events corresponding to the anomalous flights detected by iOrca using different features were compared to TABLE III: Summary of the Exceedance based Detection for the Flights detected by iOrca using different Features.

	Exceedance Severity			
	DetectedOutliers	Level 2	Level3	
temporal Feature	356	78	34	
SDF based features	820	134	42	

assess the commonalities and differences. In this discussion we focus on the commonly detected events. It can be seen (Fig. 3) that some exceedance events frequently detected by iOrca with temporal feature are "Speed High in Approach (at 500ft)," "Banking High in Approach (below 100ft)," "Deviation above Glideslope (Above 1000ft)," "Tail Strike Risk at Landing" etc. An interesting observation here is that SDF features assist iOrca to continue detecting more anomalous flights with severity Level 2 and 3 in those exceedance categories. Most probably, some of these anomalies will be reported by iOrca using temporal features for higher  $T_{top}$  values which only comes at the cost of computational expense and resources. Further investigations are required to evaluate the operational significance of the flights detected by iOrca and is beyond the scope of this paper.

1) Scalability Issues: The time complexity of the SDF algorithm depends on three parameters, number of symbols i.e. alphabet size (m), depth (D) and total number of data points or observations (N). Now depending on the selection of the feature type the complexity of the D-Markov algorithm is determined. This complexity turns to be  $O(m^D)$  + O(N) when the probability vector is used as a feature and  $O(m^{D+1}) + O(N)$  when the morph matrix is used as a feature. In this study we opted for the later feature where D was set to one and so the complexity is  $O(m^2) + O(N)$ . For D > 1, the above estimate is conservative because the number of states should be much less than  $m^D$  after state merging. Orca has been demonstrated experimentally to have a running time that is, on average, slightly greater than linear in the number of data points submitted, although its worstcase running time is  $O(N^2)$  i.e. quadratic in the number of data points (N). So the key question is whether  $m^2 < N$  that is, whether the cardinality of the PFSA feature vector is smaller than the length of the raw time series. In that case, the overall complexity of the feature extraction algorithm is of O(N). From a scalability point of view, the choice of m is critical. When  $m^4 \leq N$ , the worst-case running time for Orca with SDF based features is O(N) which is much better than  $O(N^2)$  when using temporal feature.

### **IV. CONCLUSIONS**

In this article, we have conducted a comparative study on the outcome of iOrca with two different features in context to anomaly detection on flight recorder data. The feature extraction concept for anomaly detection outlined in this paper has been reported recently in several applications but, to the best of our knowledge, this is the first attempt to explore the applicability of symbolic dynamic filtering (SDF)-based features in analyzing real-world aviation data.

The SDF-based feature extraction technique is sensitive to signal distortions, because SDF is capable of detecting the changes in statistical characteristics of the signal. At the same time, SDF is robust to measurement noise and spurious signals, because it tends to filter out these effects in its finitestate automaton structure. However, this is not to say that the SDF-based feature extraction algorithm will help iOrca to find all possible anomalies in the flight recorder data, but rather that these features are robust enough to find a variety of other anomalies in addition to a significant overlap with the state-of-the-art method using temporal features. Another advantage of SDF-based feature extraction is its adaptability to low-resolution sensing due to the coarse graining in space partitions. The reduced cardinality of the SDF feature set and the method's flexibility of implementation make it a potential solution to scale-up detection, classification and prediction problems for large avionic data sets. In the future, we intend to explore both the operational significance of some of the detected anomalies reported in this paper and theoretical research for enhancement of the algorithms.

### ACKNOWLEDGMENT

The authors would like to thank Robert Lawrence, Dr. Michael Feary, Bryan Matthews and Dr. Nikunj Oza for insightful discussions and suggestions during this research.

#### REFERENCES

- I. C. Statler, The Aviation System Monitoring and Modeling (ASMM) Project: A Documentation of its History and Accomplishments: 1999-2005, (2007), Technical Report: NASA/TP-2007-214556.
- [2] Amidan, B.G., Swickard, A.R., Allen, R.E. and Ferryman T. A., *Identifying In-Close-Approach-Changes in Air Traffic Control (ATC) Data.*, (2002), PNWD-3210. Battelle Pacific Northwest National Laboratory: Richland, WA.
- [3] S. Bahrampour, A. Ray, K. Mukherjee, T. Damarla, and N. Nasrabadi, Symbolic dynamics-based feature extraction for target detection and classification., Proceedings of 2013 American Control Conference, Washington, D.C., USA, 2013, p. companion paper submitted for review for publication.
- [4] S. D. Bay and M. Schwabacher, *Mining Distance-based Outliers in Near Linear Time with Randomization and a Simple Pruning Rule*, Proceedings of KDD'03, 2003, pp. 29–38.
- [5] Kanishka Bhaduri, Bryan L. Matthews, and Chris R. Giannella, *Algorithms for speeding up distance-based outlier detection*, Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (New York, NY, USA), KDD '11, ACM, 2011, pp. 859–867.
- [6] C.M. Bishop, Pattern recognition and machine learning, Springer, 2006.
- [7] Suratna Budalakoti, Ashok N. Srivastava, and Matthew E. Otey, Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety, IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 39 (2008), no. 1, 101–113.
- [8] Banerjee A. Chandola V. and Kumar V., Anomaly detection: A survey, ACM Comput. Surv. 41 (2009), 1–58.
- [9] Santanu Das, Bryan L. Matthews, Ashok N. Srivastava, and Nikunj C. Oza, *Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study*, Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (New York, NY, USA), KDD '10, ACM, 2010, pp. 47–56.
- [10] David L. Iverson, *Inductive system health monitoring*, Proceedings of the 2004 International Conference on Artificial Intelligence (IC-AI'04), CSREA Press, Las Vegas, NV (2004).

- [11] Lishuai Li, Maxime Gariel, R. John Hansman, and Rafael Palacios, Anomaly detection in onboard-recorded flight data using cluster analysis, Digital Avionics Systems Conference (DASC), 2011 IEEE/AIAA 30th, oct. 2011, pp. 4A4–1 –4A4–11.
- [12] G. Mallapragada, A. Ray, and X. Jin, Symbolic dynamic filtering and language measure for behavior identification of mobile robots, IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics 42 (2012), 647–659.
- [13] Markos Markou and Sameer Singh, Novelty detection: A review part 1: Statistical approaches, Signal Processing 83 (2003), 2003.
- [14] \_\_\_\_\_, Novelty detection: A review part 2: Neural network based approaches, Signal Processing 83 (2003), 2499–2521.
- [15] John A. Quinn and Christopher K. I. Williams, *Known unknowns: Novelty detection in condition monitoring*, Proceedings of 3 rd Iberian Conference on Pattern Recognition and Analysis, Lecture Notes in Computer Science 4477. Springer-Verlag, Springer LNCS, 2007, pp. 1–6.
- [16] V. Rajagopalan and A. Ray, Symbolic time series analysis via wavelet-

based partitioning, Signal Processing 86 (2006), no. 11, 3309–3320.
[17] A. Ray, Symbolic dynamic analysis of complex systems for anomaly detection, Signal Processing 84 (2004), no. 7, 1115–1130.

- [18] S. Sarkar, X. Jin, and A. Ray, *Data-driven fault detection in aircraft engines with noisy sensor measurements*, Journal of Engineering for Gas Turbines and Power **133** (August 2011), 081602 (10 pages).
- [19] Bernhard Schölkopf, John C. Platt, John C. Shawe-Taylor, Alex J. Smola, and Robert C. Williamson, *Estimating the support of a highdimensional distribution*, Neural Computation 13 (2001), no. 7, 1443– 1471.
- [20] A. Subbu and A. Ray, Space partitioning via hilbert transform for symbolic time series analysis, Applied Physics Letters 92 (February 2008), 084107–1.
- [21] Thomas R. Chidester, Understanding Normal and Atypical Operations through Analysis of Flight Data, In Proceedings of the 12th International Symposium on Aviation Psychology (2003), Dayton, Ohio.