An Approach To Mode and Anomaly Detection with Spacecraft Telemetry Data

Gautam Biswas[1], Hamed Khorasgani [2], Gerald Stanje[3], Abhishek Dubey[4], Somnath Deb[5], and Sudipto Ghoshal[6]

[1,2,3,4] *EECS Department/Institute for Software Integrated Systems, Vanderbilt Univ., USA,*
gautam.biswas@vanderbilt.edu
hamed.g.khorasgani@vanderbilt.edu
gerald.stanje@vanderbilt.edu
abhishek.dubey@Vanderbilt.edu

[5,6] *Qualtech Systems, Inc. (QSI), USA,*
deb@teamqsi.com
sudipto@teamqsi.com

ABSTRACT

This paper discusses a mixed method that combines unsupervised learning methods and human expert input for analyzing telemetry data from long-duration robotic space missions. Our goal is to develop more automated methods for detecting anomalies in time series data. Once anomalies are identified using unsupervised learning methods we use feature selection methods followed by expert input to derive the knowledge required for building on-line detectors. These detectors can be used in later phases of the current mission and in future missions for improving operations and overall safety of the mission. Whereas the primary focus in this paper is on developing data-driven anomaly detection methods, we also present a computational platform for data mining and analytics that can operate on historical data offline, as well as incoming telemetry data on-line.

1. INTRODUCTION

As engineered systems have become more complex, self-monitoring, self-diagnosis, and adaptability to maintain operability and safety have become focus areas for research and development. Typical goals of such self-diagnosis approaches are the detection and isolation of faults, identifying and analyzing the effects of degradation and wear, and providing fault-tolerant and fault-adaptive control Blanke & Schröder (2006); Chen & Patton (2012); Isermann (2005); Ji et al. (2003); Noura et al. (2009). The majority of projects dealing with monitoring and diagnosis applications rely on models created using physical principles or by human experts. However, these models are not always available, and are often incomplete, and sometimes even erroneous. Moreover, it is hard to maintain the accuracy of these models during a system's life-cycle.

More recently, data-driven alternatives have emerged that exploit the large amounts of operational data collected from systems to better understand system operations under nominal and faulty conditions Qin (2012); Yin et al. (2014). The longer-term goal is to develop Cyber Physical Systems (CPSs) Lee (2008); Marwedel (2010); Niggemann et al. (2015) that can monitor their own behavior, recognize unusual situations, and inform operators, who can then modify system operations to ensure safety and ability to complete a mission. In some situations, this information can also help to plan maintenance tasks. In the longer term, systems experts and engineers can use the information gleaned from this data to update operational procedures, increase autonomy of the system, and even redesign future versions of the system.

In this paper, we take on the challenges of developing a data-driven scheme for anomaly detection. As a case study, we analyze telemetry data that was generated by NASA's Lunar Atmosphere and Dust Environment Explorer (LADEE) spacecraft[1], a robotic mission that orbited the moon to gather detailed information about the structure and composition of the thin lunar atmosphere, and determine whether dust is lofted into the lunar sky Hine et al. (2010). The LADEE system block diagram, shown in Figure 1, shows the four primary subsystems of the spacecraft: (1) the Integrated Avionics system, (2) the Propulsion system, (3) the Attitude Control system (ACS)), and (4) the Electrical Power Subsystem (EPS).

[1]see `https://www.nasa.gov/mission_pages/ladee/main/index.html`
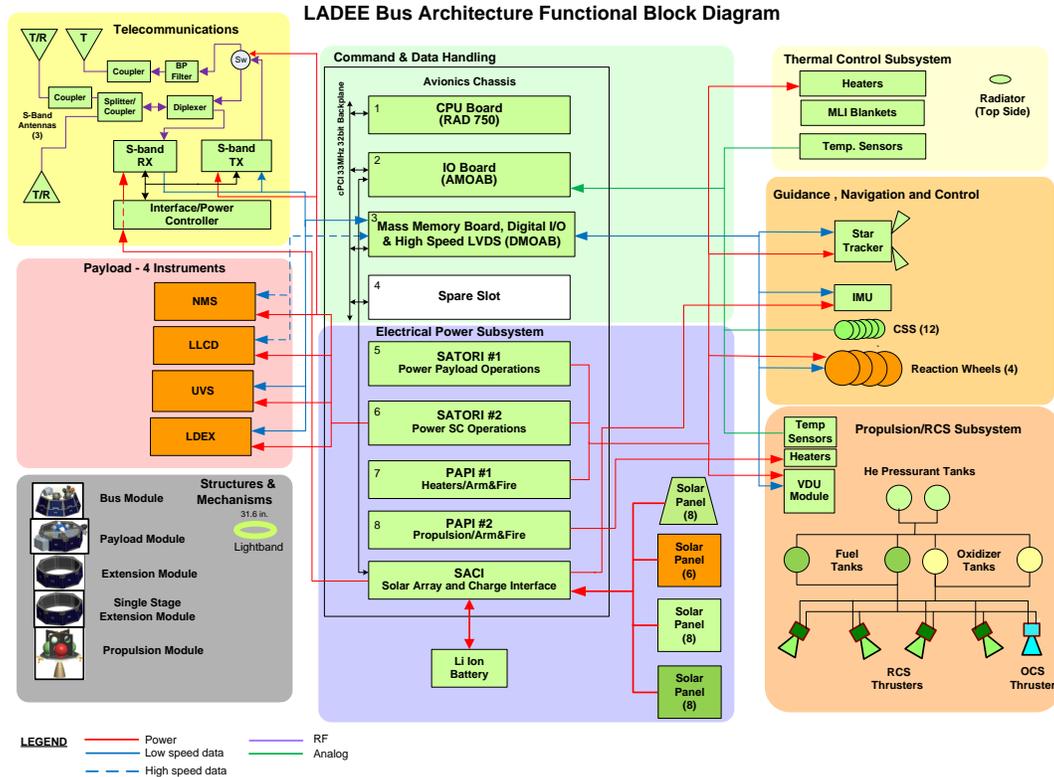
Figure 1. Functional block diagram of the LADEE spacecraft (image credit: NASA/ARC) `https://directory.eoportal.org/web/eoportal/satellite-missions/l/ladee`

Using the lessons learned from this case study, our overall goal is to develop a general data-driven monitoring approach for telemetry (i.e., streaming time series) data for purposes of health monitoring, which includes fault and anomaly detection, prognosis, and performance analysis of the monitored system.

Our primary focus in this paper is on developing unsupervised methods for data-driven anomaly detection in complex systems. However, we want the approach to be viable for long-duration space missions, which currently take place at infrequent intervals. For these one-of missions that often operate in environments that are not completely known, it makes sense that we learn about system operations and anomalies that occur by collecting and analyzing data during the mission, and then using the knowledge gained to develop fault detectors and isolation mechanisms that make it easier to keep the system operational and safe as the mission progresses. To enable this approach, we are developing a platform-based computational architecture that supports data collection and the setting up of data repositories that support analyses at multiple time scales. In addition, we demonstrate how the results of our anomaly detection approach can be used for designing online fault detectors for system health monitoring.

The rest of this paper is organized as follows. Section 2 pro-

vides an overview of the data-driven anomaly detection problem, and briefly reviews related work in this area. Section 3 formally describes our approach to anomaly detection. Section 4 shows an application of our methodology to telemetry data from the Electric Power System (EPS) of the LADEE spacecraft. Using examples, we illustrate the intertwining of the mode and anomaly detection problem. Section 5 discusses our approach to constructing the fault detection monitors given the features of the anomalies that we learn from our data-driven analyses. Section 6 then discusses our proposed computational platform architecture for implementing the anomaly detection and health monitoring schemes. Finally, Section 7 presents a discussion and conclusion based on the results of our case study, and briefly discusses how we will extend this approach in future work.

2. BACKGROUND

Anomaly detection refers to the problem of finding patterns in operational data that do not conform to expected systems behavior Chandola et al. (2009). These patterns, typically called anomalies or outliers, correspond to single data points or a small group of data points that appear to be sufficiently different from the rest of the data that make up the operational behaviors of the system. Traditional approaches to anomaly

or fault detection typically rely on a model that defines nominal behavior of a system, or on human expertise that characterizes the parameters or thresholds that separate nominal from anomalous behavior De Kleer & Williams (1987); Isermann (2005); Venkatasubramanian et al. (2003). However, in many situations, sufficiently accurate and complete models of the system may not be available. This could lead to misclassifying faults or missing faulty behavior Venkatasubramanian et al. (2003). In some situations, faulty and anomalous situations may be unknown because of a lack of sufficient experience in operating the system Mack (2013). In such situations, data-driven approaches that lead to discovering anomalies become the key to protecting system safety and integrity Mack et al. (2016, in press).

Some anomaly detection approaches are provided with or discover regions that correspond to nominal behavior. Then they apply similarity-based measures to label data points that are not *close to* the nominal regions as *anomalies* Chandola et al. (2009). Several factors can make this task challenging. For example, defining every possible region of nominal behavior for a complex system that has many operating modes may be difficult. The similarity measure used may accentuate certain feature differences more than others. Furthermore, as systems operate under different environmental and operational conditions, and the systems age, their nominal behaviors may keep drifting and evolving, and current nominal behavior may not be indicative of future nominal behaviors. And last, training data labeled as nominal and anomalous may be hard to come by. Further, even if the data is labeled, noise and corruption may distort the differences between nominal and anomalous behavior.

General approaches to exploring a feature space for identifying anomalous instances have employed learning algorithms that require highly tunable global models and error minimization procedures. Such methods include least-squares regression Bishop (2001) to derive discriminative models from data. This leads to robust algorithms that can detect a number of additive faults through the use of receiver operating characteristic curves plotted to tune the detection algorithm and set the false alarm rates Chu et al. (2010). However, these methods typically require large amounts of real and simulated data to derive general and robust solutions. Further, these methods correspond to supervised approaches, since practitioners must understand the data and the results of experiments on the model (accuracy and false positives) in order to tune the system.

The domain of aviation flight data has produced a number of techniques for discovering anomalies. These include SequenceMiner Budalakoti et al. (2009), Orca Bay & Schwabacher (2003), The Inductive Monitoring System Iverson (2004), and Morning Report Chidester (2003). These methods are built with varying amounts of data, and are com-

putationally expensive. For example, Morning Report, which was built to be run overnight on the previous days flight data to generate a report to be examined in the morning. SequenceMiner focuses on clustering methods for exploring a set of instances by reducing the features signals using a metric for measuring common sequences known as the normalized longest common subsequence Budalakoti et al. (2006). The use of this metric retains the original feature semantics, making it easier to characterize and interpret an anomaly in the original feature space.

Orca uses a scalable k-nearest neighbor approach to detect anomalies in data with continuous and discrete features. Since each data point is a sample in time and treated as independent by the algorithm, Orca struggles to detect anomalies with temporal signatures. On the other hand, the Inductive Monitoring System is a distance based anomaly detection method that focuses on continuous parameters. The method uses incremental cluster analysis to build models of the expected operations of the system, but it also does not consider temporal patterns in the data. Morning Report builds a statistical signature across each feature of a sample to reduce it to a smaller dimension. This is then used with distance metrics, such as the Mahalalanobis distance Duda et al. (2012) to find flights that are outliers from the majority of the data points. SequenceMiner and Morning Report are designed to interact with temporal signals in the data. These methods make assumptions, such as SequenceMiner requires a symbolic transformation and Morning Report requires a pass from another algorithm through the original data to help an expert characterize found anomalies.

More recent approaches that have produced good results in anomaly detection include an algorithm that combines Principal Component Analysis (PCA) and density-based clustering, DBSCAN Ester et al. (1996); Li et al. (2011). The approach uses PCA to project features in higher dimensional space to a lower dimensional space, and then applies DBSCAN to cluster the data in the lower dimensional space. Density-based clustering is robust to noise in the data, it requires little domain knowledge to define the size of a cluster, clusters can be of arbitrary shape, and outliers are easy to find. The problem with this method is temporal information is lost in the unrolling process, and the transformed space makes interpretation of the anomalies a much harder task.

Another recent approach is the semi-supervised Multiple Kernel Anomaly Detection (MKAD) method Das et al. (2010, 2011). Like SeqenceMiner, it first preprocesses all continuous time series data into symbolic sequences, and then applies a similarity measure between pairs of samples or data points. The pairwise comparisons are organized for learning by building two kernels that combine the feature streams for either continuous or discrete values. The kernel method for both types is based on the normalized Longest Common Sub-

1- Preprocessing and Feature Extraction 2- Clustering
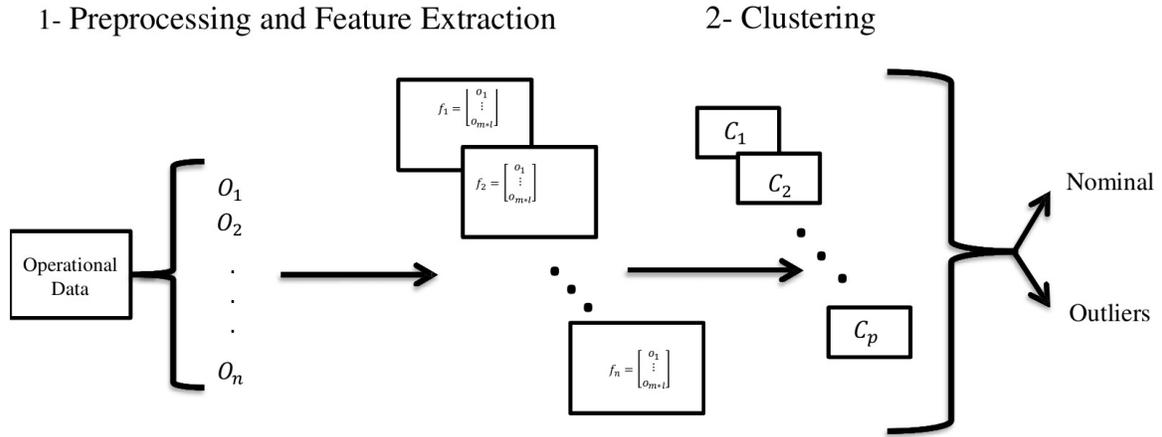
Figure 2. Unsupervised learning method to distinguish outlier behavior

sequence Budalakoti et al. (2009), the same metric used in SequenceMiner for measuring common sequences. The kernel is built for a one-class SVM classifier Rätsch et al. (2000). This method for isolating anomalies attempts to exploit common sequential information for two samples represented as a single value. Analysis of flagged anomalies is examined post-SVM, since the SVM model based on kernel methods is difficult to interpret.

3. DATA DRIVEN ANOMALY DETECTION

When dealing with one-of space missions, one may not have access to a lot of historical data on spacecraft operations from previous missions to characterize faults and errors, that may form the basis for detecting and analyzing faults during the current mission. Moreover, for long duration spacecraft missions, the spacecraft may operate in multiple modes linked to maneuvering the spacecraft and initiating a variety of science experiments.

We have developed a multi-step unsupervised learning method to distinguish outlier behaviors and link them to anomalies or faults. Figure 2 illustrates this process. Typically, a large majority of the time segments of the telemetry data will represent nominal operations of the spacecraft, but a small subset may represent anomalous and faulty behaviors. We hypothesize that the clusters or groups that contain a large number of the time segments represent nominal operations, whereas outliers (single time segments) and smaller groups may represent anomalous situations. In previous work, researchers have developed classifier or supervised methods for characterizing known faults (e.g., Hodge & Austin (2004); Mack et al. (2016, in press); Yin et al. (2014)) and semi-supervised and unsupervised methods for discovering and characterizing unknown faults and anomalies (e.g., Chandola et al. (2009); Ester et al. (1996); Das et al. (2010); Li et al. (2011)).

We propose a mixed method that combine unsupervised learning and expert analysis for anomaly detection in robotic space missions (see Figure 2). Together Figures 2 and 3 describe a four step approach: (1) data curation and preprocessing; (2) unsupervised learning using a hierarchical clustering approach; (3) identifying outlier groups and deriving the significant features that characterize each outlier group from the nominal: and (4) expert characterization of the anomalous groups. We describe the first three steps in greater detail below. Step 4 is discussed in the Case Study section 4.

3.1. Pre-processing and Feature Extraction

In this work, anomaly detection is applied to telemetry data streamed to earth stations from different subsystems of the spacecraft. As a first pre-processing step, we derive a set of objects from a curated version of the time series data. Each object is defined by a set of signals, and each signal is extracted from a longer time series signal representing the variable waveform over the entire mission. Our approach divides the time series representing the entire mission trajectory into segments, i.e., $O = \{O_1, O_2, ..., O_n\}$, and each segment represents an object of interest on the mission time line. The time interval width (the window size) is derived empirically: to reduce the number of data objects to a reasonable number, but not to make the size of the interval so large to dilute the temporal signature of anomalies. Not using small window sizes also helps to mitigate the effects of noise in the telemetry waveforms.

Since measurement sampling rates vary, aligning the data across measurements can be a complex task Mack et al. (2016, in press). We apply *data alignment* techniques to simplify subsequent analyses. We start with each time series waveforms that captures the relevant aspects of system operations represented as a variable; therefore, each data object

3- Extracting significant features and Anomaly Detection
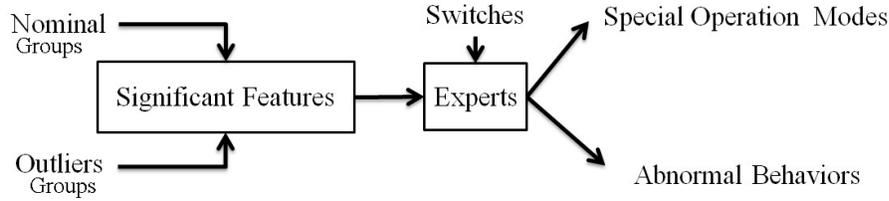
Figure 3. Anomaly detection process
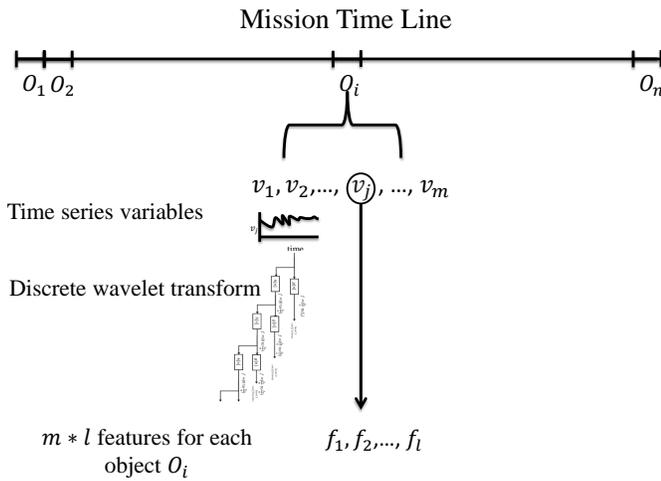
Figure 4. Data pre-processing and feature extraction

Figure 5. Discrete wavelet transform

is represented by a set of variables, $V = \{v_1, v_2, ..., v_m\}$, and each variable is a time series made up of k_j samples, $1 \leq j \leq m$. For example, in this work, our focus is on the power generation and distribution systems of the LADEE spacecraft, so we select variables related to the solar array panels, the battery, switches, and the electrical loads in the spacecraft.

The next step *feature reduction* converts each time series signal to a set of discrete features whose values are derived by applying the wavelet transform Burrus et al. (1997) to continuous signal. The wavelet transform captures the time-frequency characteristics of signal waveforms, and, in this process it can also be used to capture the frequency characteristics of the signal at different time intervals in the signal. Figure 4 illustrates the feature extraction process on the time line for each signal segment, while the feature extraction process that we describe in greater next, is illustrated in Figure 5.

We employ the discrete Haar wavelet transform (DWT) Strang (1993) to extract the time-frequency characteristics of the signals at specific intervals, and,

in this process, compress the continuous signal waveforms. Computing the Haar wavelet coefficients is equivalent to passing the continuous signal through a series of shifted and cascaded low- and high-pass filters that decomposes the signal into high and low frequency bands, $g[n]$ and $h[n]$, respectively, which are then down-sampled to capture the local time-frequency characteristics of the signal. Figure 5 shows the first three levels of the computation. The computational scheme requires the number of discrete samples in the signal must be a power of 2. In our work, we selected the first coefficient at each level as features that define the signal. Therefore, the set of features extracted for each object, O_i, is represented as a vector $f_i \in R^{m*l}$ in Figure 2, where m represents the number of time series signals, and l is the number of levels extracted for each signal by the Haar transformation. Therefore, each data object was represented by a set of $m * l$ features.

3.2. Clustering the data objects

For unsupervised learning in step 2, we have applied a hierarchical clustering algorithm Jain & Dubes (1988) (see Algorithm 1). First, we adopt the Euclidean distance as the metric to compute the dissimilarity matrix $D_{nn} = dist(O_i, O_j), 1 \leq i, j \leq n$. For clustering, we run a generic UPGMA (Unweighted Pair Group Method with Arithmetic

Mean), agglomerative (bottom-up) hierarchical clustering algorithm Day & Edelsbrunner (1984), and represent the order of cluster formation as a dendrogram. Consider two clusters X and Y, where X and Y have $|X|$, and $|Y|$ number of objects, respectively. UPGMA defines the mean distance between the clusters as the mean distance between the objects in X and Y:

$$dist(X, Y) = \frac{1}{|X||Y|} \sum_{O_i \in X} \sum_{O_j \in Y} dist(O_i, O_j). \quad (1)$$

Note that X and Y may be singletons, i.e., clusters with one data object. At each step, two existing clusters that are the closest to each other are merged into a higher-level cluster. To achieve computational efficiency in the next step of merging, the algorithm calculates the distance between the new cluster $X \cup Y$ and all other clusters Z, using the proportional averaging of $dist(X, Z)$ and $dist(Y, Z)$:

$$dist(X \cup Y, Z) = \frac{|X|dist(X, Z) + |Y|dist(Y, Z)}{|X| + |Y|}. \quad (2)$$

The algorithm saves the distances between merged clusters in a distance vector, d_v. One of the advantages of generating dendrograms by hierarchical clustering is that we can apply a number of heuristic methods to choose the level at which to cut the dendrogram, and, in this process establish the number of clusters or groups in the data set. Several approaches have been proposed for determining the number of clusters in a data set (e.g., Milligan & Cooper (1985); Yan (2005)). Typically these methods use metrics that are derived from within cluster distances Hartigan (1975); Hubert & Levin (1976), the ratio of within cluster distances to between cluster distances Caliski & Harabasz (1974), the ratio of within cluster to between cluster variances Fukunaga & Koontz (1970), or from information theory Sugar & James (2011) to establish the number of clusters in a data set.

In this paper, we select the number of groups based on a metric is derived from the distances between successive cluster formations in the dendrogram. The distance level ($y-$axis on the dendrogram) at which the clusters are partitioned is defined by a distance threshold (say, d_t) to define a distinct grouping of clusters, i.e, X and Y are merged $iff\ dist(X, Y) \leq d_t$. Therefore, by increasing or decreasing d_t, we can decrease or increase the number of clusters considered. Our approach to selecting the value d_t ensures that the clusters or groupings formed are unambiguous and stable, i.e., small changes in d_t do not result in large changes in the number of clusters generated. Toward this end, we apply a Z-test Biswas et al. (2003) to determine where the change in the distance vector, d_v is statistically significant. We consider the last N_2 distance values d_v to compute the mean value of distance dis-

tribution (assumed to be a normal distribution):

$$\mu_d(k) = \frac{1}{N_2} \sum_{i=k-N_2+1}^{k} d(i). \quad (3)$$

The last N_1 samples (typically, $N_1 \gg N_2$) to compute the variance:

$$\sigma_d^2(k) = \frac{1}{k - N_1 - 1} \sum_{i=k-N_1+1}^{k} (d(i) - \mu_d(k))^2. \quad (4)$$

The confidence level for the Z-test, α, determines the bounds, z_-, and z_+, and, therefore, the number of clusters the algorithm generates.

$$P(z_- < (d(k) - \mu_d(k)) < z_+) = 1 - \alpha. \quad (5)$$

The Z-test is implemented as follows:

$$z_- < d(k) - \mu_d(k) < z_+ \to \text{merge the clusters} \\ \text{Otherwise} \to \text{stop} \quad (6)$$

Algorithm 1 Clustering

1: **input:** O
2: **output:** G , d_v
3: **for each** $O_i, O_j \in O$ **do**
4:     $D(i, j) \leftarrow dist(O_i, O_j)$
5: $G \leftarrow O$
6: **while** $|G| > 1$ **do**
7:     **for each** $X, Y \in G$ **do**
8:         **if** dist(X,Y) is equal to min distance in G **then**
9:             merge X and Y
10:            add $dist(X, Y)$ to d_v

3.3. Extracting significant features and expert-supported anomaly detection

Steps 3 and 4 are designed to assign meanings or labels to the derived groups. As a first step, we consider large groups derived from the clustering algorithm to be nominal (this corresponds to the assumption that the system operates normally most of the time). Singletons and smaller groups that are sufficiently distant from the nominal groups are labeled as outliers or anomalies. As discussed earlier, spacecraft missions are complex, and they may involve multiple phases and operational modes, corresponding to trajectory maneuvers and conducting of scientific experiments, over the duration of the mission. Some of the smaller clusters, initially labeled as anomalies or outliers, may, in reality correspond to special modes of operation, and, therefore, are not of interest in discovering discrepant and faulty behavior. Therefore, an additional challenge we face in this work is separating the special modes of operation from truly anomalous behaviors.

We have developed an approach to extract additional cues to identify special operating modes. We map the objects con-

stituting the smaller groups back onto the mission timeline, and look for discrete switching signals that may explain the differences in system behavior from the nominal. For example, the reaction wheels are activated to correct the attitude of the spacecraft, and this can be detected by a switch turning on to supply power to the reaction wheels. The activation of the reaction wheels increases the overall load currents in the power system, but since this increase can be primarily be correlated with the switches being commanded on, the outlier group corresponding to this phenomena is labeled as a special operational mode rather than an anomaly. This approach is generalized and applied to detect other special modes of operation. Other groups of data objects not be explained by observed mode changes in spacecraft operations, are then presented to human experts for further characterization. These may turn out to be additional special modes that are not easily interpreted from the switching signals, or they may represent anomalous behaviors that are linked to faults in the system.

To facilitate discovery of anomalies and special modes, we picked *significant features* that best differentiate each anomalous group from the labeled nominal groups.

Definition 1 (Significant features)   *Significant features are a single feature or a set of features that best distinguish an outlier group from nominal operations of a system.*

These features help our human experts better understand and characterize the anomalous situation as potential faults, or special modes of operation.

Different methods, such as variance decomposition Grömping (2012) and information gain measures applied to decision trees Ishwaran (2007) can be applied to extract significant features for each outlier cluster. In our work, we developed a simple *Euclidean distance based method* to extract significant features: The distance measure between normal operation group, a, and an outlier group, b for signal variable, j is computed as:

$$D_{ab}^{j} = \sqrt{\sum_{i=1}^{l} \left(\frac{E[o_{ai}^{j}] - E[o_{bi}^{j}]}{E[o_{ai}^{j}]} \right)^2}, \quad (7)$$

where $E[o_{ai}^{j}]$ represents the mean value of the ith level coefficient of signal j in group a. When summed over all m variables, the total distance between the normal operation group, a and an outlier group, b is computed as:

$$D_{ab} = \sqrt{\sum_{j=1}^{m} (D_{ab}^{j})^2}, \quad (8)$$

We define the importance of each time series waveform v_j in distinguishing an outlier group, b, from normal operations, a,

$I_{ab}(v_j)$, as the ratio of D_{ab}^{j} to D_{ab}, i.e.,

$$I_{ab}(v_j) = \frac{D_{ab}^{j}}{D_{ab}} \quad (9)$$

The importance of a set of variables, $V_k = \{v_1, v_2, ..., v_k\}$ in distinguishing b from normal operation, a, is defined as:

$$I_{ab}(V_i) = \sqrt{\sum_{i=1}^{k} (I_{ab}(v_k))^2}. \quad (10)$$

Let $V = \{v_1, v_2, ..., v_m\}$ denote the set of variables. We select a subset of variables V_b to guarantee a minimum required importance, I_r, in distinguishing b from normal operation with minimum cardinality, i.e.,

$$\begin{aligned} \min \quad & V_b \subseteq V \\ \text{s.t.} \quad & I_{ab}(V_b) > I_r \end{aligned} \quad (11)$$

Once the significant features have been established and ranked, this information is presented to the human expert to further characterize the anomalous group. After study, the expert may establish that this group represents a true anomaly, i.e., unexpected or aberrant behavior, or otherwise it is a mode of operation that cannot be easily characterized using the discrete switching signals and the time of occurrence.

4. CASE STUDY: LADEE ELECTRICAL POWER SYSTEM

The data used for this case study was telemetry data from the LADEE lunar mission directed by the NASA Ames Research Center. This mission lasted for approximately a year from launch till the spacecraft was intentionally crashed onto the moon's surface. The telemetry data we analyzed contained $2,949$ time-series waveforms that represented variables from the different subsystems of the spacecraft. The sampling rates for the waveforms differed between subsystems, and they also differed during the different phases of the mission. Overall, the data set contained $1,894,285,525$ samples, which was about 14 GB of data.

For this case study, we focused on the Electric Power System (EPS) of the spacecraft. This subsystem included 265 time series variables from 223 days of the mission, which corresponded to $574,687$ data samples for each variable. From the 265 variables, we selected 34 continuous voltage and current variables for analysis. Of these 7 represented voltage variables; this included the battery voltage, solar panel voltages, and load voltages. 27 were current measurements, such as, battery, solar panel, and load currents. 67 variables were binary-valued, and they helped us interpret the different modes of operation of the EPS. The remaining variables, temperature measurements and status checks on the data transmission, were not considered for this study.

4.1. Data pre-processing and feature extraction

For our analysis, we subdivided the 34 continuous voltage and current telemetry waveforms into 1512 windows, with each window corresponding to a data object. Each time window contained 380 samples. As discussed earlier, the sampling rate of the recorded data was not constant, therefore, a time window represented anywhere between 5 minutes to 10 hours of operation. The average window size was 3 hours and 31 minutes. The Haar wavelet transform (from the R package Aldrich (2010)[2]) was applied to each waveform segment to extract a set of 8 wavelet coefficients as distinct features representing that segment. The result was that the set of voltage and current waveforms for each data object were transformed into $34 \times 8 = 272$ features. To apply the discrete Haar transform, each waveform segment had to be represented by a number of samples that were a power of 2, therefore, we padded our waveform represented by 380 samples with $0's$ to make $2^9 = 512$ samples.

4.2. Clustering
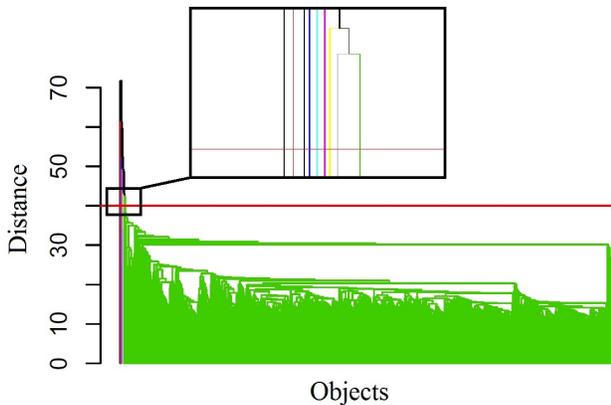
Figure 6. The dendrogram generated by applying the UPGMA hierarchical clustering algorithm. The red line represents the chosen threshold distance for cluster formation. The green section of the dendrogram (the large cluster) represents normal operations, and the outliers and smaller groups are represented by different colors

The Euclidean distance metric was used to create the dissimilarity matrix of 1512×1512 object pairs. Then we applied the UPGMA hierarchical clustering algorithm (the R function, $hclust$) to generate the dendrogram shown in Figure 6. The dendrogram is a graphical representation of the order in which the objects and groups merge to form larger clusters. Figure 7 represents 1511 distance values at which the objects and groups merged to form larger groups in the dendrogram. For this case study, which was conducted more as a proof-of-concept as opposed to an attempt to exhaustively generate all of the special modes and anomalies. Therefore, we intention-

[2]see http://www.R-project.org/

ally set a very high confidence bound of 99.7% to establish the level at which the dendrogram would be cut to establish the number of clusters.
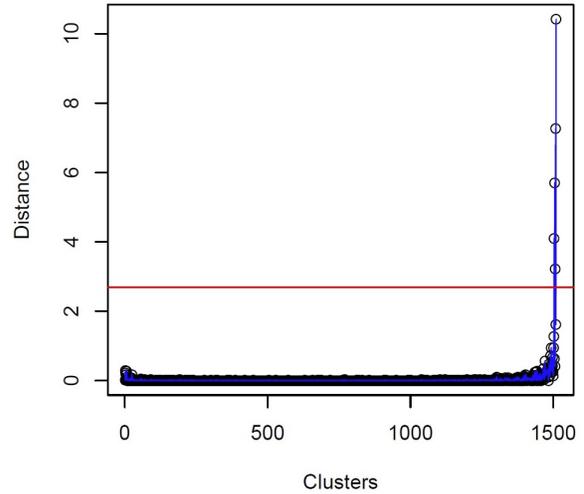
Figure 7. Distance values indicating levels of cluster formation

Application of equation (6) produced the distance threshold and the corresponding red line shown in Figure 7. As expected, this produced one large cluster that we assumed to represent the nominal behavior of the spacecraft. In addition, Figure 6 shows that we generated eight smaller clusters. We studied these groups in greater detail by comparing them against the nominal group to determine if they represented special modes and anomalies. Of the eight groups, three turned out to be modes of operation that were identifiable because of their correspondence to the switching signals. In this paper, we analyze the other five groups in greater detail. To study the five smaller clusters, we identified the objects corresponding to these clusters on the spacecraft mission timeline. Figure 8 shows these objects as dots on the timeline plot.

As a first step toward mode characterization, we studied the discrete switch values during these intervals to see if they provided information about special modes of operation. When we were unable to assign a definite interpretation to a cluster, we extracted the significant features that differentiated that cluster from the nominal group. Using equation (11) we selected significant features by setting the threshold, $I_r = 0.9$. The significant features represented an ordered subset of features that contributed the largest amounts to the distance from the mean of the outlier group to the mean of the nominal, and the chosen subset accounted for 90% of the distance between the outlier and nominal group means. Table 1 summarizes the significant features of five of the outlier groups we identified by our clustering approach. We presented the significant features for each anomalous group and the position at
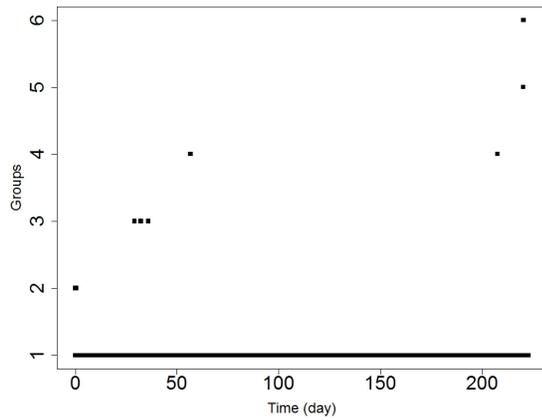
Figure 8. Clusters projected back on the mission timeline

Figure 10. Significant features for cluster 2

Figure 9. Reaction wheels (OFF=0, ON=1)

which they occurred on the mission time line to our experts to help us further characterize and classify the special modes and anomalies.

4.3. Characterizing Special Modes and Anomalies

In this section, we present our characterization of the five smaller clusters in greater detail that we generated using expert input. Our mission experts and specialists from NASA Ames are acknowledged at the end of this paper. Cluster 1 represented the nominal behavior of the spacecraft. CLusters 2-6 are discussed below.

4.3.1. Cluster 2: The reaction wheels control problem

The behavior represented by this cluster covered two time windows that occurred early in the mission. The behaviors covered 40 and 6 minutes of the corresponding data objects. Figure 9 shows that the reaction wheels went off twice (corresponding to Reaction Wheels $= 0$) during the mission. Our

experts confirmed after studying the mission operator logs that the reaction wheels only went off once during the mission, and the second zero in the figure was a case of bad data. Figure 10 shows that different currents in the SATORI board[3]#2 were the most significant features for this cluster. Figure 11 shows that the SATORI #2 current variable for both data objects in group 2 repeatedly exceeded the 95% confidence bounds of the current during nominal operations.

The high current values read by the sensors on the SATORI #2 boards indicated that this incident very likely was related to the guidance navigation and control unit. The fact that these currents were related to the three Reaction wheels further confirmed this interpretation. The experts from NASA further substantiated this anomaly as follows. In the first few orbits around the earth, the spacecraft began to spin at a faster rate than was expected, and the reaction wheels were turned off by the control software to avoid a high load current, and, therefore, draining of the battery. This stopped the spacecraft rotations, but, as a consequence, the side of the spacecraft facing away from the sun got became too cold. Several heaters went on to prevent the equipment from freezing, and this led to the high currents in a number of units connected through the SATORI board#2.

4.3.2. Cluster 3: Lunar orbit insertion

Figure 8 shows that the cluster 3 data objects corresponded to three time intervals that occurred on three different days of the mission. Each time interval was about six minutes long. Two different currents in the Power-switching and Pyro Integration boards (PAPI) board[4]#2 were the significant features that characterized this group (see Figure 10). Figure 13 shows

---

[3]The SATORI boards provide power to the Command & Data Handling System.
[4]The PAPI boards route power to the Thermal Control, Guidance, Navigation & Control and the Propulsion Subsystems.

Table 1. Summary description of the detected modes and anomalies

Group	Detected Mode or Anomaly	Significant Features	Switches
1	Normal operation mode		
2	Anomaly: Reaction wheels	• Four SATORI #2 currents	• Propulsion heater turned on • Star tracker went off
3	Mode: Lunar orbit insertion	• Two PAPI #2 currents	• Pressurant tank heater went on • Valve driver unit went on
4	Anomaly: Laser communication test (during dark phase)	• Three SATORI #1 currents • Load Current • Battery Current	• Laser communications switch went on
5	Anomaly: Eclipse lasted longer than expected	• Battery Voltage • SATORI #1 Voltage • SATORI #2 Voltage • PAPI #2 Current	• Several heaters went on (e.g. Propulsion heater)
6	Mode: Safe	• Battery Voltage • SATORI #1 Voltage • SATORI #2 Voltage • SATORI #2 High Pressure (HP) Current • PAPI #2 Current • Solar Array (SA) Current	• Several loads (e.g. star tracker) turned off

that the PAPI #2 high pressure current number 7 during these three time intervals. The high amplitude in the PAPI # 2 propulsion subsystem current was the second significant feature for this cluster. However, unlike Cluster 6, the increase in PAPI board #2 current did not occur simultaneously with the battery voltage drop.

With the help of our experts, we found out that the valve driver unit, which controls the propulsion subsystem and the pressurant tank heaters, (part of the propulsion subsystem (see Figure 1)) were ON for the three time intervals. This corresponded to a unique behavior, however, our experts confirmed that the behavior was not anomalous. Instead, it represented the lunar orbit insertion process. There were three firings of the propulsion subsystem that occurred to get the spacecraft into lunar orbit and our algorithm successfully grou-ped them into a single cluster.

4.3.3. Cluster 4: The laser communication test

This cluster included two time windows, each about 20 minutes in duration. Figure 15 shows the SATORI #1 current, which was the most significant feature for this cluster. The current values exceeded the 95% upper bound for nominal operations.

The load current and battery current are the next two significant features for this cluster. Further analysis showed that the data points in this cluster corresponded to laser communication tests, which were part of the mission plan. The laser communication tests increased load currents significantly. The two time intervals in this cluster also coincided with the occurrence of a new moon, which meant that the solar arrays were not generating any current (values recorded were very close to 0) during this period. The high battery current caused the battery to discharge below acceptable levels, and, the battery voltage dropped below set thresholds. Our experts characterized this as an anomaly in operations because the laser communications test led to unintended consequences of the
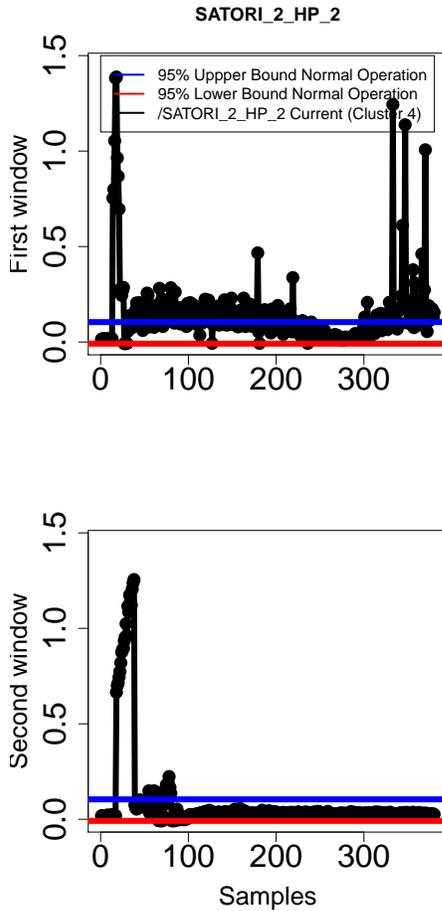
Figure 11. SATORI #2 high pressure current for cluster 2 objects

Figure 12. Significant features for cluster 3

battery voltage dropping below specified thresholds.

4.3.4. Cluster 5: The eclipse

The objects in this cluster extended over a 5 hour time span. It should also be noted that the sampling rate was also significantly lower, because this was the end of the mission. The most significant feature for this group was the battery voltage, which fell below the 95% bounds of normal operation (see Figure 16). The drop in battery voltage led to drops in the SATORI #1 and SATORI #2 voltages. Figure 17 shows SATORI #1 and SATORI #2 voltages were the next set of significant features.

The experts confirmed that cluster 5 behavior was directly the consequence of the eclipse that happened at the end of the mission. The solar array current was zero during the first 2 hours of this time interval, and it fluctuated between zero and small non zero values after that. The lack of sunlight caused a significant drop in temperatures (we did not include temperature values in this analysis), and several heaters came on to prevent large temperature drops, which would have affected spacecraft operations. This increased the load current significantly. A simultaneous increase in the load currents and decrease in the solar array current put an unprecedented load on the battery, which led to large voltage drops in the battery voltage. This likely jeopardized the battery health, therefore, it clearly represented anomalous or unexpected behavior of the EPS.

4.3.5. Cluster 6: The safe mode

The system went into the safe mode right after the eclipse ended. This mode was about $4\frac{1}{2}$ hours long. The battery voltage was again the most significant feature that distinguished this group from nominal operations. Figure 18 shows battery voltage during this time interval. The remaining significant features for this cluster and their importance factors are presented in Figure 19.

To preserve the health of the battery, several loads were switched off to reduce energy consumption and give the battery a chance to recharge. Figure 18 shows that the battery voltage came back to an acceptable level during this mode. Our experts explained that the data points in this group represented a unique behavior in spacecraft operations. However, they did not classify the behavior to be anomalous, since the spacecraft systems operated exactly as they should have to avoid larger failures and possible loss of the spacecraft power system. Hence, this was a special operational mode, i.e., the "safe mode" to allow for EPS recovery.

5. DESIGNING THE ONLINE FAULT MONITORS

In order to leverage the above anomaly detection process, and leverage the findings to make actionable decisions, such as invocation of a corrective mechanism to preserve mission goals and sub goals, accurate detection and isolation of degraded or

Figure 14. Significant features for cluster 4

Figure 13. PAPI #2 high pressure current number 7 for cluster 3 objects
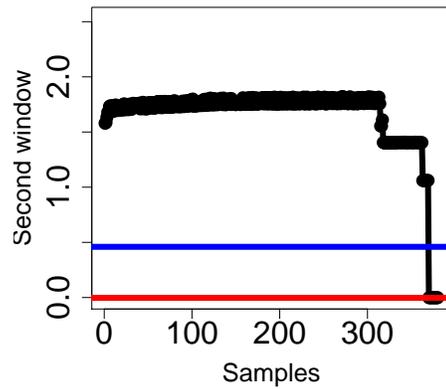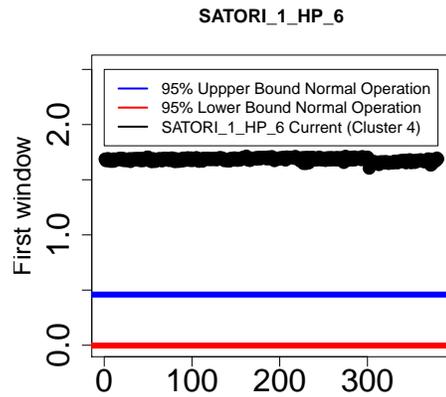
Figure 15. SATORI #1 high pressure # 6 current for cluster 4 data points

failed components is crucial. The following section focuses on leveraging the anomaly detection results for subsequent
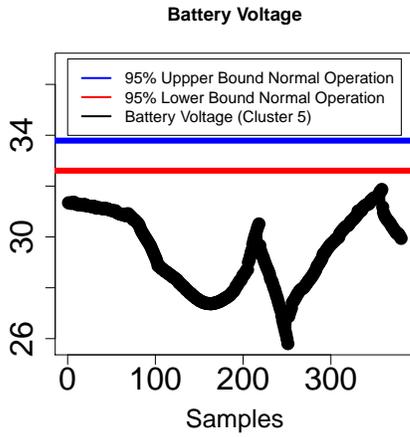
Battery Voltage

Figure 16. Battery voltage for cluster 5 data points

Significant Features

Figure 17. Significant features for cluster 5

Battery Voltage

Figure 18. Battery voltage for cluster 6 objects

Significant Features
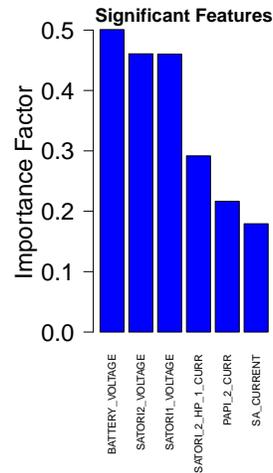
Figure 19. Significant features for cluster 6

failure identification and isolation through a model-based approach using QSI's TEAMS®suite of tools.

5.1. TEAMS®model based methodology fault identification from anomaly detection

QSI's TEAMS suite of tools captures the failure or degradation cause and its corresponding functional failure effects dependency relationships through a multi-signal modeling methodology Deb et al. (1995). The multi-signal modeling methodology used in TEAMS can be conceptualized as a *colored* dependency graph, where, as in structural models, but rendered from a failure-space perspective, the structural components' modes of failures are modeled as nodes and the directional links denote dependencies based on structural adjacency. Failure manifestations perceived through anomalous sensed data, such as out of range voltage measurements for a battery, are denoted as tests. Corresponding signals representing the loss of specific functionality, as appropriate, are

also attached to these effects and tests.

The TEAMS-RT®inference engine processes failure events (error codes, built-in test failures, performance anomalies, etc.), as they become available. It uses the data to infer the status of the root causes (the identification of one or more component faults) and is appropriate for processing onboard data that is either received in real time or downloaded post-mission/operation. The ability of the TEAMS-RT inference engine to diagnose faults in real-time for large complex interconnected systems, such as the Ares-I and the Orion spacecraft and most recently on the Exploration Test Flight launch has been demonstrated by NASA.

13

5.2. Modeling the LADEE power system in TEAMS®- designing intelligent tests from anomaly detectors

Using the high level schematics, inputs from subject matter experts (SMEs) and other documentation such as LADEE comprehensive design review (CDR) presentations, we built a simplified TEAMS model for part of the Electrical Power System (EPS). The electric power system (EPS) in LADEE consists of a solar array, a battery pack, a power distribution system and various spacecraft loads. We modeled the organization of the cells in the battery, and the solar cells in the various modules in great detail, but had to make assumptions about the sensor locations and various loads. A top level view of a hierarchical TEAMS model is shown in Figure 20.
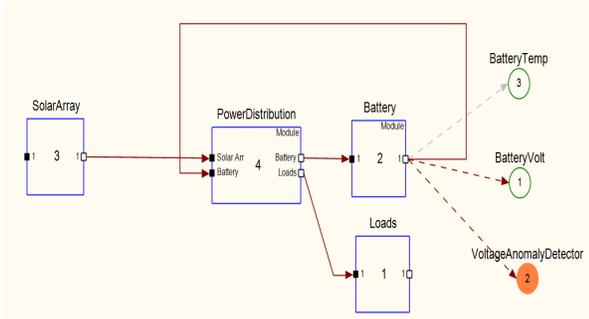
Figure 20. Top Level view of the Power System of LADEE
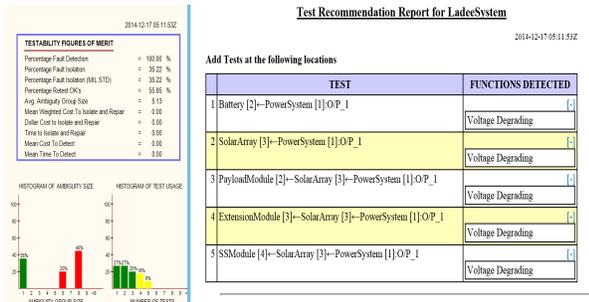
Figure 21. Testability analysis and Test recommendation report from TEAMS®

A key value proposition of this overall effort is to leverage the information gathered through the anomaly detection techniques and incorporate them in the design of intelligent fault isolation and degradation tests that can detect failures or degradations of the relevant functions of the various LADEE system components thereby improving both real-time and prognosed system health determination of the system. The TEAMS reasoning engine such as the one in TEAMS-RT, when presented with the appropriate behavior of the system reasons across the entire set of observations, translates them into appropriate test results based on these anomaly detections methods that have been incorporated as part of the mechanism for test result determination, and identify the root

cause(s) for the degradations and failures as stated earlier. In addition, of key importance is to distinguish system operational modes or phases (such as the lunar orbit insertion mode as described in Section 4.3.2) from true anomalous behavior.

In order to demonstrate the power of diagnosis driven prognostic framework, we conducted a design for prognostics study to improve prognostics capabilities in the LADEE power system model. The prognostics recommendation report for the LADEE system, shown in Figure 21 (table on the right-side), proposes adding 5 degradation detection type tests that will look across multiple existing sensors to detect voltage degradation functions. This motivated the development of the Battery Voltage Anomaly Detector test point as shown in TEAMS model in Figure 20. Multiple anomaly detection tests with their own ability to detect appropriate functional failure (signal) and can incorporate the anomaly detection mechanisms described in Sections 4.3.3 (The laser communication test) and 4.3.4 (The Eclipse) can reside in this Battery Voltage Anomaly Detector test point. In addition, the operational mode or phase related information for these two test cases can also be incorporated as detection mechanisms, provided there are no other direct and readily available mechanisms from the vehicle telemetry to assert the presence of those operational modes or phases. Examples of the different failure propagation for the same degradation failure during different operational modes through the model is shown in Figures 22 and 23.

Thus the design of intelligent fault and degradation tests based on the anomaly detectors provides proof of concept how model-based analysis has the potential to improve diagnostics and prognostics capabilities of future spacecraft systems. It enables one to judiciously utilize precious development dollars and add useful prognostics capability without necessarily increasing sensor requirements.
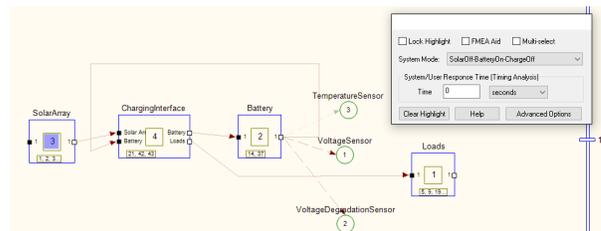
Figure 22. Solar cell failure propagation in a charge-off mode

6. A GENERIC COMPUTATION PLATFORM FOR ENCODING THE MULTI-TIMESCALE ANALYTICS WORKFLOW

Traditionally, centralized architectures have been used for both data analytics either for real time applications or for analyzing historical data. For example, in modern power distribution systems, data is collected in real-time from smart
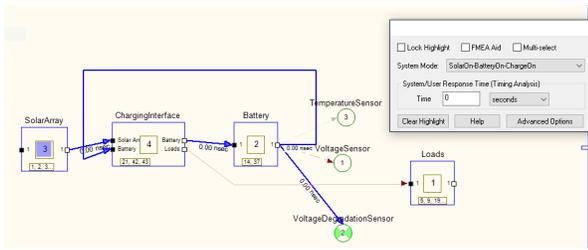
Figure 23. Same solar cell failure propagation in a mode when charging is active. Figure shows the degradation behavior being propagated through the system and detected by the Voltage Degradation Sensor anomaly detector during the charging mode

meters, however, all the analysis happens in a centralized control center Budka et al. (2010). However, as large-scale data collection, processing, and analysis are becoming commonplace, especially in large cyber-physical systems (CPS) and smart and connected communities, the shortcomings of this centralized model are becoming apparent: increased latencies, the lack of scalability, single points of failure, and saturated communication resources.

The emerging CPS/IoT ecosystem platforms such as Beaglebone Blacks, Raspberry Pi, Intel Edison and other edge devices such as SCALE Benson et al. (2015), Paradrop Willis et al. (2014) are providing new capabilities for data collection, analysis and processing at the *edge* Vaquero & Rodero-Merino (2014) (also referred to as Fog Computing). This allows the dynamic composition of computing and communication networks that can be used to monitor and control the physical phenomena closer to the source. Supported by traditional cloud services, this kind of computation platform can support both long-term model constructions, a computation heavy task and an online real-time classification, providing anomaly detection.

Figure 24 describes a potential encoding of the application defined in this paper as a workflow. As can be seen in the figure, the workflow can be cleanly separated into a section for long-term processing that is used to produce fault models, and a section for real-time online classification that supports anomaly detection and failure mode isolation. Designing and managing these applications with different temporal requirements is difficult because they require different computation and communication patterns. For example, real-time applications in this domain require a traditional control loop computation pattern. In comparison, a long running, computation heavy, big-data application is usually implemented using some notion of a computation graph supported by existing dataflow engines such as Storm Apache Software Foundation (n.d.-b), Spark Apache Software Foundation (n.d.-a), or TensorFlow Abadi et al. (n.d.). These differences require us to devise solutions that facilitate integration of and interaction between applications with varying computation models.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have developed a mixed anomaly detection method that combine unsupervised learning methods combined with human-expert support to analyze telemetry data from spacecraft missions. We have described the various steps of the method from the data pre-processing, generation of the feature space, applying a clustering algorithm, determining nominal and outlier processes, associating significant features with the outlier groups, to the consultation with experts resulting in the identification and characterization of special modes of operation as well as anomalous behavior of the system. In addition, we discussed how anomalies, once detected may be used to design and deploy online monitors within the TEAMS-RT environment. Last, we provided a general computational architecture that can support historical analysis of data for anomaly detection as well as the online mechanisms needed for fault detection and isolation.

As a case study, we applied our approach to analyzing telemetry data from the Electric Power System (EPS) of a recent lunar mission called LADEE. Our case study provided interesting results. We were successful in working with mission experts to identify a set of special modes as well as some anomalies that occurred during the mission. The use of significant features as well as the projection of the outlier data groups back onto the mission timeline greatly facilitated the mission experts' tasks of identifying and characterizing the special modes and anomalies. This approach shows great promise in generalizing to complex cyber physical systems (CPSs), where well-developed models of the system are not readily available, therefore, operational data has to be used to understand and evaluate system operations, and detect anomalies and outlier behaviors.

In future work, we will scale up the anomaly detection work using computational architectures like CHARIOT Pradhan et al. (2015), and investigate different algorithms to improve our anomaly detection and fault classifier methods. We will scale up our case study for the LADEE system to include all of the subsystems of the spacecraft.

REFERENCES

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... others (n.d.). Tensorflow: Large-scale machine learning on heterogeneous systems, 2015. *Software available from tensorflow. org*.

Aldrich, E. (2010). Wavelets: a package of functions for com-
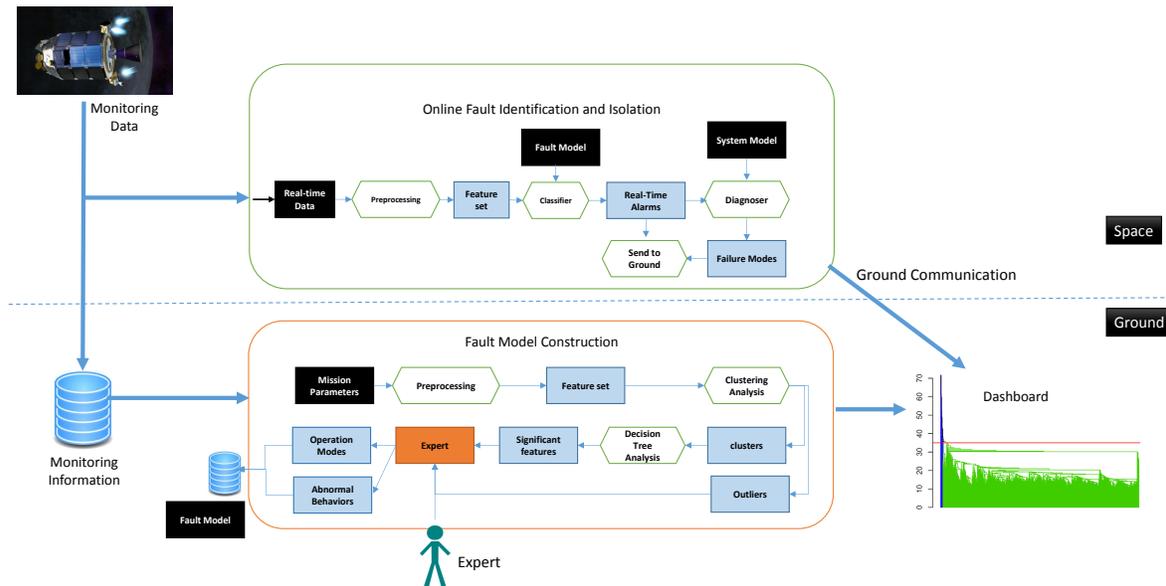
Figure 24. Analytics Workflow.

puting wavelet filters, wavelet transforms and multiresolution analyses. *R package version 0.2-60.*

Apache Software Foundation. (n.d.-a). *Apache Spark.* http://spark.apache.org/.

Apache Software Foundation. (n.d.-b). *Apache Storm.* http://storm.apache.org/.

Bay, S. D., & Schwabacher, M. (2003). Mining distance-based outliers in near linear time with randomization and a simple pruning rule. In *Proceedings of the ninth acm sigkdd international conference on knowledge discovery and data mining* (pp. 29–38).

Benson, K., Fracchia, C., Wang, G., Zhu, Q., Almomen, S., Cohn, J., . . . others (2015). Scale: Safe community awareness and alerting leveraging the internet of things. *IEEE Communications Magazine*, *53*(12), 27–34.

Bishop, C. (2001). *Bishop pattern recognition and machine learning.* Springer, New York.

Biswas, G., Simon, G., Mahadevan, N., Narasimhan, S., Ramirez, J., & Karsai, G. (2003). A robust method for hybrid diagnosis of complex systems. *In Proceedings of the 5th Symposium on Fault Detection, Supervision and Safety for Technical Processes, 2003, June*, 1125–1131.

Blanke, M., & Schröder, J. (2006). *Diagnosis and fault-tolerant control* (Vol. 691). Springer.

Budalakoti, S., Srivastava, A. N., Akella, R., & Turkov, E. (2006). *Anomaly detection in large sets of high-dimensional symbol sequences.*

Budalakoti, S., Srivastava, A. N., & Otey, M. E. (2009). Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *39*(1), 101–113.

Budka, K. C., Deshpande, J. G., Doumi, T. L., Madden, M., & Mew, T. (2010). Communication network architecture and design principles for smart grids. *Bell Labs Technical Journal*, *15*(2), 205–227.

Burrus, C. S., Gopinath, R. A., & Guo, H. (1997). *Introduction to wavelets and wavelet transforms: a primer*. Prentice-Hall, Inc.

Caliski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods 3, no. 1*, 1–27.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM computing surveys (CSUR)*, *41*(3), 15.

Chen, J., & Patton, R. J. (2012). *Robust model-based fault diagnosis for dynamic systems* (Vol. 3). Springer Science & Business Media.

Chidester, T. R. (2003). Understanding normal and atypical operations through analysis of flight data. In *Proceedings of the 12th international symposium on aviation psychology, dayton, oh* (pp. 239–242).

Chu, E., Gorinevsky, D., & Boyd, S. (2010). Detecting aircraft performance anomalies from cruise flight data. In *Aiaa infotech aerospace conference, atlanta, ga.*

Das, S., Matthews, B. L., & Lawrence, R. (2011). Fleet level anomaly detection of aviation safety data. In *Prognostics and health management (phm), 2011 ieee conference on*

(pp. 1–10).

Das, S., Matthews, B. L., Srivastava, A. N., & Oza, N. C. (2010). Multiple kernel learning for heterogeneous anomaly detection: algorithm and aviation safety case study. In *Proceedings of the 16th acm sigkdd international conference on knowledge discovery and data mining* (pp. 47–56).

Day, W. H., & Edelsbrunner, H. (1984). Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of classification*, *1*(1), 7–24.

Deb, S., Pattipati, K. R., Raghavan, V., Shakeri, M., & Shrestha, R. (1995). Multi-signal flow graphs: a novel approach for system testability analysis and fault diagnosis. *IEEE Aerospace and Electronic Systems Magazine*, *10*(5), 14–25.

De Kleer, J., & Williams, B. C. (1987). Diagnosing multiple faults. *Artificial intelligence*, *32*(1), 97–130.

Duda, R. O., Hart, P. E., & Stork, D. G. (2012). Pattern classification. *John Wiley and Sons.*.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, pp. 226–231).

Fukunaga, K., & Koontz, W. L. (1970). A criterion and an algorithm for grouping data. *IEEE Transactions on Computers 100, no. 10*, 917–923.

Grömping, U. (2012). Estimators of relative importance in linear regression based on variance decomposition. *The American Statistician.*.

Hartigan, J. A. (1975). Clustering algorithms. *New York: Wiley*.

Hine, B., Spremo, S., Turner, M., & Caffrey, R. (2010). The lunar atmosphere and dust environment explorer (ladee) mission. In *Ieee aerospace conference.*

Hodge, V. J., & Austin, J. (2004). A survey of outlier detection methodologies. *Artificial intelligence review*, *22*(2), 85–126.

Hubert, L. J., & Levin, J. R. (1976). A general statistical framework for assessing categorical clustering in free recall. *Psychological bulletin 83, no. 6*.

Isermann, R. (2005). Model-based fault-detection and diagnosis–status and applications. *Annual Reviews in Control*, *29*(1), 71–85.

Ishwaran, H. (2007). Variable importance in binary regression trees and forests. *Electronic Journal of Statistics 1: 519-537.*.

Iverson, D. L. (2004). Inductive system health monitoring. In *Proceedings of the international conference on machine learning; models, technologies & applications, mlmta '04.*

Jain, A. K., & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc.

Ji, M., Zhang, Z., Biswas, G., & Sarkar, N. (2003). Hybrid fault adaptive control of a wheeled mobile robot. *Mechatronics, IEEE/ASME Transactions on*, *8*(2), 226–233.

Lee, E. A. (2008). Cyber physical systems: Design challenges. In *Object oriented real-time distributed computing (isorc), 2008 11th ieee international symposium on* (pp. 363–369).

Li, L., Gariel, M., Hansman, R. J., & Palacios, R. (2011). Anomaly detection in onboard-recorded flight data using cluster analysis. In *Digital avionics systems conference (dasc), 2011 ieee/aiaa 30th* (pp. 4–11).

Mack, D. L. (2013). *Anomaly detection from complex temporal sequences in large data* (Unpublished doctoral dissertation). Vanderbilt University.

Mack, D. L., Biswas, G., Koutsoukos, X. D., & Mylaraswamy, D. (2016, in press). Learning bayesian network structures to augment aircraft diagnostic reference models. *IEEE Transactions on Automation Science and Engineering*.

Marwedel, P. (2010). *Embedded system design: Embedded systems foundations of cyber-physical systems*. Springer Science & Business Media.

Milligan, G. W., & Cooper, M. C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika 50.2*, 159–179.

Niggemann, O., Biswas, G., Kinnebrew, J. S., Khorasgani, H., Volgmann, S., & Bunte, A. (2015). Data-driven monitoring of cyber-physical systems leveraging on big data and the internet-of-things for diagnosis and control. *26th International Workshop on Principles of Diagnosis, Paris, France*.

Noura, H., Theilliol, D., Ponsart, J.-C., & Chamseddine, A. (2009). *Fault-tolerant control systems: Design and practical applications*. Springer Science & Business Media.

Pradhan, S. M., Dubey, A., Gokhale, A., & Lehofer, M. (2015). Chariot: a domain specific language for extensible cyber-physical systems. In *Proceedings of the workshop on domain-specific modeling* (pp. 9–16).

Qin, S. J. (2012). Survey on data-driven industrial process monitoring and diagnosis. *Annual Reviews in Control*, *36*(2), 220–234.

Rätsch, G., Schölkopf, B., Mika, S., & Müller, K.-R. (2000). *Svm and boosting: One class*. GMD-Forschungszentrum Informationstechnik.

Strang, G. (1993). Wavelet transforms versus fourier transforms. *Bulletin of the American Mathematical Society*, *28*(2), 288–305.

Sugar, C. A., & James, G. M. (2011). Finding the number of clusters in a dataset. *Journal of the American Statistical Association*.

Vaquero, L. M., & Rodero-Merino, L. (2014). Finding your way in the fog: Towards a comprehensive definition of fog computing. *SIGCOMM Comput. Commun. Rev.*, *44*(5), 27–32.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., & Kavuri, S. N. (2003). A review of process fault detection and diagnosis: Part i: Quantitative model-based methods.

Computers & chemical engineering, *27*(3), 293–311.

Willis, D., Dasgupta, A., & Banerjee, S. (2014). Paradrop: A multi-tenant platform to dynamically install third party services on wireless gateways. *Proceedings of the 9th ACM workshop on Mobility in the evolving internet architecture*, 43–48.

Yan, M. (2005). *Methods of determining the number of clusters in a data set and a new clustering criterion* (Unpublished doctoral dissertation). Virginia Polytechnic Institute and State University.

Yin, S., Ding, S. X., Xie, X., & Luo, H. (2014). A review on basic data-driven approaches for industrial process monitoring. *Industrial Electronics, IEEE Transactions on*, *61*(11), 6418–6428.