Semi-Supervised Machine Learning for Spacecraft Anomaly Detection & Diagnosis

Sowmya Ramachandran
Stottler Henke Associates, Inc.
1650 S. Amphlett Blvd.
San Mateo, CA 91109
sowmya@stottlerhenke.com

Maia Rosengarten
Stottler Henke Associates, Inc.
1650 S. Amphlett Blvd.
San Mateo, CA 91109
mrosengarten@stottlerhenke.com

Christian Belardi
Stottler Henke Associates, Inc.
1650 S. Amphlett Blvd.
San Mateo, CA 91109
cbelardi@stottlerhenke.com

Abstract—**This paper describes Anomaly Detection via Topological-feature Map (ADTM), a data-driven approach to Integrated System Health Management (ISHM) for monitoring the health of spacecraft and space habitats. Developed for NASA Ames Research Center, ADTM leverages proven artificial intelligence techniques for rapidly detecting and diagnosing anomalies in near real-time. ADTM combines Self-Organizing Maps (SOMs) as the basis for modeling system behavior with supervised machine learning techniques for localizing detected anomalies. A SOM is a two-layer artificial neural network (ANN) that produces a low-dimensional representation of the training samples. Once trained on normal system behavior, SOMs are adept at detecting behavior previously not encountered in the training data. Upon detecting anomalous behavior, ADTM uses a supervised classification approach to determine a subset of measurands that characterize the anomaly. This allows it to localize faults and thereby provide extra insight. We demonstrate the effectiveness of our approach on telemetry data collected from a lab-stationed CubeSat (the "LabSat") connected to software that gave us the ability to trigger several real hardware faults. We include an analysis and discussion of ADTM's performance on several of these fault cases. We conclude with a brief discussion of future work, which contains investigation of a hierarchical SOM-architecture as well as a Case-Based Reasoning module for further assisting astronauts in diagnosis and remediation activities.**

TABLE OF CONTENTS

1. INTRODUCTION.. 1
2. RELATED WORK 2
3. SELF-ORGANIZING MAP BACKGROUND 2
4. METHODS... 3
5. EXPERIMENTS AND DISCUSSION 5
6. CONCLUSIONS AND FUTURE WORK..................... 8
7. REFERENCES.. 8
8. BIOGRAPHY 9

1. INTRODUCTION

Integrated System Health Management (ISHM) technologies are mission-critical for space-exploration. Space habitats are made up of a complex web of subsystems. Traditionally, model-based reasoning techniques have been effective in monitoring the health of such operations, but the rising demand for rapid fault detection and response in deep-space habitats calls for autonomous monitoring software that is agile and can respond to previously unseen events. In particular, communication delays between on-board crews and Earth-bound experts could make the difference between a successful and failed mission, risking the loss of both equipment and crew. The accelerated plans for establishing a lunar outpost within the next decade and for sending human exploration teams to Mars in the 2030s are making these considerations particularly salient.

Model-based reasoning systems are effective at detecting and diagnosing faults that are known, but their knowledge can be insufficient when faced with novel situations. Complete mathematical models of systems and subsystems may overcome these problems. However, they are computationally expensive, especially for modeling complex systems with a high number of interdependent parts. On the other hand, data-driven approaches using machine learning techniques overcome model-based limitations by evolving their knowledge with new situations. But they are not as effective at generating explanations or tracing root causes as rule-based modeling systems. Approaches to causal representations like Bayesian Networks do not scale to the level necessary to model a complex system such as a space habitat. A well-designed solution to this problem will offer computational efficiency and scalability while also combining the strength of rule-based and machine-learning based approaches.

This paper will describe a solution called Anomaly Detection via Topological-feature Map (ADTM) that combines proven data-driven and knowledge-based artificial intelligence techniques in a unique way to learn models of system behavior from data and use these models to identify and diagnose anomalies. We use Self-Organizing Maps (SOMs) as the basis for modeling system behavior. A SOM is a two-layer artificial neural network (ANN) that uses unsupervised learning to produce a low-dimensional representation of the training samples. Once trained on normal system behavior, SOMs are powerful at detecting behavior previously not encountered in the training data (i.e., anomalies).

978-1-7821-2734-7/20/$31.00 ©2020 IEEE

Upon detecting anomalous behavior, ADTM uses a supervised classification approach to determine a subset of measurands that characterize the anomaly. This allows it to localize faults and thereby provide extra insight.

In future work, we will combine the benefits of this data-driven approach with a Case-Based Reasoning approach to matching system behavior with similar prior observations. The goal is to assist astronauts in remediation and recovery activities by drawing from a knowledge base of known anomalies and response activities. While a situation might not have been encountered before in the exact form, there may be enough similarities with prior events that it can apply the knowledge to narrow the search space for diagnosis and response. Once encountered, new behaviors can be used to retrain a SOM and be added as a new example to the case-base; in this way, the anomaly detection and diagnosis system can evolve as system behavior evolves over time.

2. Related Work

The focus of this work was on unsupervised anomaly detection for discrete sequences of subsystem data using SOM-based models trained on nominal subsystem behavior. Similar approaches to anomaly detection have been applied in existing research. Principal Component Analysis has been a widely used algorithm for anomaly detection across a wide breadth of applications, including diagnosing offshore wind turbines [1], cyber networks [2], and space telemetry [3]. Furthermore, Gaddam used a supervised approach to anomaly detection by combining *k*-means clustering with ID3 decision tree classification [4]. The classification decisions across the clusters and decision trees were combined for a final decision on class membership. The main challenge for such an approach is access to labeled fault data, which can be limited in the space domain.

NASA Ames Research Center (ARC) uses *k*-means and density-based clustering techniques for system monitoring in its IMS and ODVEC software systems [5]. Similarly, Gao, Yang, and Xing used a K-Nearest-Neighbor (kNN) approach for anomaly detection of an in-orbit satellite using telemetry data [6]. SOMs have been used for fault detection and diagnosis in several industries. Datta, Mabroidis and Hosek combine SOMs with Quality Thresholding (QT) to refine the resolution of clusters learned by SOMs within the semi-conductor industry [7]. Similarly, Tian, Azarian, and Pecht train a SOM on nominal cooling fan bearing data but use a kNN approach in place of the more traditional Minimum Quantization Error (MQE) to assign test data anomaly scores based on their distance to centroids learned by the kNN model [8]. Cottrell and Gaubert apply anomaly scores to aircraft engine test data using the MQE approach that we have used in this paper (see *Section 4.1*) and leverage the visualization capabilities of SOMs to visualize the transition states of engines from run-to-failure datasets [9].

ADTM contributes to this existing bed of clustering research by combining a Self-Organizing Map with an Extra Tree Classifier for both detecting and localizing faults, which has rarely (if at all) been used in the ISHM space domain. Significantly, ADTM also provides remediation capabilities with a Case-Base Reasoning (CBR) module that assists end-users in responding to detected anomalies. It does so by retrieving records of similar past behavior with pertinent information about the anomaly and, when relevant, past troubleshooting activities.

Such assistance mirrors the role of Mission Control during a failure onboard a spacecraft. In such a situation, teams of scientists and domain experts on the ground help astronauts inflight quickly respond to a failure to mitigate further risk. They do so by drawing upon years of experience with the systems onboard the spacecraft as well as familiarity with past anomalies, either from test scenarios or real-time failures. ADTM's CBR module aims to mirror such remediation assistance in the context of deep-space exploration, where crew dependency on Mission Control is no longer an option due to significant communication delays.

3. Self-Organizing Map Background

ADTM leverages the benefits of an unsupervised neural network called a Self-Organizing Map (SOM) for implicit data clustering and anomaly detection. Also known as a Kohonen map, a SOM is a two-layer artificial neural network (ANN) that uses unsupervised learning to produce a low-dimensional representation of the training samples [10]. Inspired by the way sensory input (auditory, olfactory, tactile, etc.) map to specific areas of the cerebral cortex, SOMs are also tuned to various patterns of input data during training (Figure 1). Consistent with this analogy, the nodes in the output layer of a SOM are also called "neurons."

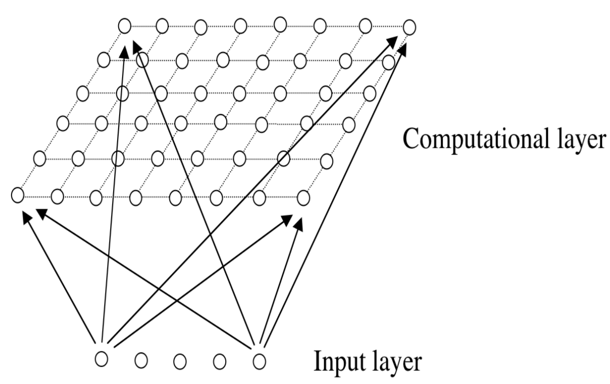

Figure 1: Self-Organizing Map Diagram

The goal of training a SOM is to transform incoming inputs to a 1- or 2-dimensional map in a topologically ordered fashion such that points that are close together in the higher-dimensional input space are close together in the lower-dimensional output space as well. This mapping allows us to detect patterns of normal or anomalous behavior in a system, as different types of behavior map to different output units.

Specifically, the N-dimensional input data is fed into the SOM in the first layer and fully connected to a lattice of (l rows, p columns) output neurons O_i in the second layer. Each neuron O_i is associated with a N-dimensional weight vector w_i . We represent O_i by a two-dimensional coordinate of its position in the (l by p) grid, e.g., $O_i = (x_i, y_i)$. Like the clustering technique k-means, the values of l and p are parameters that are tuned during model validation.

Based on the literature [11], we chose l and p such that $lp = 5\sqrt{N}$. Unlike k-means, however, the clusters learned during SOM training are topologically ordered through the following competitive learning process:

Each input vector $x_i \in X$ is compared with the N-dimensional weight vector $\{w_{1j}, w_{2j}, \ldots, w_{Nj}\}$ associated with each output node O_j. The closest O_j is chosen as the winner, or 'Best Matching Unit' (BMU), where 'close' is defined by a distance function (we chose Euclidean distance). Each BMU is associated with an entire neighborhood of related neurons whose weight vectors are also updated, though to a lesser extent, proportional to their distance to the BMU in the 2D output lattice.

In other words, entire neighborhoods of related neurons get updated in the direction of the input data that is closest to them, so that the topology of the N-dimensional input space is preserved in the 2-dimensional output space. A common choice of neighborhood function $h(j,k)$ that computes the relation between two neurons O_j and O_k is:

$$h(j,k) = e^{-\frac{D^2}{2\delta(t)^2}} \tag{1}$$

where $k \neq j$ and D is the lateral distance between the neurons O_j and O_k in the output grid, and $\delta(t)$ is the time-dependent exponential decay.

Together, the update rule for the BMU is:

$$\Delta w_{ji} = \alpha(t)\, h(i,j)(x_i - w_{ji}) \tag{2}$$

given

$$||w_j - x_i|| \leq ||w_k - x_i|| \tag{3}$$

for all $j \neq k$ where x_i is the input vector and $\alpha(t)$ is the learning rate.

We can think of this learning rule as pulling the weight vector w_j associated with the BMU in the direction of the input vector x_i. All neurons in the same 'neighborhood' are also dragged along, but to a lesser extent.

4. METHODS

An effective Integrated System Health Management (ISHM) system has several key goals. The first is anomaly detection, which we achieve through a Self-Organizing Map-based approach as described in *Section 4.1*. Once an anomaly is detected, the second goal is to localize such deviation to the effected subsystems and/or components, which is a necessary step towards tracing its root causes. We achieve this goal through a supervised machine learning technique described in *Section 4.2*. The final goal is to quickly formulate and evaluate the most useful courses of action to mitigate the situation; our approach to such remediation assistance through Case-Base Reasoning techniques is described in *Section 4.3*.

Anomaly Detection

Once trained on nominal data (as described in *Section 3*), the SOM maps new data seen online to the most similar weight vector w_i of the output neurons O_i, using Euclidean Distance as the similarity metric. Recall that we refer to the winning O_i as the Best Matching Unit (BMU). The difference between the BMU's weight vector and the test point is the Minimum Quantization Error (MQE). A low MQE implies that the new sample closely aligns with a previously seen sample from the training data and is therefore nominal, whereas a higher MQE connotes that the point is anomalous, either because it contains a true fault or because it captures novel nominal behavior unseen during training.

For this preliminary research, we defined a range of nominal MQE scores and classified all samples as anomalous during testing if they fell outside that range. Tuning these thresholds is tricky in the absence of data representing faulty behavior. The range was chosen by re-running the training data through an already-trained SOM and setting the 1-percentile value and the 99-percentile value of the resulting MQEs as the lower and upper bounds respectively. Fine tuning this threshold is an area of future work.

While flagging novel nominal behaviors (in addition to faults) is a significant benefit of our approach, it is useful insofar as the SOM learns to recognize such novel behaviors as nominal in the future. Otherwise, future instances of such behaviors would continue to (incorrectly) be flagged as anomalous, causing unnecessary noise and false positives.

To this end, our future work includes a framework for quickly retraining SOMs on previously unseen nominal behaviors upon detection, allowing our ISHM tool to quickly adapt to the changing circumstances of deep-space exploration. As the SOMs learn a wider state space of nominal behaviors over time, the anomalies that they do detect will trend towards real faults or rare (and unknown) nominal behaviors; the corresponding alerts will thus prove more meaningful and useful to end-users.

Anomaly Localization

Reporting the key contributors to an anomaly aids users in localizing the causes of a system performance problem. ADTM uses a supervised learning approach to this task. This is based on the insight that behavior identified as anomalous can be treated as belonging to a distinct class from behavior identified as non-anomalous. The task then is to learn the features that accurately predict the class given the data.

Note that for this analysis, we are not concerned about the accuracy of anomaly identification, i.e., the external consistency of anomaly detection with respect to ground truth. Our goal instead is to determine the features that accurately separate two given segments of data. The data points labeled as anomalies are grouped into one class, and the weight-vectors learned by the SOMs during training on nominal data form the second class. This highlights one of the benefits of the SOM-based approach: the weight vectors of the SOM are effectively a reduced representation of the training data and lead to efficient storage for future analysis.

A number of supervised feature extraction approaches are available. We experimented with two such approaches in our research: Recursive Feature Elimination (RFE) [12], and Extra Tree Classifier [13]. RFE is an iterative technique that successively eliminates the set of least significant features for a classifier, until a desired number of features remains. It relearns a classifier during each iteration from a reduced set of features. It can be used with any supervised learning technique; we experimented with logistic regression. Extra Tree Classifier is a variant of the Random Forest approach [14]. We found that Extra Tree Classifier produced superior results compared to RFE with logistic regression.

Remediation Assistance

ADTM uses Case-Based Reasoning (CBR) as its fundamental remediation modeling and analysis mechanism. CBR is an artificial intelligence (AI) technique that aims to solve problems by analogy. With this approach, automated systems solve new problems by retrieving solutions to previous similar problems and altering them appropriately to meet current needs.

The field of CBR focuses on developing intelligent and efficient techniques for defining similarity metrics, retrieving cases based on these metrics, and modifying similar solutions to fit the target problem. The underlying inspiration is the analogical reasoning mechanism that humans often use to solve novel problems. While a problem itself may be novel, analogical reasoning helps situate it in the context of similar prior experiences and to discover a new solution by adapting prior solutions [15].

CBR consists of the following basic four steps:

- RETRIEVE the most similar cases(s)
- REUSE the information and knowledge in the retrieved case(s) to solve the problem.
- REVISE the solution used in the retrieved case(s)
- RETAIN the parts of this experience for future problem solving.

For ADTM, each case represents an operating mode, nominal or faulty, of the system. The core attribute of a case is a SOM trained on the data represented by the case. SOMs serve as the index into the case-bases for retrieving cases resembling the situation of interest. As with anomaly detection, we use the minimum quantization error (MQE) as the similarity metric for this purpose. That is, like before, once trained on nominal data, the SOM maps new target data to the most similar weight vector w_i of the output neurons O_i (i.e., the Best Matching Unit, BMU), using Euclidean Distance as the similarity metric.

Recall, we can interpret the set of weight vectors associated with each O_i as a condensed representation of the space of states seen in the training data. Thus, the difference between the BMU's weight vector and the target point of interest is the error, or MQE, which reflects the SOM's ability to categorize new input data into one of these known states. A low MQE implies that the target sample has characteristics very similar to a sample seen during training. The lower the MQE, the greater the similarity between the target observations and the SOM of comparison. Thus, the MQE metric can be used to RETRIEVE similar prior reference to support CBR. Given a set of observations of interest, ADTM will retrieve the closest set of cases, either nominal or faulty, based on the MQE of the corresponding SOMs.

A case will include other pertinent information such as an explanation of the operational mode and, where necessary, a recommended set of actions to address or mitigate the situation. Each case will also contain a label to say if it represents a nominal or a faulty condition.

Once found, the system REUSES the matching cases. It presents to the user the information associated with the case, including information pertinent to diagnosing and fixing the problems. Sometimes a matched case will be similar enough to the target problem that its solution can be reused without modifications. In other cases, the information from the reference case will help the users develop a REVISED solution on their own.

The revised solution, along with a new SOM trained on relevant data, becomes a new case for the model. This is the RETAIN phase of CBR. While human effort is still required to understand and solve truly novel situations, access to similar prior reference situations provided by the CBR approach will assist users in applying their analogical problem-solving skills to find a solution more effectively. This new knowledge then becomes a part of system memory and can be reused in the future using the RETRIEVE mechanism.

The distance of the target situation from the reference case will determine the extent to which the solution is transferable. ADTM will provide a similarity score for each retrieved case to help users judge its utility.

As mentioned, each new problem-solving episode becomes a new case for the model. Adding a new case involves training a new SOM for the sensor data covering the duration of the episode and adding supporting details about diagnosis and mitigation. In case the new episode is marked as a variation of an existing case, the system will merge the two by retraining the associated SOM with the new data and updating the supporting information. Thus, case-based reasoning enables an easy extension of the model based on new observations. The initial case base for a system will consist of a small set of cases representing nominal operations as well as known anomalies. This will grow over time as experiences build up.

5. Experiments and Discussion

We validated the performance of ADTM's anomaly detection and localization capabilities on a CubeSat (named "LabSat"), originally designed after the Ionospheric-Thermospheric Scanning Photometer for Ion-Neutral Studies (IT SPINS) project to study the nocturnal ionosphere. We leave validation of our remediation assistance capabilities via CBR to future work.

We divide the results of our experiment into the following three subsections: *Data Collection, Anomaly Detection Analysis, and Anomaly Localization Analysis*.

Data Collection

The LabSat was subdivided into three circuit boards. Board 1 was designated for power generation and storage (solar array simulators and batteries), while Boards 2 and 3 had redundant regulators and loads consuming power from Board 1.

The measurands for each board consisted of:

1. An outgoing voltage (in Volts) sensor for every component.
2. An outgoing current (in Amps) sensor for every component.
3. Three-state control switches connected to power-consuming loads on Boards 2 and 3. These loads were either powered by Solar Array 1 (SA1), Solar Array 2 (SA2), Battery 1 (BAT1), or Battery 2 (BAT2) from Board 1, and they were thus connected either to Power Bus 1, Power Bus 2, or neither (i.e., they were shut off).

Importantly, the LabSat connected to software that enabled us to insert a variety of hardware faults, which we used as test sets to validate ADTM's performance. Table 1 lists three such fault cases and includes descriptions of the faults as well as which boards were affected. The last test is nominal data that we used as a baseline to evaluate the performance of our approach on the fault cases.

We collected nominal data from a 15-minute run of the LabSat across all three boards, with telemetry collected once per second, and split the nominal data for each board into a training set and a test set. The former, as the name implies, was used to train a SOM on nominal behavior specific to that board (i.e., BOARD1-SOM, BOARD2-SOM, BOARD3-SOM), while the latter was used to compare against the fault data results; thus, the nominal test set gave us indication of how ADTM treated nominal and anomalous data differently during online monitoring.

Table 1. Description of the Test Sets for Boards 1, 2, 3

Test Set	Description	Boards Affected
Solar Array 1, 1 Cell Shorted	Solar Array 1 has one photovoltaic cell shorted, reducing voltage from ~13.1 V (nominal) to ~11 V.	Board 1. Any Board 2/3 components connected to Solar Array 1 are affected by reduced power.
CDH* Voltage Sensor Drifts High	The voltage sensor outgoing the CDH Load monotonically drifts high from sensor degradation over time. This is not a fault in the CDH Load itself, but rather with its voltage sensor.	Board 2
5V Regulator 1 - Failed	There is a redundant pair of 5V Regulators on Board 3. In this test case, the first "5V Regulator 1" has faulted.	Board 3
Nominal	Data of nominal operation was collected for each board and held out from the training set.	N/A

***CDH: Communication and Data Handling Load**

We trained three separate SOMs, one per LabSat Board, and ran our anomaly detection and localization analysis on the test sets described in Table 1. Table 2 describes the number of samples used to train each SOM and the number of measurands within each sample point. Data from the LabSat was collected at a rate of one sample point per second.

Table 2. Training Datasets used for LabSat SOMs

Subsystem	# Rows	# Measurands
Board 1	2304	20

Board 2	1154	20
Board 3	2022	22

Anomaly Detection Analysis

Running our anomaly detection algorithm on each test set produced the results displayed in Table 3. It is of note that the percentages of False Positives and True Positive are with respect to the target fault of interest for each test set in Table 1, but do not pertain to unknown anomalies that we also detected.

Table 3. Percent Anomalies Detected

Test Cases	BOARD1-SOM	BOARD2-SOM	BOARD3-SOM
SA1-SHORT1	99.5%	98%	4%
CDH-VDRIFT	0%	100%	3%
5VREG-FAIL	3%	0%	100%
NOMINAL	1%	1%	0%

Of critical note is the finding that ADTM identified *all known* faults in the data (i.e., there are no False Negatives). However, we do note the presence of false positives (red) for BOARD1-SOM and BOARD3-SOM. Upon investigation, we discovered that these false positives were because the test data deviated significantly from the nominal data used for training in ways that are *not faulty* (i.e., such deviation was independent of the faults triggered in Table 1), yet still important. Thus, ADTM was successful in detecting all novel behavior patterns, though only a subset represents real faults.

These results show that confounding variables influenced the MQE scores, and that our anomaly detection approach not only flagged known faults, but also flagged unknown novel behaviors unseen during training. This is not a weakness of our approach, but rather a strength, because it points to the ability of ADTM to identify *multiple* anomalies occurring at the same time. In other words, in addition to the target faults of interest described in Table 1, many of our test sets also contained activity that was previously unseen during training; these instances, in addition to the known faults, were successfully flagged as anomalous.

Anomaly Localization Analysis

For the datasets flagged as anomalous by each SOM, we used the Extra Tree Classifier (ETC) approach to identify the most salient measurands that contributed to each anomalous classification, ranked by importance. The feature and corresponding importance scores for the three test cases containing faults are listed in Tables 4-6, with the greater the score, the greater the importance. We selected the subset of measurands with a score of at least 10. The following analysis explores the ETC results in more detail for each test case.

Solar Array 1 - One Cell Shorted Fault—Recall from Table 1, this failure case involved Solar Array 1 (SA1) shorting 1 cell, which resulted in a drop in voltage from ~13.V to ~11V. The BOARD1-SOM correctly flagged over 99% of the samples from the Board 1 sensors of the SA1-SHORT1 data set. As shown in Table 4, the ETC identified two voltage sensors as the measurands contributing the most to deviation in the test data. The first monitors the first Power Point Tracker that is directly downstream from the faulted solar array (PPT1V), while the second monitors the faulted solar array itself (SA1V). This makes perfect sense. One photovoltaic cell shorting causes the outgoing Solar Array 1 Voltage to drop from 13.1 V (nominal) to 11 V. The voltage goes through PPT1 before traveling through the redundant set of batteries to Boards 2 and 3. Thus we would expect the voltage sensor monitoring PPT1 to experience an anomalous drop, just as the voltage sensor monitoring SA1 does.

Table 4. Solar Array Short Fault Localization

SOM	**Salient Feature**	**Importance Score**
BOARD1-SOM	PPT1 V	37.7
	SA1 V	26.9
	Switch Battery1 to PPT2	12.5
BOARD2-SOM	UNREG COMM V	20.9
	CDH Amps	20.7
	UNREG COMM on Bus2	13.7
	REG COMM on Bus2	10.6
BOARD3-SOM	N/A	N/A

The ETC also identified "Switch Battery1 to PPT2" as contributing to the fault, though to a degree that was less than half that of "PPT1 V." As the name suggests, the measurand "Switch Battery1 to PPT2" is the switch state connecting the first battery (Battery1) to the second Power Point Tracker (PPT2) that is directly downstream the second solar array (SA2). The measurand value is independent of the Solar Array fault and exists due to discrepancy between the switch state in the training data versus the test set. During training, the switch between Battery1 and PPT2 was off for the entire duration, while during this test run, it was on. This difference represents a novel nominal state in the test state that was also identified by the ETC as contributing to the anomaly. Again, this highlights the ability of our Anomaly Localization techniques to identify deviant measurands across multiple anomalies occurring simultaneously.

BOARD2-SOM also identified anomalies in this test set, though the measurands it flagged as deviant were unrelated to the fault triggered in the Solar Array. Unsurprisingly, this was because Board 2 loads were not connected to Solar Array 1, which experienced a shorted cell. Instead, they drew their power from Battery 1, which was connected to Solar Array 2 and in perfect health.

What BOARD2-SOM detected instead was novel behavior seen in the SA1-SHORT1 test set that was not captured by the training set. In particular, it identified differences in the features UNREG COMM V, CDH Amps, UNREG COMM on Bus 2, and REG COMM on Bus 2. The former two measurands refer to the incoming voltage and current sensors of the Unregulated Communication Load (UCL) and the Communication and Data Handling Load (CDH) on Board 2 respectively. The latter two measurands refer to the switch states connecting the UCL and the Regulated Communication Load (RCL) to Power Bus 2 (outgoing Battery 2), respectively. A brief explanation of the behavior of these measurands in the SA1-SHORT1 test data follows.

The UCL Voltage sensor differs significantly between the SA1-SHORT1 test set and the training set, as the load is in "Station Beaconing" mode during training, but in "Station Contact" mode during the fault. In Station Contact mode, the UCL transmits a message for 100 milliseconds before shutting off, whereas in Station Contact mode it transmits a message for 100 milliseconds every 4 seconds.

Similarly, the outgoing voltage sensor of the RCL ("REG COMM V") in the test set is far lower than that of the training set. Finally, the switch states connecting the UCL and RCL to Power Bus 2 are different between the training and test sets. While the UCL is connected to Power Bus 2 in training, it is disconnected from Power Bus 2 during testing.

Lastly, the ETC identifies CDH Amps as a significant contributor to this test case, though the CDH Amps measurand is similar in both the training and test cases. It is likely that BOARD2-SOM learned a positive correlation between Unregulated Communication Voltage ("UNREG COMM V") and CDH Amps from the training set, as the Unregulated Communication Load was always in Beaconing Mode (so UNREG COMM V sensor was high) while CDH was powered on (so CDH Amps sensor was also high). Thus, when the Unregulated Communication Load switched to Station Contact Mode in the test case (and thus to a lower voltage), the ETC likely identified the inverse correlation between "UNREG COMM V" (which went to 0V during standby) and "CDH AMPS" (which remained high) as salient, since the relationship was never seen before.

Had we had sufficient training data that covered both Station Beaconing and Station Contact modes, however, CDH Amps would likely not appear as a salient measurand. This points to the importance of obtaining training data that covers a wide breadth of operational states and retraining the SOMs on newly observed nominal behaviors so that it learns new nominal relationships between measurands and subsystem behaviors over time.

CDH Voltage Sensor Drift Fault—Recall, this test data contained a failure in the voltage sensor monitoring the Communication and Data Handling (CDH) load on Board 2, which drifted monotonically high due to degradation over time. While the ETC correctly identifies the CDH Voltage sensor (CDH V) as one of the top salient measurands distinguishing the fault data, it ranks several confounding variables with higher importance scores, as shown in Table 5. In particular, it identified "Switch Reg COMM on Bus 2" and "Switch Reg COMM on Bus 1" as the top two salient measurands distinguishing the anomalous test data from the training set.

Table 5. CDH Voltage Sensor Drift Fault Localization

SOM	**Salient Feature**	**Importance Score**
SOM 1	N/A	N/A
SOM 2	Switch Reg COMM on Bus 2	36.5
	Switch Reg COMM on Bus 1	20.0
	Reg 3V2 Amps	15.9
	CDH V	13.4
SOM 3	N/A	N/A

Again, this demonstrates that the ETC has identified confounding variables between the test and training data, which highlights our ability to detect multiple anomalies occurring at the same time. In this case, the Regulated Communication Load was strictly connected to Power Bus 1 during training and Power Bus 2 during testing. Because it is connected to Power Bus 2 during testing, the outgoing current sensor from the second 3.3V Regulator (i.e., "REG 3v2 Amps") on Board 2 (connected to Power Bus 2) also has higher current draw in the test case than in training. Presumably, if our training data spanned a wider breadth of nominal operating modes – in this case, the different permutations of switch states between the Regulated Communication load and the redundant power buses – the ETC would not have identified these switches as salient.

5V Regulator1 Fault—Recall, this fault case involved the first regulator on Board 3 (Regulator1) failing, such that it no longer transmitted power. The ETC correctly identified the failed regulator's voltage sensor ("Reg 5V1 V" in Table 6) as the most significant contributor of the anomaly. It also identified the voltage sensors monitoring the ADCS and CTIP loads on Board 3 as salient, which was as expected. Both these loads were receiving power through "Regulator 1" when it failed, at which point their voltage sensors dropped to 0V; thus, it makes perfect sense that the ADCS and CTIP voltage sensors were identified by the ETC as deviant.

Table 6. 5V Regulator Fault Localization

SOM	Salient Feature	Importance Score
SOM 1	N/A	N/A
SOM 2	N/A	N/A
SOM 3	Reg 5V1 V	38.6
	ADCS V	35.4
	CTIP V	17.1

6. Conclusions And Future Work

Because we were able to inject known hardware faults into the LabSat, the data served as a testbed for validating the accuracy of ADTM's anomaly detection and localization techniques. The LabSat experiment highlighted four important takeaways, which we have used to inform our future work. Principally, these are:

1. ADTM identified all known faults in the LabSat test cases as well as novel, nominal behavior unseen during training. Thus, our results show that ADTM is effective at detecting both known and unknown anomalies, in addition to capturing *multiple* confounding anomalies occurring at the same time.

2. It is important to train a SOM on nominal data that covers a wide and sufficient breadth of nominal operative conditions in order to reduce false positives with respect to actual faults.

3. Upon identifying previously unknown nominal behavior, it is important to retrain a SOM on the new behavior so that it learns to identify such instances as nominal in the future, thereby reducing false positives.

4. It is possible to use supervised machine learning techniques to identify the most salient sensors contributing to deviant behavior flagged by a SOM, including from multiple anomalies occurring at the same time.

In addition to these principal findings, we intend to extend ADTM capabilities in two significant ways in future research. The first is through remediation assistance via the Case-Base Reasoning (CBR) methodology we described in *Section 4.3*. When one or several SOMs flag data as anomalous, our CBR module will use the corresponding MQEs as indices to retrieve the closest reference cases.

Two scenarios are possible when retrieving similar cases. One is that a situation similar to the target exists in the case-base. The other is that the target situation is truly novel and significantly different from all the cases contained in the model. Our LabSat experiment indicates that the MQE is a useful metric for distinguishing between these two scenarios. When the target data is very similar to the training data, its MQE falls within the same range as the MQE for the training data. This is not the case when the target data deviates substantially from the training data.

Generalizing this to CBR case retrieval, ADTM will first use the MQE measure to find the closest reference case. It will then compare the MQE from the target data to predefined upper and lower thresholds of the training data MQE for the reference case to test whether the two situations are similar. If the target MQEs fall outside the threshold range, this is a sign that there is a significant deviation between the two. This also implies that no case can be found that matches the MQE for this target case; i.e., we have encountered a novel anomaly. On the other hand, if the target MQEs fall within the threshold range of the reference SOM MQE, then the target and reference situations are similar enough for effective knowledge transfer.

In addition to implementing ADTM remediation capabilities through CBR, we also intend to further ADTM's diagnosis capabilities through a hierarchical-SOM architecture in which SOMs are trained on varying degrees of system granularity. We expect this to assist in localizing faults to particular subsystems and components within those subsystems. For instance, in addition to training SOMs on each of the three circuit boards making up the LabSat, we can imagine an architecture that also leverages separate SOMs trained on each of the loads and sub-subsystems within the three circuit boards. Identifying the salient features of these finer-grained SOMs would enable end-users to more rapidly prioritize a smaller set of sensors behaving anomalously, which can expedite remediation activities further.

The most direct targets for transition of this proposed effort are the large number of various future manned and unmanned spacecraft that would significantly benefit from autonomous, intelligent health maintenance systems. NASA's Lunar Gateway is primary target of interest for the application of this technology.

Acknowledgements

This work was performed under a contract awarded and administered by National Aeronautics and Space Administration Agency (NASA).

References

[1] Bennouna O, Heraud N, Leonowicz Z (2012) Condition monitoring & fault diagnosis system for Offshore Wind Turbines. 2012 11th International

Conference on Environment and Electrical Engineering. doi: 10.1109/eeeic.2012.6221389

[2] Pascoal C, de Oliveira M, Valadas R et al. (2012) Robust feature selection and robust PCA for internet traffic anomaly detection. 2012 Proceedings IEEE INFOCOM. doi: 10.1109/infcom.2012.6195548

[3] Nassar B, Hussein W, Mokhtar M (2019) Space Telemetry Anomaly Detection Based on Statistical PCA Algorithm. In: Zenodo. http://doi.org/10.5281/zenodo.1109667.

[4] Gaddam S, Phoha V, Balagani K (2007) K-Means+ID3: A Novel Method for Supervised Anomaly Detection by Cascading K-Means Clustering and ID3 Decision Tree Learning Methods. IEEE Transactions on Knowledge and Data Engineering 19:345-354. doi: 10.1109/tkde.2007.44

[5] Iverson D, Martin R, Schwabacher M et al. (2009) General Purpose Data-Driven System Monitoring for Space Operations. AIAA Infotech@Aerospace Conference. doi: 10.2514/6.2009-1909

[6] Gao Y, Yang T, Xu M, Xing N (2012) An Unsupervised Anomaly Detection Approach for Spacecraft Based on Normal Behavior Clustering. 2012 Fifth International Conference on Intelligent Computation Technology and Automation. doi: 10.1109/icicta.2012.126

[7] Datta A, Mavroidis C, Hosek M (2007) A Role of Unsupervised Clustering for Intelligent Fault Diagnosis. Volume 9: Mechanical Systems and Control, Parts A, B, and C. doi: 10.1115/imece2007-43492

[8] Tian J, Azarian M, Pecht M (2014) Anomaly Detection Using Self-Organizing Maps-Based K-Nearest Neighbor Algorithm. European Conference of the Prognostics and Health Management Society 5

[9] Cottrell M, Gaubert P, Eloy C et al. (2009) Fault Prediction in Aircraft Engines Using Self-Organizing Maps. Advances in Self-Organizing Maps 37-44. doi: 10.1007/978-3-642-02397-2_5

[10] Kohonen T (1982) Self-organized formation of topologically correct feature maps. Biological Cybernetics 43:59-69. doi: 10.1007/bf00337288

[11] Fytilis, N., & Rizzo, D. M. (2013). Coupling self-organizing maps with a Naïve Bayesian classifier: Stream classification studies using multiple assessment data: A Combined Naïve Bayesian-SOM Classification Network. *Water Resources Research*, *49*(11), 7747-7762.

[12] Guyon I, Weston J, Barnhill S, Vapnik V (2002) Gene selection for cancer classification using support vector machines. Machine Learning 46:389-422. doi: 10.1023/a:1012487302797

[13] Geurts P, Ernst D, Wehenkel L (2006) Extremely randomized trees. Machine Learning 63:3-42. doi:10.1007/s10994-006-6226-1

[14] Breiman L (2001) Machine Learning 45:5-32. doi: 10.1023/a:1010933404324

[15] Carbonell, J. (1985). *Derivational analogy: A theory of reconstructive problem solving and expertise acquisition.* (No. CMU-CS-85-115). Carnegie-Mellon Univ Pittsburgh PA Dept of Computer Science.

BIOGRAPHY

Dr. Sowmya Ramachandran is a research scientist at Stottler Henke Associates where her research focuses on the application of artificial intelligence (AI) and machine learning to improve human performance in a broad range of domains such as intelligence analysis and tactical decision-making. This includes research on intelligent performance support systems and advanced training technologies. She led the development of an after-action review tool for large team training exercises that used machine learning techniques to automatically group chat stream messages into coherent topics. She is currently the PI on efforts to automate performance assessment in simulation-based training systems by generalizing from expert demonstrations of solutions using machine learning approaches. Dr. Ramachandran has led the development of Intelligent Tutoring Systems for a diverse set of domains, from training paramedics to training Information Systems Technicians in the U.S. Navy. She combines a rigorous background in artificial intelligence with deep knowledge of the science of human performance and training, both for individuals and teams. Her doctoral dissertation developed a novel approach to learning Bayesian Networks using a neural network training approach. The objective was to leverage the strengths of neural networks while endowing them with a semantic interpretation that is comprehensible to human experts. Dr. Ramachandran holds a Ph.D. in Computer Science from The University of Texas at Austin. She earned a Bachelor of Technology degree from the Indian Institute of Technology, Chennai, India.

Maia Rosengarten earned her B.A. in Computer Science and Political Economy at the University of California, Berkeley and currently works as an AI Developer at Stottler Henke Associates. She was a key engineer on EPS-MAESTRO (EPS Management through AdaptivE, autonomouS, faulT diagnosis and rescheduling Optimization), funded by NASA Ames Research Center (ARC) to intelligently and autonomously manage spacecraft Electrical Power Systems. The architecture implements fault detection and diagnoses modules and leverages NASA-funded and NASA-developed technologies. She is currently extending EPS-MAESTRO into a distributed architecture, DisMAESTRO, to manage subsystems across the entire spacecraft. She was also the lead engineer on ADTM (Anomaly Detection via Topological-feature Map), which leverages unsupervised machine learning techniques to significantly expand upon NASA's real-time and offline Integrated System Health Management capabilities for future deep-space exploration. Finally, she contributed to ViPER, an operational mission planning application developed for the U.S. Navy.

Christian Belardi received a Bachelor of Science in Computer Science from Cornell University's College of Engineering in May 2018. In addition to ongoing work on Self-Organizing Maps, Christian has contributed to a variety of projects spanning operational domains since joining Stottler Henke in August 2018. He is the lead engineer on a DARPA funded research effort for the development of machine learning and data fusion techniques to assess the pathogenic potential of novel bacteria. This effort runs in parallel with DARPA's Friend or Foe program and shares the ultimate goal of improving national security by furthering the state-of-the-art in biosurveillance/ biodefense technology. In another ongoing effort, Christian is working on a combination of statistical and knowledge-based techniques to characterize anomalous satellite behavior from multi-domain intelligence. Sponsored by the Air Force Research Laboratory, this project's objective is to help protect U.S. assets in space by improving space situational awareness. Christian was also instrumental in the development of a naval Inverse Synthetic Aperture Radar (ISAR) target recognition system for the Strategic Capabilities Office. This system uses deep learning to classify maritime targets in ISAR images.