Combining Model-Based and Feature-Driven Diagnosis Approaches – A Case Study on Electromechanical Actuators

Sriram Narasimhan[1], Indranil Roychoudhury[2], Edward Balaban[3] and Abhinav Saxena[2]

[1] *UARC, NASA Ames Research Center, Intelligent Systems Division, MS 269-4, Moffett Field, CA 94035, USA*
sriram.narasimhan-1@nasa.gov

[2] *SGT Inc., NASA Ames Research Center, Intelligent Systems Division, MS 269-4, Moffett Field, CA 94035, USA*
indranil.roychoudhury@nasa.gov
abhinav.saxena@nasa.gov

[3] *NASA Ames Research Center, Intelligent Systems Division, MS 269-3, Moffett Field, CA 94035, USA*
edward.balaban@nasa.gov

ABSTRACT

Model-based diagnosis typically uses analytical redundancy to compare predictions from a model against observations from the system being diagnosed. However this approach does not work very well when it is not feasible to create analytic relations describing all the observed data, e.g., for vibration data which is usually sampled at very high rates and requires very detailed finite element models to describe its behavior. In such cases, features (in time and frequency domains) that contain diagnostic information are extracted from the data. Since this is a computationally intensive process, it is not efficient to extract all the features all the time. In this paper we present an approach that combines the analytic model-based and feature-driven diagnosis approaches. The analytic approach is used to reduce the set of possible faults and then features are chosen to best distinguish among the remaining faults. We describe an implementation of this approach on the **Fl**yable **E**lectro-mechanical **A**ctuator (FLEA) test bed.

1   INTRODUCTION

Electro-mechanical Actuators (EMAs) are used in a variety of aerospace applications, from civilian airliners to robotic spacecraft (Blanding, 1997, Jensen et al, 2000). Actuators are safety-critical components of an aerospace system and an undetected actuator fault can lead to serious consequences. EMA fault diagnosis poses an interesting research problem as these actuators are composed of electrical, electronic, and mechanical subsystems, which results in intricate failure modes. Any fault in these sub-assemblies needs to be successfully and efficiently detected, identified, and isolated using a limited set of sensor signals available.

EMAs, being relatively new to the field of aerospace, have not yet been deployed for a long enough time, or in large enough numbers, to accumulate reliable fault statistics. Most of the commercial airliners in service still rely on hydraulic actuators for their primary flight surfaces, landing gear, and other major components. Small EMAs are sometimes used for secondary functions, such as trim tab actuation, and spoiler or speed break deployment. The situation is similar in military applications, where most aircraft in the active inventory rely on hydraulic actuation. However, there are efforts currently under way to deploy electro-mechanical actuators in utility roles (landing gear, aerial refueling doors, and weapons bay doors) in the future models of some of the new designs, such as the F-35 Joint Strike Fighter. Recently designed commercial aircraft, such as Boeing 787 or Airbus 380, are also starting to use more EMAs in the roles traditionally reserved for hydraulic systems. On the Boeing 787, for example, EMAs, in addition to spoilers and trim actuation, operate landing gear breaks, and are part of the environmental control system. Space vehicles currently use EMAs for functions such as positioning of antennas and movement of robotic arms, with some of the future rocket launcher designs intending to use EMAs for their thrust vector control. The challenges with obtaining performance statistics in this domain are that only a small number of each vehicle type is usually built, and the onboard actuators are rarely used for more than a few hours total throughout the entire service life of a vehicle.

Some typical faults in EMAs are *jams* in ball bearing paths, *spalls* (development of indentations in metal surfaces at high stress contact points), *winding shorts*, and common *sensor faults* like bias, drift and scaling. It is possible to derive analytical relations for the electrical, mechanical and thermal behavior of the EMAs, however the vibration behavior is much too complicated to model precisely. Although electrical, mechanical and thermal data can be used to isolate most of the faults, the vibration data is essential for diagnosing some of the faults (like spall) as well as for distinguishing between sensor and component faults. As a result, a real-world diagnostic solution for EMAs would have to somehow combine the information available from the two types of data in order to isolate all of the common fault types.

In this paper we present an approach that uses analytical models for the electrical, mechanical and thermal properties in conjunction with a classifier based on features derived from vibration data. Given that extraction of features from vibration data which is usually sampled at high speeds is computationally intensive; our approach uses a "lazy" feature extraction approach where necessary features are extracted only when needed. This approach was motivated by a need for a diagnosis engine that would be applicable to the EMA test stands at NASA Ames Research Center. Section 2 presents the diagnostic problem for EMA including a look at some common fault modes and sensors. Section 2 also presents the various EMA test stands at NASA Ames Research Center, with a focus on the **Fl**yable **E**lectro-mechanical **A**ctuator (FLEA) test stand. Section 4 describes our hybrid (combination of model-based and data-driven methods) approach to diagnosis which has general applicability but is focused on the EMA domain (specifically the FLEA). Section 4 discusses the application of our diagnosis method on the FLEA, and Section 5 concludes the paper and presents future work

2 Electromechanical Actuator Diagnosis

EMAs are characterized by rotary motion provided by a motor being converted into linear displacement of the actuator shaft via screws and/or gears to which the motor is attached. Of specific interest to us is the ball screw type of EMA (referred to as EMA henceforth) which is commonly used in aerospace applications. A threaded shaft provides a spiral raceway for ball bearings which act as a precision screw. The ball assembly acts as the nut while the threaded shaft is the screw.

Given this configuration, we can identify four main categories faults that may occur in EMAs – mechanical,

motor, electrical/electronic, and sensor. Table 1 lists the most common fault modes in EMAs. This information comes primarily from the following sources: Failure Modes, Effects, and Criticality Analysis (FMECA) information provided by Moog Corporation; published industrial information (Bodden *et al.* 2007, Gokdere *et al.* 2005); information from the US military reports (AIR5713, 2006); as well as from our general survey of publications related to actuator diagnostics (Arvallo, 2000; Tesar and Hvass, 2004). More details can be found in (Balaban *et al.* 2009a).

Table 1: Common EMA Fault Modes

Component	Fault	Diagnosis Approach
Actuator	Jam	Physics
	Spall	Hybrid
Motor	Winding short	Physics
Connectors	Degraded operation	Data
	Intermittent contact	Data
Stator	Stator coil fails open	Physics
	Insulation deterioration / Wire chafing	Data
Resolver	Coil fails open	Physics
	Intermittent coil failures	Data
	Insulation deterioration / wire chafing	Data
Sensor	Bias	Physics
	Scaling	Physics
	Drift	Data
	Noise	Data
	Intermittent dropout	Physics
Power supply	Short circuit	Hybrid
	Open circuit	Hybrid
	Intermittent performance	Data
	Thermal runaway	Hybrid
Controller capacitors	Dielectric breakdown	Hybrid
Controller transistors	Dielectric breakdown	Hybrid
Wiring	Short circuit	Hybrid
	Open circuit	Hybrid
	Insulation deterioration / wire chafing	Data
Solder joints	Intermittent contact	Hybrid

Typical sensors used in this domain are load cells, position sensors (encoders and potentiometers for example), current sensors, thermocouples, and accelerometers. (Balaban *et al.* 2009b) provides an in depth study and classification of faults in these types of sensors. The general categories of sensor faults are identified as bias, scaling, noise, drift and intermittent dropouts.

Three EMA test stands have been built at NASA Ames Research Center in order to allow researchers to develop and test diagnostic and prognostic technologies for this type of actuators. The first one, Actuator Prognostics Experiment (i.e. APE) is a large stand capable of testing actuators of a wide range of sizes and configurations with a loading mechanism capable of applying up to 5 metric tons of force. The stand is equipped with a comprehensive sensor suite measuring vibration, temperature, load, currents, and the exact

position of the test actuator piston. The control system allows execution of custom load and motion profiles, while the data acquisition system provides sampling rates of up to 64 kHz. The second stand is based on a section of a Boeing 727 wing and is meant to provide a realistic platform to study and illustrate the impact of various actuator fault modes on a control surface of an aircraft. The original hydraulic actuator driving the aileron was replaced with a Moog EMA and a control system similar to the one on the APE was designed and implemented. Work is currently under way to build a load system for the aileron section that can realistically simulated the aerodynamic loads applied to that control surface during the various phases of flight. As fault injection techniques are developed and tested on the APE, they will be transferred onto this test stand. In this paper we will focus on the third stand named the **Fl**yable **E**lectromechanical **A**ctuator (FLEA) illustrated in Figure 1, which is by far the smallest and lightest of the three and is meant for airborne testing of EMA fault-injection techniques and diagnostic/prognostic algorithms.

Figure 1: The FLEA

2.1 THE FLEA

The key idea of FLEA is to design, build and fly a self-contained, lightweight EMA test bed. It is compact enough to fit into a standard 19-inch equipment rack present on some aircraft to allow for it to be flown aboard. It is a largely self-contained unit. The only external interfaces required are those for the aircraft data bus and power.

The FLEA is designed to be flown on different types of aircraft including the C-17 transports and the UH60 helicopter (two C-17 flights have already been completed). Data from the operation of one of the control surface actuators on those aircraft is fed to the FLEA control system which would exercise the FLEA in a manner equivalent to motion performed and loads experienced by the aircraft actuator.

The FLEA contains three actuators: two test actuators and one load actuator. One test actuator is healthy and the other test actuator contains a fault. Control is switched from the nominal to the faulty actuator during experiments, thus providing the fault injection capability for the test stand without having to modify the actuator in flight. The load actuator is used to emulate the dynamic load conditions experienced by actual aircraft actuators. Coupling of test actuators (one at a time) to the load actuator is accomplished via electro-magnets. Figure 2 illustrates this setup. We already have test actuators that have ball screw jam and spalls injected into them. We are also in the process of acquiring a programmable rheostat that would let us artificially inject winding short faults that progress over time.
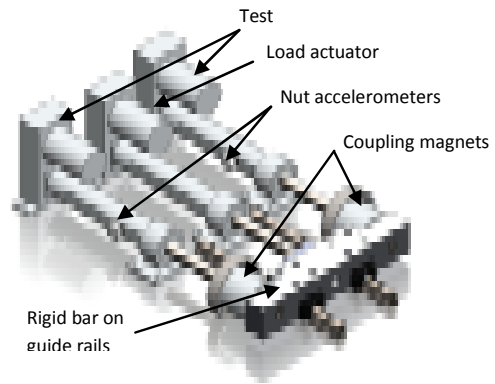

Figure 2: FLEA Actuator Coupling System

A comprehensive sensor suit has been installed on the FLEA as described in Table 2. It is also possible to inject faults, such as sensor faults, by altering the data coming from the data acquisition system.

Table 2: FLEA Sensor Suite

Sensor	Location
Load cell	Between the load actuator and the test actuator coupling
Accelerometer	On the nut of the ball screw
Thermocouple	On the ball screw nut and motor housing
Rotary encoder	On the test actuator motors
Linear potentiometer	Along the load actuator screw
Voltage Sensor	Motor controller boards
Current sensor	Motor Controller Boards

The system software consists of an actuator control system, data acquisition system, flight interface system, a diagnostic system, and a prognostic system. All of these are implemented in LabVIEW™ on a Windows XP operating system. However the underlying algorithms for the diagnoser and prognoser are implemented in MATLAB™. The diagnoser is responsible for monitoring the sensor data and determines if any faults are present in the system. The details of the diagnostic algorithm that form the core of this paper, which is implemented in MATLAB, are described in Section 4.

3 DIAGNOSIS APPROACH

Our diagnosis approach combines model-based analytical redundancy relations with a feature-driven diagnosis approach for the detection and isolation of single faults in EMA-like systems, containing slow and high speed sensors. First, we discuss the model-based and feature-driven methods that we will be using, and then we describe how these can be combined to synthesize a diagnosis engine for EMA.

3.1 Model-based method

We adopt the TRANSCEND diagnosis approach (Mosterman and Biswas, 1999) as our model-based diagnoser. Figure 3 illustrates the TRANSCEND diagnosis architecture. The *observer*, takes as inputs, the control inputs sent to the system, and the measurement readings obtained by the sensors to track the system dynamics, and estimate the unobservable system states. The observer can be implemented as a particle filter (Arulampalam *et al.*, 2002), which makes estimates using a system of first order differential state-space equations systematically derived from the system *bond graph* (BG) model (Karnopp, *et al.*, 2000). The BG modeling paradigm provides a framework for domain-independent, energy-based, topological, lumped-parameter modeling of energy-exchange mechanisms in physical processes. The nodes of a bond graph represent energy storage (capacitors, *C*, and inertias, *I*) dissipation (resistors, *R*), transformation (gyrators, *GY*, and transformers, *TF*), and input-output elements (effort sources, *Se*, flow sources, *Sf*, effort detectors, *De*, and flow detectors, *Df*), as well as two connection elements or junctions: 0- and 1-junctions (which represent parallel and series connections in the electrical domain). The connecting edges, or *bonds* (drawn as half arrows), define energy pathways between elements. Each bond has an associated effort, *e*, and flow, *f*, variables, whose product defines the power transferred through the bonds.

For *fault detection*, TRANSCEND uses a statistical Z-test (Mosterman and Biswas, 1999) on each sensor output to determine whether the deviation of a sensor output from its nominal expected value is statistically significant, taking into account sensor noise and other uncertainties. Once a significant deviation is detected in any one measurement, the *symbol generation* module is initiated, and *every* measurement residual, $r(t) = y(t) - \hat{y}(t)$, where $y(t)$ is an observed
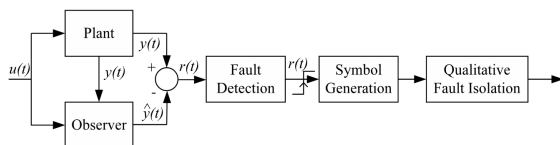
measurement, and $\hat{y}(t)$ is the measurement estimate calculated based on the state estimates obtained from the observer, is converted into qualitative +, − and 0 symbols, based on whether or not the observed measurement is above, below, or at its expected nominal value, respectively.

The detection of a fault triggers the *qualitative fault isolation* module to determine the *fault hypotheses,* i.e., all possible system parameters and their direction of change that could explain the observed measurement deviation from nominal. To generate the fault hypotheses, TRANSCEND use a qualitative diagnostic model called the *Temporal Causal Graph* (TCG) (Mosterman and Biswas, 1999), which is essentially a signal flow graph, whose nodes represent system variables, edges denote causality information, and edge-labels denote how one variable affects another, either immediately, or over time. Fault hypotheses are generated by propagating the first observed deviation backwards through the TCG.

Then, for each fault hypothesis, the direction of change is propagated forward along the TCG to generate *fault signatures*, i.e., an ordered set of two 0, +, or − symbols, one for magnitude, and the other for slope, which represent how each measurement residual would deviate if that fault was the only fault in the system (in this work, we restrict our discussion to single faults).

After the fault signatures are generated, qualitative diagnosis involves comparing an observed deviation with the expected fault signatures of each fault for that measurement, and removing any fault hypothesis that does not explain the observed deviation from consideration, thereby refining the fault hypotheses set.

3.2 Feature-driven diagnosis

Feature-driven diagnosis works by extracting a set of features or condition indicators from sensor data that can distinguish between fault classes of interest, detect and isolate a particular fault at its early initiation stages. These features should be fairly insensitive to noise and within fault class variations. Appropriate feature extraction lays the foundation of a successful (accurate and reliable) diagnostic system. A diagnostic feature is a system parameter (or derived system parameter) that is sensitive to the functional degradation of one or more components contained in the system. Diagnostic features can be used to predict the occurrence of an undesirable system event or failure mode. For a successful practical implementation it is desirable that features not only be computationally inexpensive but also be explainable in physical terms. Furthermore, they should be characterized by a) large between-class
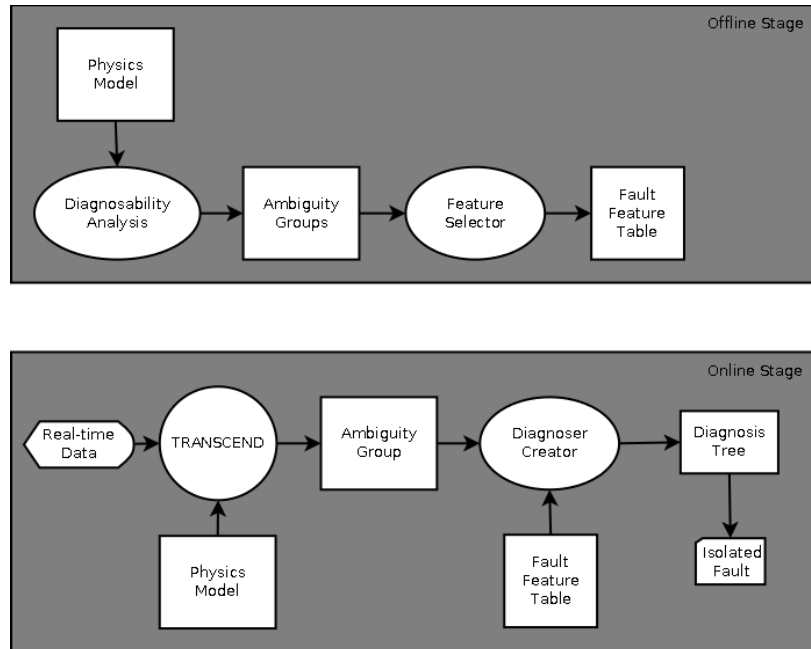

Figure 3: TRANSCEND Diagnosis Architecture

Figure 4: Hybrid Diagnosis Approach

and small within-class variance, b) should be fairly insensitive to external variables like noise, and c) should be uncorrelated with other features. Expert knowledge and analysis of nominal and faulty data is used to select features of interest and also to train the classifier to perform optimally.

Features from the accelerometer data such as various statistical moments, kurtosis, peak values, rms, and spectral energies are some of the candidates most commonly used in analysis of rotating machinery. Other possible features are extracted through coherence and correlation calculations. When the information is shared between the time and frequency domain, features in the wavelet domain offer an appropriate tradeoff between the two domains. If multiple features for a particular fault mode are available, it might be desirable to combine or fuse uncorrelated features to enhance the fault detectability. Indirect features may be developed using other measurements, such as those for current, voltage, temperature, position, and load (Balaban *et al.,* 2009a). The next step is the feature selection process which is largely application dependent. A smaller subset of features is sought, for a particular class of fault modes, from a large candidate set that possesses properties of fault distinguishability and detectability while achieving a reliable fault classification in the minimum amount of time. The features are then integrated into a classifier that can isolate the fault based on the feature selected. One example of such a classifier is the Artificial Neural Network approach presented in (Balaban *et al.* 2009a) with features as the inputs and the isolated fault as the

output. Sample data is used to train the artificial neural network (ANN) and estimate the internal parameters. At run time all the computed features are supplied to the ANN which then returns the possible fault in the system.

3.3 Hybrid Diagnosis Approach

The model-based approach works well when we are able to derive analytical models for all the faults under consideration. However it may not be possible to do so for all faults. On the flip side the feature-driven approach requires a lot of data under varying experimental conditions for training the classifier. Additionally when the classifier has to consider all faults and other experimental conditions the size and complexity of the classifier becomes intractable.

We present a hybrid method that combines these two approaches as illustrated in Figure 4. Our approach consists of an offline and an online stage.

Offline Stage

In the offline stage we first derive a BG model of the EMA system. We then perform a qualitative diagnosability analysis on this model. The BG model can be used to generate qualitative signatures for all faults represented by changes in the parameters of the BG. By comparing the qualitative signatures we can identify the ambiguity groups (groups of faults that have the same fault signatures). These groups represent faults that need to be disambiguated using the feature-driven approach. For each ambiguity group a set of features are identified (using domain knowledge or by

5

experimentation) that might contain diagnostic information to disambiguate faults in that group. This results in a fault feature table that indicates how specific features are influence by faults.

Online Stage

The online stage is performed in two phases. In the first phase the TRANSCEND approach described earlier is used to observe the system, detect and qualitatively isolate fault ambiguity groups. In the second phase the isolated ambiguity group triggers the selection of rows from the fault feature table. These rows correspond to the faults in the ambiguity group. The selected sub-table can then be converted to a diagnoser tree using the measurement selection procedure detailed in (Narasimhan *et al.* 98).

The nodes of this diagnosis tree are groups of faults and the edges represent specific values for features. The root of the tree is the ambiguity group that we start with. At each level of the tree features are selected that partition the ambiguity group in the most balanced fashion. This can be formally specified as the partitioning with the least difference between the largest and smallest partition. Once the best feature has been identified, the ambiguity group is partitioned into sub-groups corresponding to the possible values for the selected feature (one sub-group for each possible feature value). For each of the sub groups of size greater than 1 the next best feature is selected which creates the most balanced partitions. This process continues until only sub-groups of size 1 are left or there are no more features left to select. If there are any sub-groups of size greater than 1 then it indicates indistinguishable faults.

Once the diagnoser tree has been identified, fault isolation is performed by walking down this tree from the root node. First the feature associated with edges from the root node is computed. Depending on the value for the feature, the corresponding partition of the ambiguity group becomes our current belief. Again we compute the feature associated with that node to further reduce the size of the current ambiguity group until we reach a leaf node of the tree. At this point we are done with the fault isolation and the final ambiguity group can be reported as the diagnosis.

The construction of the diagnoser tree can actually be done offline for the set of identified ambiguity groups in order to reduce some computation at run time. However if more computational power is available a lazy approach might be best. In the lazy approach, the entire diagnoser tree is not constructed. Rather only the first best feature is identified. This feature is then
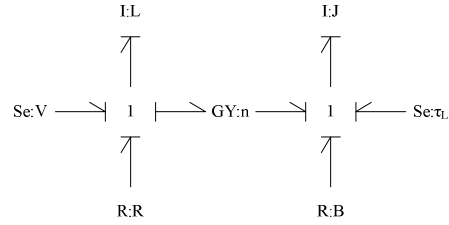

Figure 5: Bond graph model of FLEA

computed and the ambiguity group reduced. We repeat this procedure (computing only first best feature) for the current ambiguity group. Again this process is repeated until a single fault has been isolated or all features have been computed.

4 Application of Hybrid Diagnosis Approach on FLEA

The key steps to applying our hybrid approach are the development of physics-based model, determining the fault feature table, and the creation of the run time architecture. In this section we present our efforts to implement this diagnosis engine for the FLEA.

4.1 Model Parameter Estimation

As explained earlier, our hybrid diagnosis approach makes use of a physics-based BG model of the system for estimating nominal system behavior and also for qualitative fault hypothesis generation and refinement. Figure 5 shows the BG model of the load or test EMA in the FLEA. The EMA is modeled as a DC motor input voltage, V, winding inductance, L, and winding resistance, R. The GY element models the conversion of electrical energy to mechanical energy. The mass of the shaft is denoted by the inductance, J; the damping coefficient is represented by the resistor, B; and τ_L represents the opposing load torque.

The state-space equations can be systematically derived from the bond graph model, and are given below.

$$\frac{di}{dt} = \frac{1}{L}\left(V - Ri - K_m\omega\right),$$

$$\frac{d\omega}{dt} = \frac{1}{J}\left(K_t i - B\omega - \tau_L\right),$$

where i denotes the current drawn by the actuator, ω denotes the angular velocity, and K_t and K_m denotes the torque and motor constants, respectively.

One of the first steps for model-based diagnosis is to estimate the model parameters. To this end, we first collected nominal data by running the FLEA under different load profiles. Then we developed a MATLAB Simulink model of the EMA that accepts the same inputs that were given to the FLEA, and estimate the

variables that are measured by the available sensors. Note that in this estimation scheme, all but the accelerometer readings were used, since, as explained earlier, the modeling of accelerometers is non-trivial. Finally, we ran an optimization script in MATLAB to estimate model parameters that would minimize the error between the actual and predicted values of the available measurements. The estimated parameter values were then included in the state-space equations of the particle filter observer to generate high fidelity estimates of unknown hidden states.

4.2 Feature Driven Approach Setup

Table 3 indicates the fault-feature signature matrix for some EMA faults and the two accelerometer features. In this work, the signatures represent the fact as to whether or not a feature will be affected by a fault, and denoted by a '1' or '0', respectively. In this work, we consider the standard deviation of each accelerometer reading as a feature.

Table 3: Fault Feature Table for the FLEA

Faults	Accelerometer 1 Standard Deviation	Accelerometer 2 Standard Deviation
Spall	0	1
Jam	0	1
Ball nut friction	0	1
Rotor shaft eccentricity	1	1
Accelerometer 1 sensor fault	1	0
Accelerometer 2 sensor fault	0	1
Backlash	1	1
Winding short	0	0

Based on this fault-feature signature matrix, given the present set of fault hypotheses, we generate a tree data-structure that gives the subset of features, and the sequence in which they should be used to refine the fault hypotheses set to (ideally) a singleton set the fastest.

4.3 Runtime Architecture

The control, data acquisition, and user interface modules for FLEA are implemented using LabVIEW. Our hybrid diagnoser is implemented using MATLAB and Simulink, and is invoked from LabVIEW using MATLAB script nodes for online diagnosis. The LabVIEW data acquisition module acquires the data continuously and sends this data to the diagnoser (which has been initialized when the LabVIEW control software starts). An observer synthesized from the bond graph models uses this data to determine if a fault has been detected. The qualitative fault isolation code also implemented in MATLAB attempts to isolate the faults resulting in an ambiguity group. The next best feature selector as well as the feature extractor are also implemented in MATLAB. The fault detection flag as

well as the ambiguity group as it is being refined is communicated back to the LabVIEW user interface module.

4.4 Experiment Plans & Results

FLEA consists of two test actuators, one of which is nominal and other being faulty. Different kinds of faulty actuators (jammed, spalled etc.) can be switched in to mimic the occurrence of the corresponding fault. The nominal actuator will be active for certain duration of time and then the control system will switch operation to the faulty actuator (without the knowledge of the diagnoser). The diagnoser is expected to continuously monitor the data and determine if and when the switch occurred (identifying it as an occurrence of a fault in the nominal actuator rather than a switch event).

We ran the above-mentioned diagnoser on a set of pre-defined scenarios with varying position (sine, trapezoidal, triangular, sine sweep) profiles and load (constant load between -70 and +70 pounds) profiles. Subsets of these scenarios were used for fault runs which included hardware-injected faults (jam, motor failure, and spall) and software-injected faults (sensor faults). The results are listed in Table 4.

Table 4: Diagnoser Results

Fault	Scenarios	Correct	Accuracy
Nominal	134	133	99.25
Current Bias	15	15	100
Current Dead	15	15	100
Current Drift	15	15	100
Position Sensor Failure	21	13	61.9
Current Scaling	15	15	100
Jam	15	10	66.67
Motor Failure	15	15	100
Spall	15	15	100
Temp Bias	15	15	100
Temp Dead	15	15	100
Temp Drift	15	15	100
Temp Scaling	15	15	100
Total	**320**	**306**	**95.625**

The FLEA has also been involved in two flight experiments on the C-17 aircraft. We have collected data on the nominal and jammed actuators on those

flights. We have also completed 2 flight experiments on the UH-60 helicopters. On the latter of these flights a preliminary version of the diagnoser was run on board. We plan to continue flight experiments on the UH-60 helicopters with the diagnose running online. Several faults will be injected to test the accuracy of the diagnostic system.

5 CONCLUSIONS AND FUTURE WORK

This paper presented a diagnostic approach that supports the fusion of model-based and data-driven methods. We also introduced a portable test stand called FLEA that can connect to aircraft data buses and mimic operation of control surface actuators. We are in the process of building a real-time diagnostic system for the FLEA. The FLEA also allows us to "inject" faults into actuators in a non-intrusive manner. We have flown on board the C17 aircraft twice so far. We plan to run several more experiments on C17 aircraft and UH 60 helicopters with different faulty actuators and test the performance of the diagnostic system.

In the future we would like to make this test stand available for other diagnostic algorithm developers as a means to mature and test their technologies. We are planning to continue our current flight test program and potentially extend it to other types of aircraft. We also want to "inject" several different kinds of faults on the FLEA to test the coverage of the diagnostic system. As mentioned earlier NASA Ames Research Center also has two other actuator test stands (APE and Wing) which are not portable but can be used to assess the performance of our diagnostic algorithms.

ACKNOWLEDGMENTS

REFERENCES

(AIR5713, 2006) AIR5713, In-Service Reliability Data of Continuously Active Ballscrew and Geared Flight Control Actuation Systems. *AIR5713, SAE Aerospace,* 2006.

(Arulampalam *et al.,* 2002) M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking, *in IEEE Transactions on Signal Processing, vol. 50, no. 2, pp.174-188,* 2002.

(Arvallo, 2000) A. Arvallo. Condition-based Maintenance of Actuator Systems Using A Model-Based Approach. *PhD Dissertation, University of Texas, Austin,* 2000.

(Balaban *et al.,* 2009a) E. Balaban, A. Saxena, P. Bansal, K. Goebel, P. Stoelting, and S. Curran. A Diagnostic Approach for Electro-Mechanical Actuators in Aerospace Systems, *in Proceedings of IEEE Aerospace conference,* 2009.

(Balaban *et al.*, 2009b) E. Balaban, A. Saxena; P. Bansal, K. F. Goebel, S. Curran. Modeling, Detection, and Disambiguation of Sensor Faults for Aerospace Applications, in *Sensors Journal, IEEE, vol.9, no.12, pp.1907-1917,* 2009.

(Blanding, 1997) D. E. Blanding. An Assessment of Developing Dual Use Electric Actuation Technologies for Military Aircraft and Commercial Application, in *IECEC Energy Conversion Engineering Conference,* 1997.

(Bodden *et al.*, 2007) D. S. Bodden, S. Clements, N. Schley *et al.* Seeded Failure Testing and Analysis of an Electro-Mechanical Actuator, *in IEEE Aerospace Conference, Big Sky, MT, 2007, pp. 1-8,* 2007.

(Gokdere *et al.,* 2005) L. U. Gokdere, S. L. Chiu, K. J. Keller, *et al.* Lifetime control of electromechanical actuators, *in IEEE Aerospace Conference, Big Sky, MT, 2005, pp. 3523-3531.* 2005.

(Jensen *et al.,* 2000) S. C. Jensen, G. D. Jenney, and D. Dawson. Flight Test Experience with an Electromechanical Actuator on the F-18 Systems Research Aircraft. *IEEE Digital Avionics Systems Conference,* 2000.

(Karnopp *et al.,* 2000) Karnopp, D. C., Margolis, D. L., and Rosenberg, R. C., (2000), *Systems Dynamics: Modeling and Simulation of Mechatronic Systems,* 3rd ed. New York, NY, USA: John Wiley & Sons, Inc.

(Mosterman and Biswas, 1999) P. J. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions, *in IEEE Transactions on Systems, Man and Cybernetics, Part A, vol. 29, no. 6, pp. 554-565,* 1999.

(Narasimhan *et al.*, 1998) S. Narasimhan, P. J. Mosterman, and G. Biswas. A Systematic Analysis of Measurement Selection Algorithms for Fault Isolation in Dynamic Systems, in *Proceedings of the International Workshop on Diagnosis Principles, pp. 94-101, Cape Cod, MA,* 1998.

(Tesar and Hvass, 2004) D. Tesar and P. B. Hvass, Condition Based Maintenance for Intelligent Electromechanical Actuators. *Technical Report, University of Texas, Austin,* 2004.