Preprints of the 8th IFAC Symposium on Fault Detection,
Supervision and Safety of Technical Processes (SAFEPROCESS)
August 29-31, 2012. Mexico City, Mexico

Fault Isolation for Spacecraft Systems: An Application to a Power Distribution Testbed

Joshua D. Carl * **Daniel L.C. Mack** * **Ashraf Tantawy** *
Gautam Biswas * **Xenofon D. Koutsoukos** *

* *Institute for Software Integrated Systems, Vanderbilt University, Nashville, TN 37235* {*carljd1, dmack, tantawam, biswas, koutsoxd*}*@isis.vanderbilt.edu*

Abstract: Modern electrical power disribution systems play a critical role in system operations. Therefore, early fault detection and isolation is essential to maintaining system safety and avoiding catastrophic failures. This paper discusses a fault isolation scheme based on a qualitative fault signature-based isolation mechanism that applies to abrupt, incipient and intermittent faults in the system. We discuss the isolation algorithms for a combination of these faults, and demonstrate their performance on a set of test cases generated from a NASA Ames spacecraft power distribution testbed. Our results show good isolation accuracy with 103 out of 134 faulty scenarios isolated correctly. Most of the isolation errors can be attributed to errors in the detection scheme.

Keywords: Fault Detection, Qualitative Fault Isolation, Fault Diagnosis

1. INTRODUCTION

Fault isolation, i.e., establishing the true cause for an observed failure or degradation in a system, is key to maintaining efficient system operations and ensuring system safety. Isolation algorithms combine information from multiple sensors to generate diagnostic conclusions. A combination of an observer and a fault detection scheme is used to determine and represent deviations from nominal in the sensor in qualitative form (i.e., $\pm$ – increase/decrease) . The fault isolation methodology uses a qualitative fault signature scheme to generate diagnostic conclusions. This paper develops a combined qualitative fault isolation and identification methodology using the framework presented in [Mosterman and Biswas, 1999], and further developed in [Daigle et al., 2008].

We demonstrate that a qualitative fault signature is an effective means to isolate faults in a simplified spacecraft power distribution system. We test our approach on a set of test cases developed by NASA Ames in the context of the DX10 diagnosis competition [Kurtoglu et al., 2010]. Each scenario in the test set represents a nominal case where no fault is injected into the scenario, or has one of three different fault profiles (abrupt persistent, abrupt intermittent, and incipient profiles) injected into one of the system components or sensors. We evaluate our approach on all 159 test scenarios generated for the competition. Our diagnosis scheme identified the correct fault profile and component in 73% of the faulty scenarios, and the correct component but not the correct fault profile in 84% of the faulty scenarios. These results gave us a second place finish in the competition.

The rest of the paper is organized as follows. Section 2 introduces the case study based on the DXC'10 competition. Section 3 discusses the computational system architecture for detection and diagnosis in continuous systems. In section 4 we define the detection and isolation problem. Section 5 discusses our implementation approach. Section 6 presents the case study results. We conclude with a summary of our work and analysis of the results.

2. CASE STUDY OVERVIEW

The case study is organized around a set of test cases that use fault scenario data generated from the NASA ADAPT-Lite Electrical Power System (EPS). The test cases were used to evaluate a number of algorithms submitted to the DXC'10 diagnosis competition. A complete description of the DXC'10 competition format can be found in [Kurtoglu et al., 2010] and the algorithm evaluation metrics in [Kurtoglu et al., 2008].

The EPS supplies power to spacecraft systems and payloads. The EPS schematic in Figure 1 shows a battery connected to a load bank through a set of switches, circuit breakers and an inverter. Since the dynamics of the inverter (a fast switching system that converts DC voltage to AC) and switching elemetns going on and off were not a factor in this competition, the rest of the system represents a power source that is connected to resistive loads that can be reconfigured using the switches. Therefore,the system operates in steady state and the detected measurement deviations correspond to steady state fault profiles.

The competition was sponsored by researchers at NASA Ames, and was designed to mimic the operations of a real spacecraft power distribution system. Therefore, to support online analysis a fast and accurate diagnosis and isolation algorithm was a requirement to do well in the competition. There were 154 scenarios in the competition

and faulty scenarios used abrupt persistent, abrupt intermittent and incipient fault profiles. Each of these fault types are described in detail in section 4.

3. FAULT DETECTION AND ISOLATION ARCHITECTURE

Our observer-based approach for fault detection and isolation is illustrated in Figure 2. Both the physical system being monitored and the observer receive the same input signals. The system output, i.e., the actual system measurements, are labeled as $y[n]$, and the observer estimates are labeled as $\hat{y}[n]$. The system residual vector is computed as $y[n] - \hat{y}[n] = r[n]$ at time step n. The residual vector and the system output are analyzed further in the fault isolation and fault identification units to establish the true fault hypothesis and the magnitude of change in the fault parameter.

The fault detector uses hypothesis testing methods to determine if the computed residual signals imply a fault in the system. The fault detector has to be robust to measurement noise, system disturbances as well as model inaccuracies. Our approach assumes a fault detector monitoring each sensor. Each detector outputs the parameter b, which indicates a faulty or nominal signal. A non-zero value for b implies that the measurement is devaiant from its nomial value. If the signal is faulty, then the detector also outputs the estimated fault type, and the parameters describing the changes in the residual, θ.

The fault isolator aggregates the data from all of the detectors to build a fault signature vector based on the value of b from each of the detectors and the direction of the change from θ. The fault signature is then looked up in a table of possible faults to find the fault candidates which identify the fault and faulty component. The fault isolator needs to deal with uncertainties in the fault signature results, since false alarms and missed detections are unavoidable in any fault detector. The fault isolator then sends the possible fault candidates, and all of the residuals and raw sensor data to the fault identifier.

The fault identifier narrows down the list of fault candidates to the most likely candidate and then provides a final estimate of the fault parameters, θ. It applies any necessary transformations to the residual estimates generated by the fault detector in order to estimate the fault magnitude of the failed component in the system. This transformation may not be required if there is a one-to-one correspondence between the measured variable and the faulty component. After it identifies the fault parameters, the fault identifier sends all of the fault information to the user to be evaluated.

In this work we focus on the design of the fault isolator and fault identification components, and we will group their functionality into one unit and refer to them simply as the isolator. The design of the fault detector is discussed in previous work [Carl et al., 2012]. The isolator is designed to accomplish the following tasks:

(1) Construct, at any time during operation, a system wide fault signature using information provided by the detectors.

(2) Package the fault type and fault parameters in a way useful to the operator.

4. PROBLEM FORMULATION

4.1 Fault Hypothesis

We assume that signals generated by the physical system have added independent and identically distributed Gaussian Noise, represented as $w[n]$, with zero mean and an unknown variance that is unaffected by system faults. The variance can be calculated with knowledge of nominal system behavior and the sensor measurements before a fault occurs in the system.

When a fault occurs in the system starting at time t_{inj}, the system measurements can be defined as:

$$y[n] = \begin{cases} s[n] + w[n] & t < t_{inj} \\ s_{f_i}[n] + w[n] & t \geq t_{inj} \end{cases}, \tag{1}$$

where $s[n]$, the nominal signal value at time step n, is known from available system behavior data, or is estimated using an observer scheme [Basseville and Nikiforov, 1993] and $w[n]$ represents the noise in the measurement that is typically attributed to the sensor. After the fault occurrence, the signal value is linked to faulty system behavior and is expressed as $s_{f_i}[n]$ for $n \geq t_{inj}$. The detection problem can be expressed as:

$$\begin{aligned} \mathcal{H}_0 : &\quad r[n] = w[n] \\ \mathcal{H}_i : &\quad r[n] = \Delta s_{f_i}[n] + w[n] \quad i = 1, 2, \ldots, m \end{aligned} \tag{2}$$

where $\mathcal{H}_0$ is the null hypothesis of no fault, $\mathcal{H}_i$ is the alternative (fault) hypothesis, m is the number of faults, and $\Delta s_{f_i}[n] = s_{f_i}[n] - s[n]$ represents the deviation in the measurement as a result of the fault.

We formulate the detection problem for three different fault profiles. The detector needs to estimate a variety of parameters for each fault type, and the detection problem for each fault is defined by the fault profile and the set of parameters associated with the profile [Carl et al., 2012, Tantawy, 2011]. The isolation problem is to identify the fault type and the faulty system component.

4.2 Fault Profiles

Abrupt Persistent The abrupt persistent fault profile, shown in Figure 3A, is characterized by the nominal signal changing by an unknown positive or negative additive fixed value. An example from the competition is shown in Figure 5. For an abrupt persistent fault the detector needs to estimate the fault time of injection, t_{inj}, and the change in the magnitude of the signal, A, caused by the abrupt fault. The residual signal for the fault is expressed as:

$$r[n] = A + w[n]. \tag{3}$$

Stuck Sensor This is a special case of the abrupt persistent fault where the sensor stops reading new data and only reports the stuck-at value, c. The value of c is independent of the actual measurement, and there is no more white noise in the signal because the sensor stops sampling data. Since the detector looks for a change in the residual signal, it may not be able to detect this kind of fault if the stuck value is within the noise threshold. If

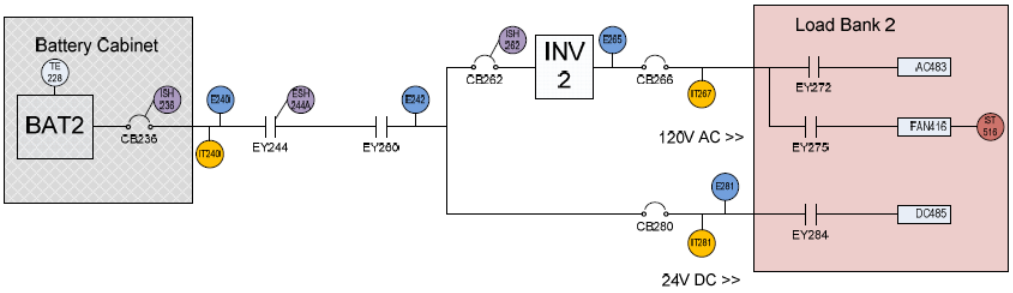

Fig. 1. NASA Electrical Power System Schematic Diagram. The circles represent the sensors in the system with the different colors indicating different kinds of sensors.

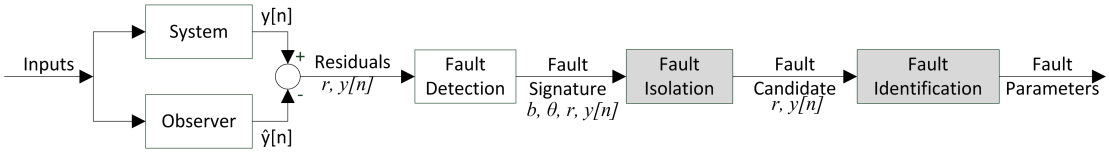

Fig. 2. Fault detection and isolation system block diagram.

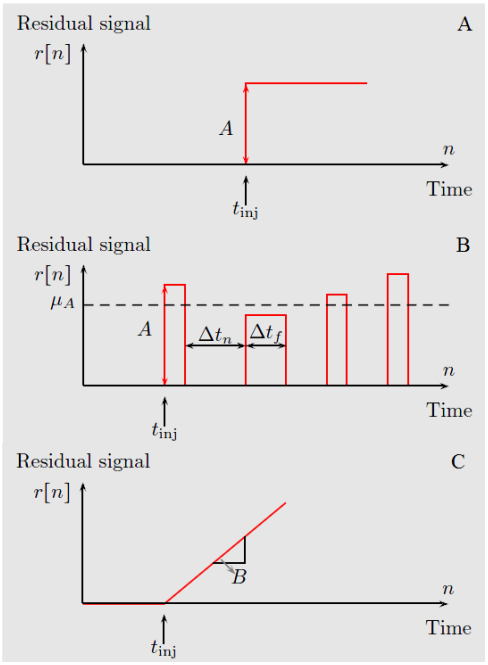

Fig. 3. Idealized fault profiles. Noise is removed from the residual signal for clarity.

the residual signal is outside of the fault threshold, then the detector will detect the fault as an abrupt persistent fault. In either case, the isolator is able to identify the sensor fault because the sensor is at a fixed value, whereas the other measurements continue changing according to the additive Gaussian noise. If the isolator identifies the fault as a stuck sensor fault then the isolator will ignore any other faults reported by the detectors.

Abrupt Intermittent An abrupt intermittent fault profile, shown in Figure 3B, is modeled as a repeated abrupt persistent fault that resets itself after a random time interval. An example from the competition is shown in Figure 6. The fault persistence time, Δt_{fi}, and the inter-arrival time, Δt_{ni}, for each fault repetition are drawn from exponential distributions of $\exp(\mu_f, t_f)$ and $\exp(\mu_n, t_n)$, respectively. The change in residual signal magnitude, A, caused by the fault is drawn from a Gaussian distribution with mean μ_A and variance σ_A^2. The detector needs to estimate the fault time of injection, t_{inj}, the mean residual signal magnitude, μ_A, mean persistence time for the fault, μ_f, and mean inter-arrival time for the fault, μ_n. The residual signal for the fault can be expressed as:

$$r[n] = A[n]Z[n] + w[n], \tag{4}$$

where the function $Z[n]$ is a binary random process representing the presence or absence of the fault, defined by:

$$Z[n] = \begin{cases} 0 & \text{fault absent} \\ 1 & \text{fault present} \end{cases}. \tag{5}$$

Incipient An incipient fault profile, shown in Figure 3C, is a linear change (positive or negative) in the sensor signal. An example from the competition is shown in Figure 7. Incipient faults can be approximated by a linear profile, because they evolve slowly in time. For an incipient fault the detector needs to estimate the time of injection, t_{inj}, and the slope of the signal, M. The residual signal for the fault can be expressed as:

$$r[n] = B * (n - t_{inj}) + w[n] \tag{6}$$

where $B = MT_s$, M is a constant representing the slope of the drift, and T_s is the sampling period.

4.3 Isolation

In our application, a fault signature is a qualitative representation of the zeroth-order derivative changes on the system residual values and on the system signals due to a fault [Mosterman and Biswas, 1999, Daigle et al., 2008]. One can isolate the true single fault in the system by combining information from the different fault detectors into the system fault signature. The two main types of faults in the competition are sensor faults and component faults, but both types of faults follow the given fault profiles. A sensor fault is when a sensor stops reporting accurate information on the nominal system operation.

A component fault is when the component parameters change from their nominal values.

Fault Signature The fault signature is a string of characters where each character represents the output from one of the detectors. We derived the fault signatures for all of the possible faults in the system by analyzing the magnitude of the change (zeroth order time-derivative) each fault will have on the system. The combined list of fault signatures is the fault table. The fault signature uses 'X','+', '-', and '0' to represent no deviation, positive magnitude change, negative magnitude change and signal dropped to zero, respectively. A '0' to represent a signal that has dropped to zero. A '0' is needed to help differentiate failed off faults or switch failed open faults, where a measured signal will go to 0, from a negative magnitude fault, where the signal will drop but not necessarily go to 0.

Sensor Fault A sensor fault only affects the measurements the sensor is reporting, it does not affect the overall system operation. This means that only one detector will report a fault. For sensor faults the detector needs to estimate the time of injection, t_{inj}, and the isolator needs to estimate what sensor failed, the failure mode, and the fault parameters for that failure mode.

For stuck sensor faults the isolator runs a check on all of the recent time steps (configurable per sensor) to determine if the signal is at a constant value. If the signal is constant, then the isolator will report a stuck sensor fault and ignore any other faults reported by the detectors (most likely an abrupt persistent fault). The isolator needs to report the time of injection, t_{inj}, the sensor that failed, and the stuck value, c.

Component Fault Component faults, through propagation, will affect multiple measurements in the system, and therefore require a system level view to make the correct diagnosis. For example, in the case of an electrical power system, an abrupt change in the value of a load implies a change in current through the load and, therefore, the other parts of the system. Each of the detectors will recognize the fault as a change in the measurement value, and the isolator logic uses the type and direction of the change in each measurement to uniquely isolate the fault candidate. For component faults the detector needs to estimate the time of injection, t_{inj}, and the isolator needs to estimate what component has failed, the failure mode, and the fault parameters for that failure mode.

A switch failing open is handled as a special case of a component fault because some measured signals will go to 0, instead of simply increasing or decreasing. In the case of a switch failing open the isolator does not need to report the magnitude of the change as the fault type (Switch Failed Open) inherently includes the magnitude, but it does need to estimate the time of injection, t_{inj}, and the switch that failed.

5. IMPLEMENTATION

The fault isolation architecture is shown in Figure 4. The residual parameters are sent to the fault signature creation and the fault type determination modules. The fault signature creation module builds the fault signature vector based on the information in the individual measurement components b. The fault signature is then used to find the actual fault in the fault table. This allows for a general framework, where (1) detectors can be tuned separately to match the characteristics of the system they are being applied to, and (2) the isolator can be designed and implemented independent of the detector tuning; its qualitative fault signatures are determined offline using a dynamic model of system behavior. This division of labor makes our approach practical for online applications across different kinds of systems. The fault type determination module is needed because each of the detectors can report a different fault type, but there can only be one fault in the system (according to the competition description). When there are mismatches, logic is used to determine the most likely fault type. The fault parameter estimation module determines the actual magnitude of the fault based on the estimated residual parameters from the detectors.

5.1 Fault Signature Construction

The fault signature is built in the isolator by aggregating the information from all of the detectors. Each detector reports an 'X', '+', or '-'. The '0' in the fault signature is derived in the isolator based on if the detector reported a negative fault and if the raw system signal output dropped below a threshold. The '0' cannot be derived in the detector because the detector only receives residual values. The position of each sensor in the fault signature is listed in Table 1. A few example fault signatures from the case study are shown below in Table 2. Refer to Figure 1 for information on the system structure.

An example from the competition will help to illustrate the fault construction in the isolator. Each character in the fault signature represents the information from one detector. One of our fault signatures in Table 2 is XX+-++0XXX. This means that for a DC485 component failed off fault, sensors ISH236, ESH244A, ST516, IT267, and E265 should not report a fault, sensors E240, E242, and E281 should report a positive abrupt fault, sensor IT240 should report a negative fault, and sensor IT281 should report a zero.

Fault Signature Position	Sensor
1	ISH236
2	ESH244A
3	E240
4	IT240
5	E242
6	E281
7	IT281
8	E265
9	IT267
10	ST516

Table 1. A listing of each sensor's position in the fault string.

5.2 Fault Signature Isolation

The isolator uses fault signatures and a pre-defined fault lookup table and robust table lookup logic to match the system behavior to the most likely fault.

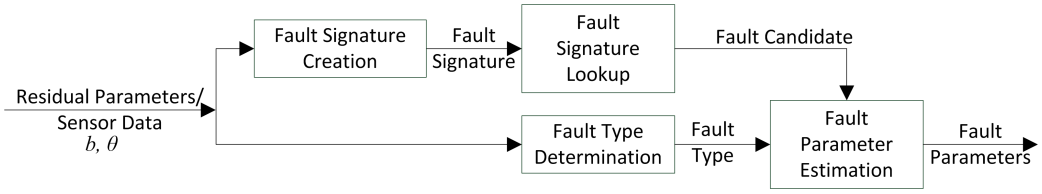

Fig. 4. Fault isolation architecture.

Fault Description	Fault Signature
Relay EY272 Failed Off	XX00000000
Breaker CB280 Failed Open	XX+-+00X+X
Component DC485 Failed Off	XX+-++0XXX
Component AC483 Offset	XX-+—-+X
Sensor E265 Offset	XXXXXXX+XX
Sensor E265 Offset	XXXXXXX-XX

Table 2. Example fault signatures.

The lookup table is built by analyzing the system schematics and determining the effect each fault would have on the system. For this case study we used a pre-defined fault lookup table. If the system is much larger or complicated than the one described in the case study, pre-defining the fault lookup table is going to be difficult, tedious, and error prone. Online fault signature generation and analysis may help alleviate this problem [Mosterman and Biswas, 1999]. Regardless of how the analysis is performed, the lookup table has the potential problem of multiple faults appearing with the same fault signature. If multiple faults have the same signature, then it is impossible to isolate the fault using this lookup method. However, if each fault has a unique signature, then every fault can be isolated. As an example, in our case study it is impossible to tell the difference between load DC485 failing off and switch EY284 failing open, because both faults have the exact same effect on the system and, therefore, their fault signatures are the same.

To combat these problems the lookup table is augmented with robust table lookup logic. It is likely that the generated fault string will not exactly match the lookup string in the fault table, so the lookup logic needs to be robust to pick the most likely fault profile even if the fault signature does not exactly match. This includes two types of logic. The first is a generic heuristic based ranking to sort through all of the signatures and keep only the most likely signatures and to then differentiate between the selected signatures. The most naive form of this heuristic is a string similarity metric. The other type is system specific rules that help eliminate certain signatures. For example, if one detector keys on only one specific system component, then these rules may eliminate signatures that do not correspond to faults based around that component.

5.3 Fault Parameter Estimation

After the isolator finds the fault in the table, it begins to process the fault by calculating the fault parameters and packaging the fault information for reporting. If the fault is a sensor fault, then the isolator simply uses the residual parameters calculated by the responsible detector as the fault parameters. If the fault is a component fault, then the isolator needs to calculate the fault magnitude based on what the detector reports using simple circuit analysis. For example, there is no sensor directly sensing the resistance of the load DC485. However, if the fault is an abrupt resistance increase in DC485, that will be detected in the nearby voltage and current sensors E281 and IT81, respectively. The detectors responsible for those two sensors will detect a fault, but the magnitude of the resistance change will need to be calculated based on the sensed magnitudes of the E281 and IT281 detectors. Once the fault is packaged appropriately, it is reported.

6. EXPERIMENTAL RESULTS

There were 154 scenarios in the competition, and examples from the competition data of each primary fault type are shown in Figures 5, 6, and 7. The detector and isolator's performance is shown in 3. The first three rows in the table outline the detector and isolator performance for faults that fit neatly into one of the three fault profiles outlined in section 4. The final row of the table presents the results for all scenarios, including nominal scenarios and scenarios where the fault did not fit into one of the three categories (i.e., switch failed open, or fan overspeed). Our detector and isolator were able to correctly recognize and isolate 95 of the scenarios in the time allocated for detection. There were 24 cases, especially for incipient faults, where the detectors were not able to robustly determine the correct fault type before the data series ended, but the isolator was still able to report the correct component or sensor from the initial detector results. With the addition of the extra 24 cases the isolator performed correctly in 119 scenarios. This indicates that the isolator was able to build a system wide fault signature, determine the actual faulty component, and package the fault information in a way useful to the operator even though the supplied fault information was not completely correct. Where the failure happened in the remaining 35 scenarios is difficult to analyze because of how tightly the isolator performance is linked to the detector performance. It is very likely that the isolator reported an incorrect component because the detectors reported a fault signature that the isolator did not recognize, and therefore the isolator had to "guess" the faulty component.

The isolator performance varied somewhat depending on if the fault was a sensor fault or a component fault. Out of 70 sensor faults the isolator reported the correct sensor in 51 of the scenarios (73%) and out of 64 component faults the isolator reported the correct component in 52 scenarios (81%). This indicates that for component faults the extra data from the detectors was instrumental in finding the correct component, even though some of that data may be incorrect. The sensor faults only required one piece of data for isolation, which indicates that there is no chance to deduce the correct sensor if the one piece of supplied data is incorrect or is masked by false positives.

Fault Type	Total Scenarios	Detected and Isolated	Only Component Isolated
Abrupt Persistent	37	29	33
Abrupt Intermittent	35	25	28
Incipient	37	19	32
All Scenarios	154	95	119

Table 3. Detector and Isolator performance, NASA DXC'10 competition.

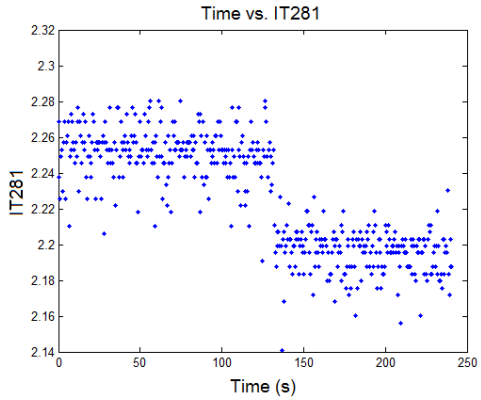

Fig. 5. IT281 Abrupt fault. Fault magnitude: -0.05A.

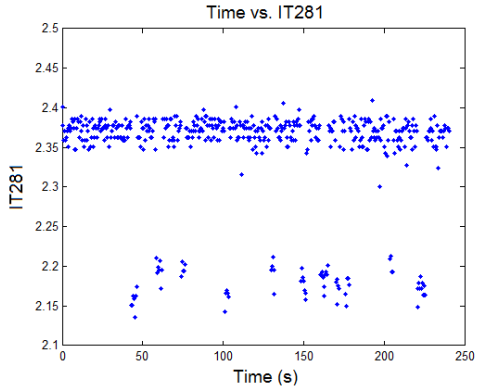

Fig. 6. IT281 Abrupt intermittent fault. Average fault magnitude: -0.19A, average duration time: 3.21 seconds, and average inter-arrival time: 16.14 seconds.

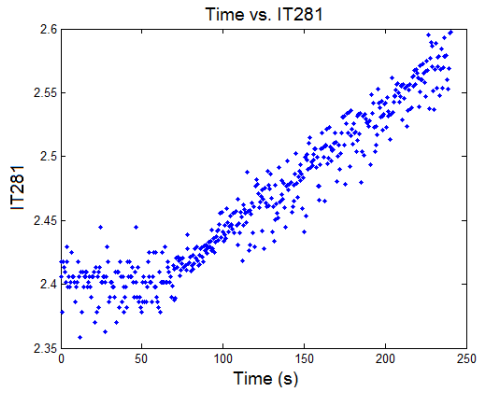

Fig. 7. IT281 Incipient fault. Slope: 0.001.

7. CONCLUSIONS

We used a pre-defined fault lookup table to determine the fault based off of information from the detectors. This worked fine for the simple system in our case study, and provided better results for more complicated fault signatures than for simple signatures. A table lookup scheme also has the advantage of being very fast, and will not likely be a performance constraining part of the system. However, for more complicated systems predefining the fault table can be tedious and error-prone. In that situation an on-line method of deriving the data in the fault table is recommended. The isolator needed robust lookup logic to account for incorrect information from the fault detectors, and even with the robust lookup logic the isolator still occasionally isolated the fault incorrectly. Increasing the reliability of the fault detectors is the best way to increase the reliability of the fault isolator.

ACKNOWLEDGEMENTS

We would like to thank NASA Ames for sponsoring the 2010 Diagnostic Competition. This work was partially supported by a NASA grant NRA NNX07AD12A and by DARPA META contract FA8650-10-C-7082.

REFERENCES

M. Basseville and I. V. Nikiforov. *Detection of Abrupt Changes: Theory and Applications*. Prentice Hall, Inc., 1993.

J. D. Carl, A. Tantawy, G. Biswas, and X. D. Koutsoukos. Detection and estimation of multiple fault profiles using generalized likelihood ratio tests: A case study. In *16th IFAC Sysid*, 2012.

M. Daigle, X. Koutsoukos, and G. Biswas. An integrated approach to parametric and discrete fault diagnosis in hybrid systems. Technical report, Institute for Software Integrated Systems at Vanderbilt University, 2008.

T. Kurtoglu, S. Narasimhan, S. Poll, D. Garcia, L. Kuhn, J. de Kleer, and A. Feldman. The Diagnostic Challenge Competition - DCC'09. Technical report, NASA Ames Research Center, Palo Alto Research Center, Delft University of Technology, 2008. `https://c3.nasa.gov/dashlink/projects/36/`.

T. Kurtoglu, S. Narasimhan, S. Poll, D. Garcia, L. Kuhn, J. de Kleer, and A. Feldman. Second International Diagnostic Competition (DXC'10), Industrial Track, Diagnostic Problem Descriptions. Technical report, NASA Ames Research Center, 2010. `https://www.phmsociety.org/competition/dxc/10`.

P. Mosterman and G. Biswas. Diagnosis of continuous valued systems in transient operating regions. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 29(6):554 –565, Nov 1999.

A. Tantawy. *Model-based Detection in Cyber-Physical Systems*. PhD thesis, Vanderbilt University, Nashville,TN USA, October 2011.