

A ROBUST METHOD FOR HYBRID DIAGNOSIS OF COMPLEX SYSTEMS

Gautam Biswas¹, Gyula Simon¹, Nagabhushan Mahadevan¹, Sriram Narasimhan²,
John Ramirez¹, and Gabor Karsai¹

¹Institute for Software Integrated Systems (ISIS)
Dept. of EECS
Vanderbilt University
Nashville, TN 37235.

²QSS Group Inc.
NASA Ames Research Center
Moffett Field, CA 94035.

[gautam.biswas, gyula.simon, nag.mahadevan, john.ramirez, gabor.karsai@vanderbilt.edu;](mailto:gautam.biswas,gyula.simon,nag.mahadevan,john.ramirez,gabor.karsai@vanderbilt.edu)
[sriram@mail.arc.nasa.gov;](mailto:sriram@mail.arc.nasa.gov)

ABSTRACT

The AI model-based diagnosis community has developed qualitative reasoning mechanisms for fault isolation in dynamic systems. Their emphasis has been on the fault isolation algorithms, and little attention has been paid to robust online detection and symbol generation that are essential components of a complete diagnostic solution. This paper discusses a robust diagnosis methodology for hybrid systems that combines fault detection with a combined qualitative and quantitative fault isolation scheme. We focus on fault detection, symbol generation, and parameter estimation, and illustrate the effectiveness of this method by running experiments on the fuel transfer system of aircraft. ©: 2003 *IFAC*

Keywords: Fault detection, Fault isolation, Qualitative analysis, Parameter estimation, Robust performance.

1. INTRODUCTION

This paper addresses the problem of designing and implementing online monitoring and diagnosis algorithms for complex systems whose behavior is hybrid (discrete + continuous). Hybrid models capture the behavior of embedded systems that are common in the avionics, automotive, and robotics domains. This work deals with a special class of embedded hybrid systems characterized by continuous plant dynamics and a discrete supervisory controller. The behavior of the plant evolves in continuous time governed by the physical parameters of plant components and their interconnections. The controller generates actuator signals at discrete time points that can change (i) the operational modes of the plant by turning components ON and OFF, (ii) component parameter values, and (iii) the set points of regulators. These operating mode changes produce discrete changes in the dynamic models of the system behavior, and behavior

analyses require multiple system models. As a result, tasks like monitoring, fault diagnosis, and control require appropriate model selection and switching to be performed online in real time.

Current techniques in model-based diagnosis apply well to dynamic systems whose behavior is modeled by discrete event [Lunze and Schroder, '02; Sampath *et al.*, '96], or continuous models [Gertler, '97; Mosterman and Biswas, '99]. For hybrid diagnosis, the discrete-event approach defines abstractions of system behavior (both nominal and faulty) that map to discrete event representations. The resultant information loss may be critical for tasks like fault isolation and control. [Manders and Biswas, '01] have demonstrated that behavior transients are the key to quick diagnosis of abrupt faults in continuous systems. Discrete event models also require pre-enumeration of all faulty and non faulty behavior trajectories, which may be computationally infeasible. Traditional algorithms

for continuous diagnosis are based on a single model with no provision for discrete changes. Therefore, discrete effects of mode changes have to be modeled by complex continuous non-linear functional relations that are hard to analyze online in real time.

Recent work on diagnosis of hybrid systems [Dearden and Clancy, '02; Hofbaur and Williams, '02; Zhao *et al.*, '01] has focused on discrete faults, and required the enumeration of the model in all modes to perform diagnostic analysis. This work discusses a model-based diagnosis methodology for parametric faults in hybrid systems that do not require explicit pre-enumeration of models in all modes of system operation. Our online hybrid diagnosis scheme uses a novel approach that combines fast qualitative reasoning techniques with parameter estimation methods to achieve more refined and accurate diagnoses [Narasimhan and Biswas, '02]. The qualitative approach overcomes limitations of quantitative schemes, such as convergence and accuracy problems in dealing with complex non-linearities and lack of precision of parameter values in system models. It significantly cuts down the computational complexity to facilitate online processing. The qualitative reasoning scheme is fast, but it has limited discriminatory ability. To uniquely identify the true fault candidate, we employ a quantitative parameter estimation scheme, which also returns the magnitude of the deviated parameter. The paper focuses on fault detection, symbol generation, and parameter estimation algorithms that work in conjunction with the qualitative fault isolation scheme. To deal with realistic situations, the algorithms are designed to be robust to modeling errors and measurement noise.

2. TRACKING HYBRID BEHAVIOR

Our diagnosis architecture implements a scheme to track the nominal system dynamics using a robust observer scheme implemented as a combination of an extended Kalman filter (EKF) and a hybrid automaton. A fault detector triggers the fault isolation scheme, which first generates an initial candidate set, and refines it by tracking and analyzing the fault transients using fault signatures. The hybrid nature of the system complicates these tasks, because mode transitions cause model switching, which has to be included in the online behavior tracking and fault isolation algorithms.

The hybrid observer has to track (i) continuous behavior in individual modes of operation, and (ii) discrete mode changes (controlled and autonomous). At mode changes, the new state space model and the initial state of the system are recomputed. The hybrid observer scheme is designed as an extension of the continuous extended Kalman filter. Model uncertainty and measurement noise are implemented as white, uncorrelated Gaussian distributions with zero mean. The state space model in mode q is defined as:

$$\mathbf{x}_{k+1} = \mathbf{F}_q(\mathbf{x}_k)\mathbf{x}_k + \mathbf{G}_q(\mathbf{x}_k)\mathbf{u}_{k+1} + \mathbf{w}_k$$

$$\mathbf{y}_{k+1} = \mathbf{C}_q(\mathbf{x}_k)\mathbf{x}_{k+1} + \mathbf{D}_q(\mathbf{x}_k)\mathbf{u}_{k+1} + \mathbf{v}_{k+1}$$

where \mathbf{w} is distributed $N(0, Q)$ and \mathbf{v} is distributed $N(0, R)$, and Q and R are process and measurement noise covariance matrices. It is assumed that \mathbf{w}_k incorporates the $\Delta F_{q,x_k}$ term that captures modeling errors in the system. In our work, the Q and R matrices were determined empirically. The extended Kalman filter algorithm follows the methodology presented in [Gelb '96].

Mode change calculations are based on the system mode at time step k , q_k , and the continuous state of the system, \mathbf{x}_k . The discrete controller signals to the plant are assumed known. For controlled transitions, we assume such a signal is input at time step k , and the appropriate mode transition is made at time step $k+1$ to q_{k+1} . For autonomous transitions, the estimated state vector, \mathbf{x}_k is used to compute the Boolean functions that signal mode transitions. A mode transition results in a new state equation model, i.e., the matrices F_q , G_q , C_q , and D_q are recalculated. To simplify analysis, we assume that mode changes and faults occur only after the Kalman filter state estimate has converged to its optimal behavior. Further details of the observer implementation are presented in [Narasimhan '02].

3. FAULT DETECTION AND SYMBOL GENERATION

The fault detector continually monitors the measurement residual, $\mathbf{r}(k) = \mathbf{y}(k) - \hat{\mathbf{y}}(k)$, where \mathbf{y} is the measured value, and $\hat{\mathbf{y}}$ is the expected system output, determined by the hybrid observer. Ideally, any non-zero residual value implies a fault, which should trigger the fault isolation scheme. In most real systems, the measured values are corrupted by noise (Gaussian with zero mean and unknown but constant variance), and the system model (thus the prediction system) is not perfect. Therefore, statistical techniques are required for reliable fault detection.

3.1 Fault detection scheme

We start by defining a signal deviation at time step k in terms of an average residual for the last N_2 samples, i.e.,

$$\hat{\mu}_{N_2}(k) = \frac{1}{N_2} \sum_{i=k-N_2+1}^k r(i)$$

A hypothesis testing scheme based on the Z-test is employed to establish the significance of the deviation. To perform the Z-test, the variance of the measurement residual must be known. (For unknown variance the T-test may be performed, but its confidence interval is much larger.) To approximate the conditions necessary for the Z-test, the variance of the signal is estimated, but from a larger data set containing N_1 samples, i.e., $N_1 \gg N_2$:

$$\hat{\sigma}_{N_1}^2(k) = \frac{1}{N_1 - 1} \sum_{i=k-N_1+1}^k (r(i) - \mu_{N_1}(k))^2.$$

The Z-value has distribution $N(0,1)$:

$$Z = \frac{\hat{\mu}}{\sigma / \sqrt{N_2}} \quad (1)$$

The confidence level, defined by α , defines the bound $[z_-, z_+]$:

$$P(z_- < z < z_+) = 1 - \alpha. \quad (2)$$

This bound can be transformed to another bound $[\mu_-, \mu_+]$ using equation (1), and the approximation $\sigma \cong \hat{\sigma}_{N_1}$:

$$\begin{aligned} \mu_- &= z_- \hat{\sigma}_{N_1} / \sqrt{N_2} \\ \mu_+ &= z_+ \hat{\sigma}_{N_1} / \sqrt{N_2} \end{aligned}$$

The Z-test is employed in the following manner:

$$\begin{aligned} \mu_- < \hat{\mu}_{N_2} < \mu_+ &\Rightarrow \text{No Fault} \\ \text{otherwise} &\Rightarrow \text{Fault} \end{aligned}$$

The proposed fault-detection scheme is sub-optimal compared to the well-known CUSUM algorithm [Basseville and Nikiforov '93]. However, its advantage is that it makes no assumptions concerning the properties of the changed mean value (it does not have to be constant), and it is computationally simpler.

3.2 Symbol Generation

The transients in the deviant measurements are tracked over time and compared to predicted fault signatures to establish the fault candidates. A fault signature is defined in terms of magnitude and higher order derivative changes in a signal [Mosterman and Biswas, '99]. However, to achieve robust and reliable analysis with noisy measurements, we assume that only the signal magnitude and its slope can be reliably measured at any time point. Since the fault signatures are qualitative, the symbol generation scheme is required to return (i) the *magnitude* of the residual, i.e., $0 \Rightarrow$ at nominal value, $+ \Rightarrow$ above nominal value, and $- \Rightarrow$ below nominal value, and (ii) the slope of the residual, which takes on values, $\pm \Rightarrow$ increasing or decreasing, respectively. Also, measuring only magnitude changes and slopes of residuals implies that the direction of the discontinuity plus the slope of the signal provides the discriminatory evidence needed for fault isolation. Otherwise, only the first change in the signal provides the discriminatory evidence for fault isolation [Manders *et al.*, '00].

The magnitude of the residual is computed as the sign of $\hat{\mu}$. When a discontinuity is detected, the slope of the residual after the discontinuity is computed by making the assumption that the time point of fault detection is k_0 . The approximate variance of the residual at this point is $\hat{\sigma}_r^2 = \hat{\sigma}_{N_1}^2(k_0 - N_2)$. It is assumed that the noise variance of the signal does not change due to the fault. A delayed value is used to prevent distortion of the variance estimate.

Like in the case of symbol generation for the residual magnitude, a statistical test on the mean value is used to make the decision on the value of the slope. The size of the window used to calculate the mean is increased until the symbol is successfully generated. The estimated 'mean slope' of the signal after fault detection is defined as:

$$\mu_d(k_0 + k) = \begin{cases} \frac{1}{k} \sum_{j=1}^k (\mu_{r_0}(N_3) - r(k_0 + j)) = \mu_r(N_3) - \frac{1}{k} \sum_{j=1}^k r(k_0 + j), & k > N_3 \\ 0 & k \leq N_3 \end{cases} \quad (2)$$

where $\mu_{r_0}(N_3)$ is an estimate of the 'initial' residual value after the fault detection, using N_3 samples:

$$\mu_{r_0}(N_3) = \frac{1}{N_3} \sum_{i=0}^{N_3-1} r(k_0 + i).$$

The variance of μ_d is $\sigma_d^2(k_0 + k) \approx \frac{\sigma_r^2}{k}$, while the variance of μ_{r_0} is $\sigma_r^2 \approx \frac{\sigma_r^2}{N_3}$. The uncertainty of the

initial residual value depends on the noise and N_3 , while the uncertainty of the mean estimate depends on the noise and the number of samples used in the calculation. Using a confidence value α and the corresponding z_+ value defined in equation (2), the condition of for a + slope symbol is given by:

$$\mu_d - z_+ \sigma_d > z_+ \frac{\sigma_r}{\sqrt{N_3}}$$

The condition for the negative symbol can similarly be derived. The rules for generation of the slope symbol can be summarized as follows.

$$\mu_d > z_+ \sigma_r \left(\frac{1}{\sqrt{N_3}} + \frac{1}{\sqrt{k}} \right) \Rightarrow \text{slope symbol} = +$$

$$\mu_d < -z_+ \sigma_r \left(\frac{1}{\sqrt{N_3}} + \frac{1}{\sqrt{k}} \right) \Rightarrow \text{slope symbol} = -$$

The method is illustrated in Figure 1. The first plot shows the noisy residual, while the other plots show the slope estimate μ_d with the corresponding confidence bounds, and also the confidence bound of the initial residual estimate μ_{r_0} . As the figure illustrates, the choice of N_3 is not straightforward. A small value results in a large threshold and a large value may cause significant delay. Another disadvantage of a large value is that it may suppress short transients in the residual. The best values for N_3 were between 5 and 20.

4. FAULT ISOLATION AND IDENTIFICATION

Once a fault has been detected, fault isolation and identification is performed to uniquely isolate the fault and determine its magnitude. Our fault isolation and identification architecture, presented in Figure 2 involves three steps: (i) *qualitative roll-back*, (ii) *qualitative roll-forward*, and (iii) *quantitative parameter estimation*.

For hybrid systems, discontinuous changes in measured variables can only occur at the point of failure or when discrete mode changes occur in the plant behavior. At all other time points the plant behavior is continuously differentiable. We take advantage of this fact for qualitative analyses of all measured variables, y_k . The deterministic form of the corresponding residual, r_k is continuously differentiable after the fault occurrence, and after each mode change, so it can be approximated by the Taylor series expansion:

$$r_t = r_{T_{fo}} + r'_{T_{fo}} \frac{(t - T_{fo})}{1!} + r''_{T_{fo}} \frac{(t - T_{fo})^2}{2!} + \dots + r^{(k)}_{T_{fo}} \frac{(t - T_{fo})^k}{k!} + R_k$$

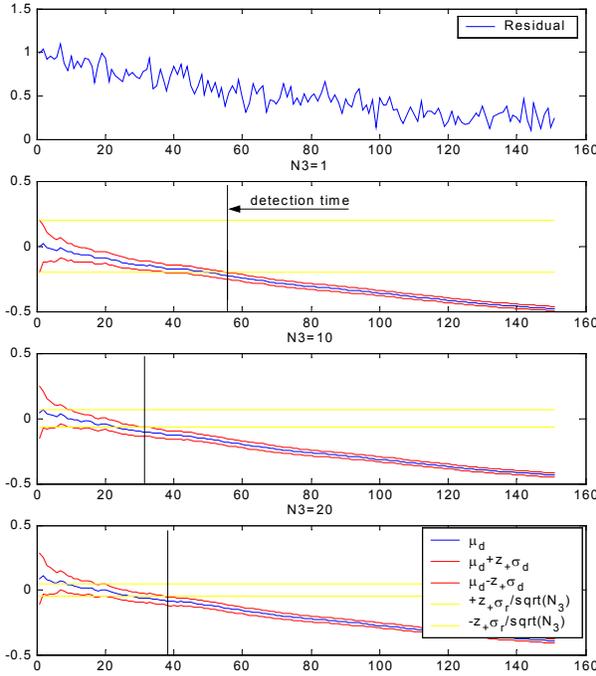


Figure 1: Slope Symbol Generation with different N_3 Values

We use this formulation to define the fault signature corresponding to a residual as the qualitative value of the magnitude and higher order derivative terms of the Taylor series. As discussed above, the qualitative values used are: $(-, 0, \text{ and } +)$.

The qualitative roll-back algorithm can be summarized as follows. Given the observer estimated mode trajectory $Q = \{q_1, q_2, \dots, q_k\}$, we first use the back propagation algorithm [Mosterman and Biswas, '99] to generate hypotheses in mode q_k . The deviated symbols at the time of fault detection (α) are back propagated through the temporal causal graph in mode q_k to identify causes for the deviations. Since the fault may have occurred in previous modes, we then go back in the mode trajectory and create hypotheses in each of the previous modes $q_{k-1}, q_{k-2}, \dots, q_{k-n+1}$, where n is a number determined externally by diagnosability studies. During the crossover from a mode to a previous mode, the symbols are propagated back across the mode change using the inverse of the reset functions (γ^{-1}) associated with the mode transition. The hybrid hypotheses generation algorithm returns a hypotheses set, $H = \{h_1, h_2, \dots, h_m\}$, where each

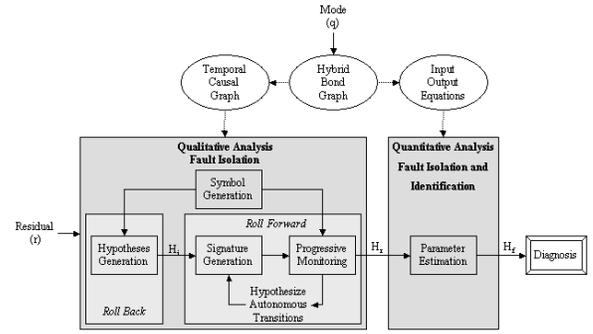


Figure 2: Fault Isolation and Identification Architecture

hypotheses h_i is a three-tuple $\{q, p, \lambda\}$, and q represents the mode in which the fault is hypothesized to have occurred, p is the parameter whose deviation corresponds to the fault, λ is the direction of deviation of parameter p .

The next step is to generate fault signatures for each hypothesis in the current mode, and match them against the observed behavior. The occurrence of the fault may change the parameters of the functions that determine autonomous transitions leading the observer to incorrectly predict (or not predict) an autonomous transition. Hence the current mode of the system has to be estimated for each hypothesis. But this cannot be done till the faulty parameter value is estimated. To overcome this problem, we apply all observed controlled transitions, and calculate the fault signatures in the new mode. When fault signatures do not agree with the observations, autonomous mode transitions are hypothesized, new fault signatures computed, and the matching process is continued. This process, again limited to n steps (diagnosability limit) is the roll-forward process [Narasimhan and Biswas '02]. Further mismatches in signatures and symbols eliminate hypotheses.

[Manders *et al.*, '00] have shown the limited discriminatory capabilities of the qualitative progressive monitoring scheme leads to multiple fault hypotheses being reported as the diagnostic result. We use a parameter estimation technique for further fault isolation and identification. Even when isolation is reduced to a single candidate, it is important to estimate the faulty parameter value. Due to the hybrid, possibly non-linear nature of the system traditional parameter estimation techniques cannot easily be applied. A novel mixed simulation-and-search algorithm is applied to estimate physical parameter deviations in the system model. For multiple fault hypotheses, multiple optimizations are run simultaneously, and each one estimates one scalar degradation parameter value.

The parameter estimation scheme is initiated at the time point of fault detection, T_{fault} . The current state variable values and a set of N measurement samples that includes the system input and output signals are used (currently N is pre-defined). The estimation scheme is based on an optimization algorithm (technically any non linear optimization algorithm may be employed), and the goal is to find the fault parameter

value that minimizes the least square error between the expected system output generated by the simulator and the available measurement values over the N samples. A greedy search algorithm is applied to

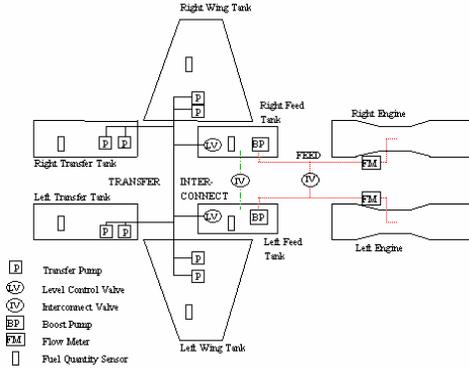


Figure 3: Fuel System Schematic

minimize the error using an error surface that is parameterized by the fault parameter, p . The simulator uses the hybrid automata model of the system to generate system behavior, \hat{Y} , from an arbitrary initial state (currently from T_{faults} , with $X(T_{fault})$ as initial state) using the state space model of the system. The simulator is parameterized, thus the fault parameter can be modified for different simulation runs.

Theoretically the minimum of the error surface $\varepsilon^2(p)$ can be determined by scanning the possible parameter range and determining the minimum value of ε^2 . The calculation of each point $\varepsilon^2(p)$ of the error surface involves a run of the simulator with parameter p . Since each run is computationally expensive, the number of simulation runs must be kept as low as possible. A practically feasible solution is to use an iterative scheme that calculates the error values for a small number of p values by making the assumption that error surface is almost parabolic. The optimization in this case is performed by a series of parabolic fits, with a relatively small number of simulator runs. This scheme is run for every fault hypothesis, and the one that returns the minimum least square error is defined to be the true fault. This scheme has been successfully applied to isolating and identifying the true fault in a number of experiments that we have conducted.

5. EXPERIMENTAL RESULTS

Figure 3 shows the fuel system schematic of fighter aircraft that we have used as our diagnostic test bed. The fuel system is designed to provide an uninterrupted fuel supply at a constant rate to the aircraft engines, and at the same time to maintain the centre of gravity of the aircraft.

The system is symmetrically divided into the left and right parts (top and bottom in the schematic). The four supply tanks (Left Wing (LWT), Right Wing (RWT), The Left Transfer (LTT), and Right Transfer (RTT)) are full initially. During engine operation, fuel is transferred from the supply tanks to the receiv-

Table 1: Fuel System Experiments

Faults		Performance Parameters							
Fault Type	Fault Magnitude	Fault Detection Time		Fault Isolation Time		Initial/Final Candidate Set		Parameter Estimation Error	
	Noise level	2%	3%	2%	3%	2%	3%	2%	3%
LTT-Pump	67%	422	555	197	127	14/3	13/4	5.43	1.79
	40%	182	183	144	240	13/4	13/4	1.28	1.49
	20%	134	134	225	398	13/5	13/4	0.88	2.19
RWT-Pump	67%	0	1	55	106	13/4	13/4	21.50	16.11
	40%	83	3	183	139	13/3	13/4	1.67	1.52
	20%	117	285	211	170	13/4	13/4	0.68	0.68
RLCV - Block (valve)	x 100	51	52	97	79	23/1	23/2	0.62	6.e-5
	x 75	63	63	103	46	25/2	25/1	2.49	4.61
	x 50	63	51	58	86	25/2	25/2	2.27	5.42
Leg21 -Pipe (Block)	x 100	93	93	76	202	14/3	14/2	0.19	0.34
	x 70	95	95	90	303	14/3	14/2	1.65	1.57
	x 40	99	100	136	350	14/2	14/2	0.78	1.58

ing tanks (Left Feed (LFT) and Right Feed (RFT)) in a pre-defined sequence. The pump is modeled as a source of effort (pressure) with a transformation factor that defines its efficiency, and the tanks are modeled as capacitances. The pipes are modeled as nonlinear resistances.

The diagnosis experiments used a controller se-

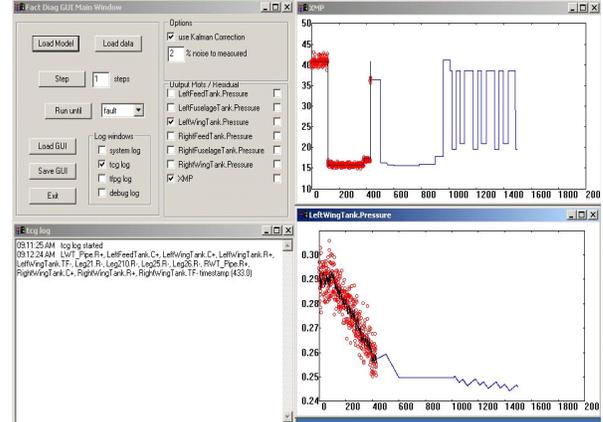


Figure 4: Transfer Manifold and Right Wing Tank Pressure at Fault Detection

quence provided by the manufacturer. Table 2 shows the parameters that were tuned to achieve a desired diagnostic performance. These parameter values were determined empirically. Their values depend on the nature of the system, the set of faults that we wish to isolate, and the trade-off of time to detection and isolation versus accuracy. It is clear that the results of the fault isolation scheme are very dependent on the choice of parameters for the Kalman filter and the fault detector. This issue is often ignored in diagnosis studies.

The results of diagnosis experiments for a set of faults appear in Table 1. The parameters varied for the experimental runs were the percentage of noise in the measurement, and the fault magnitudes (see [Narasimhan, '02] for details).

In what follows, we give demonstrate the details of a fault run, where the system's left wing tank performance degraded to 66% of its original value at

time step = 150. In this and other experiments, the pressures at the output of the six tanks plus the pressure at the transfer manifold were the measured values. Mode changes and other factors, such as magnitude of the fault led to late fault detection at time step = 433. This is illustrated in Figure 4. The TCG log appears on the left bottom of the screen. In the Transfer Manifold Pressure plot (top right), the deviation of the observer estimation (black line) from the sensor measurement (red dots) can be observed. The TCG log shows the initial list of probable fault candidates. At the very next time step, the fault isolation system uses the discontinuity in the transfer manifold pressure to reduce the number of candidates to 10, i.e., the candidates that are inconsistent with the discontinuity are dropped.

At time step = 469, the fault detector connected to the Left Wing Tank Pressure fires, and this reduces the probable fault candidate list to 6. Finally, when a right wing tank pressure deviation is detected, the candidate set is pruned down to four: three pipe resistance increases, and the reduction in pump performance. At this point, no more reduction is possible in the candidate set. The qualitative analysis (TCG) is stopped and the quantitative analysis (Parameter Estimation) starts, and the candidate LeftWingTank.TF (Pump parameter) return the least error after the estimation. Its value is set to the estimated value (0.66) and the observer is found to continue tracking the faulty system well.

6. SUMMARY

This paper has discussed an integrated approach to solving the tracking, fault detection, isolation, and identification tasks for hybrid systems. A key issue that we have demonstrated for the model-based diagnosis community is the use of qualitative reasoning techniques for robust diagnosis in real situations. This required the use of statistical techniques for fault detection and symbol generation. Our work is motivated by the requirements of

Table 2: FDI System Design Parameters

Component	Parameter Name	Parameter Value
Kalman Filter	Sensor Accuracy	0.05
	Modeling Error	0.001
Fault Detector	Fault Detection Threshold	2-3
	Window Size (Variance Estimation)	50
	Window Size (Mean Estimation)	5
Fault Isolation: Symbol Generation	Slope Detection Sensitivity	1
	Window Size	11
Fault Identification: Parameter Estimation	Number of Samples	<=400
	Error Function	Parabolic

the fault accommodation task, where diagnosis has to be performed online for embedded systems during their operation. Hybrid diagnosis techniques directly apply to embedded systems and else where [Narasimhan, '02] we have also demonstrated

through time and space complexity analysis that our algorithms can be applied to online analysis in resource constrained environments.

Acknowledgments

The DARPA/IXO SEC program (F30602-96-2-0227), NASA IS NCC 2-1238, and the Boeing Company (Kirby Keller and Tim Bowman) have supported the activities described in this paper. We would also like to thank Tivadar Szemethy and Eric Manders for their help.

References

- Basseville, M. and I.V. Nikiforov, *Detection of Abrupt Changes – Theory and Applications*, 1993, Prentice Hall.
- Chen, J. and R.J. Patton, *Robust Model-based Fault Diagnosis for Dynamic Systems*. 1998: Kluwer Academic Publishers.
- Dearden, R. and D. Clancy. *Particle Filters for Real-Time Fault Detection in Planetary Rovers*. in *Thirteenth International Workshop on Principles of Diagnosis (DX '02)*. 2002. Semmering, Austria.
- Gelb, A. *Applied Optimal Estimation*, 1996. MIT Press.
- Gertler, J., *Fault Detection and Isolation using Parity Relations*. Control Engineering Practice, 1997. **5**(5): p. 653-661.
- Hofbaur, M.W. and B.C. Williams. *Mode Estimation of Probabilistic Hybrid Systems*. in *Fifth International Workshop on Hybrid Systems: Computation and Control (HSCC '02)*. 2002. Stanford, CA, USA: Springer.
- Lunze, J. and J. Schroder, *Sensor and Actuator Fault Diagnosis of Systems with Discrete Inputs and Outputs*. Submitted to IEEE Transactions on Systems, Man, and Cybernetics, 2002.
- Manders, E. and G. Biswas. *Transient Detection and Analysis for Diagnosis of Abrupt Faults in Continuous Dynamic Systems*. in *International Workshop on Intelligent Signal Processing (WISP '01)*. 2001. Budapest, Hungary.
- Manders, E., et al. *A combined Qualitative/Quantitative approach to efficient Fault Isolation in complex dynamic systems*. in *Symposium on Fault Detection, Supervision, and Safety Processes (SafeProcess '00)*. 2000. Budapest, Hungary.
- Mosterman, P.J. and G. Biswas, *Diagnosis of Continuous Valued Systems in Transient Operating Regions*. IEEE Transactions on Systems, Man, and Cybernetics, 1999. **29**: p. 554-565.
- Narasimhan, S., *Model-based Diagnosis of Hybrid Systems*, in *Computer Science*. 2002, Vanderbilt University: Nashville.
- Narasimhan, S. and G. Biswas, "An Approach to Model-Based Diagnosis of Hybrid Systems," *Hybrid Systems: Computation and Control, Lecture Notes in Computer Science*, vol. LNCS 2289, C.J. Tomlin and M.R. Greenstreet, eds., Springer Verlag, pp. 308-322, 2002.
- Samphath, M., et al., *Failure Diagnosis using Discrete-Event Models*. IEEE Transactions on Control Systems Technology, 1996. **4**: p. 105-124.
- Zhao, F., et al. *Distributed Monitoring of Hybrid Systems: A Model-Directed Approach*. in *International Joint Conference on Artificial Intelligence*. 2001. Seattle, WA.