Model-based Diagnosis of Hybrid Systems

Sriram Narasimhan
QSS Group
NASA Ames Research Center
Moffett Field, CA 94035
sriramfajarc.nasa.gov

Gautam Biswas
Dept. of EECS & ISIS
Vanderbilt University
Nashville, TN 37235
gautam.biswas@vanderbilt.edu

Abstract

Recent years have seen a proliferation of embedded systems that combine a digital (discrete) supervisory controller with an analog (continuous) plant. Diagnosing faults in such hybrid systems, require techniques that are different from those used for discrete and continuous systems. In addition, these algorithms have to be deployed online to meet the real time requirements of embedded systems. This paper presents a methodology for online tracking and diagnosis of hybrid systems. We demonstrate the effectiveness of the approach with experiments conducted on the fuel transfer system of fighter aircraft.

1 Introduction

This paper addresses the problem of designing and implementing online monitoring and diagnosis systems for complex systems whose behavior is hybrid (discrete + continuous) in nature. Hybrid modeling covers naturally occurring systems, such as cell-cycle control systems in biology. They also capture the behavior of embedded systems that are common in the avionics, automotive, and robotics domains. This wide applicability of hybrid systems has inspired a great deal of research from both control theory and theoretical computer science.

We focus on a special class of embedded hybrid systems, characterized by continuous plant dynamics and a discrete supervisory controller. The plant dynamics are defined by continuous state variables associated with the components of the plant. The controller generates actuator signals at discrete time points that can change the plant configuration by turning components ON and OFF, and changing component parameter values and the set points of regulators. Therefore, hybrid system models have to seamlessly integrate discrete and continuous behavior analyses using multiple system models. As a result, tasks like monitoring, fault diagnosis, and control require appropriate model selection and switching to be performed online as system behavior evolves.

This paper discusses methodologies for the model-based diagnosis (MBD) of hybrid systems. Current techniques in MBD apply well to dynamic systems whose behavior is modeled with discrete event [9, 17], or continuous models [5, 14]. Discrete event approaches to hybrid system diagnosis are based on abstractions of nominal and faulty behavior system behavior into event trajectories. This process may result in loss of information critical for fault isolation and control. Our work in continuous diagnosis has demonstrated that behavior transients are the key to quick diagnosis of abrupt faults[10]. It may also be computationally expensive to pre-enumerate all possible nominal and faulty behavior trajectories. Traditional algorithms for continuous diagnosis use a single model that does not accommodate discrete changes. Therefore, discrete effects of mode changes have to be modeled by complex continuous non-linear functions that are hard to analyze online in real time.

Recent work on diagnosis of hybrid systems [3, 6, 8] has focused on discrete faults, and requires the pre-enumeration of the model in all modes to perform diagnosis. We present an online model-based diagnosis methodology for parametric faults in hybrid systems that is based on tracking hybrid behaviors (continuous behaviors interspersed with discrete changes), but unlike hybrid automata models [1] pre-enumeration of all system modes is avoided by generating models at runtime as mode switches occur.

The fault isolation task has to take into account possible mode changes during diagnostic analysis. The occurrence of a fault necessarily implies that one no longer has a correct model of system behavior; therefore, mode changes cannot be correctly predicted. To address this, the fault isolation task incorporates a search process, where mode changes may have to be hypothesized and incorporated into the consistency-based diagnosis scheme. The fault isolation process becomes even more complicated if fault detection is delayed, and the diagnosis algorithm has to roll back modes to identify the mode in which the fault occurred. We have developed a generic tracking, fault detection, and fault isolation scheme, which address all of the issues we have outlined above. The rest of the paper presents our hybrid modeling, tracking, and diagnosis scheme for solving the hybrid diagnosis problem.

2 Unified Modeling Framework

We use a unified modeling framework called hybrid bond graphs that extends continuous bond graph modeling [7]

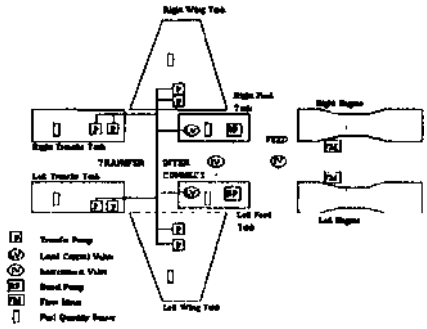

Figure 1: Fuel System Schematic

to provide compact representations for hybrid models. Its component based and hierarchical representation is expressed as topological structures that facilitate causal analysis of system dynamics. It also provides standard techniques for deriving state space and input output equation formulations that are suitable for tracking and estimation tasks [15].

2.1 Bond Graphs

A bond graph (BG) is a domain-independent topological representation that captures energy-based interactions among the different physical processes that make up the system. The vertices in the graph represent subsystems modeled as generic physical processes: capacities, inertias, and dissipators that can have linear and non-linear behaviors. Bonds are energy pathways by which subsystems/processes exchange energy in the system. Two additional types of vertices (0 and 1 junctions) represent domain independent generalizations of Kirchoff s laws and are used as connection points between the sub-systems. There exist systematic techniques to construct the bond graph from the system description [2].

2.2 Hybrid Bond Graphs

Additional mechanisms are introduced into the continuous BG language to include discrete transitions and model switching. We use switched junctions proposed by Mosterman and Biswas [13], where each junction in the bond graph may be switched on (activated) and off (deactivated). An activated junction behaves like a conventional BG junction. All the bonds incident on a junction turned off are made inactive, and hence do not play any part in the system dynamics. Note that activating or deactivating junctions affect the behavior of adjoining junctions.

A Finite State Automaton (FSA) implements the junction switching function. The FSA may have several states, and each state maps to either the off mode (i.e., it causes the junction to turn off) or the on mode (i.e., the junction turns on) of the junction. Mode transitions defined solely by external controller signals define *controlled switching,* and those expressed by internal variables crossing boundary values define *autonomous switching.* For controlled switching the control signal is provided as input to the FSA. For autonomous switching, the function determining the transition condition is provided as input to the FSA. The overall mode of the system is determined by a parallel composition of modes of modes of the individual switched junctions.

Formally, hybrid bond graphs can be defined as a triple: *HBG = {BG, M. a},* where *BG* is the bond graph model, $M = \{M_1\ M_2, \ldots, M_k f$ is a set of finite state of automata, and *a* is the mapping between each *M,* and a junction in the bond graph. Each *M,* is a finite state automaton of the type described above, with an output function that maps each state of *M,* to either on or off. A system mode change is defined by one or more junction automata changing state, and this results in a new bond graph model.

Figure 1 illustrates the fuel transfer system that we use for our experiments. The fuel system is designed to provide an uninterrupted supply of fuel at a constant rate to the aircraft engines, and at the same time to maintain the center 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), Left Transfer (LTT), and Right Transfer (RTT)) are full initially. During engine operation, fuel is transferred from the supply tanks to the receiving tanks (Left Feed (LFT) and Right Feed (RFT)) based on a pre-defined sequence. The fuel transfer sequence is controlled by valves on pipes at the outlet of the supply tanks and the inlet to the feed tanks. The hybrid bond graph segment for the connection between a wing tank and a feed tank, shown in Figure 2, illustrates the component-oriented modeling approach for the HBG. An element called switching implements the finite state automata discussed earlier. The HBG framework also associates one or more parameters with system components. We exploit this in defining a component-based diagnosis methodology, where faults in components are represented as deviations in their parameter values. For example, there are six potential fault candidates in the fuel transfer subsystem in Figure 2. (Pump, Efficiency, Wing Tank, Pipe, Switched, and Feed). In earlier work, we have shown the equivalence between the HBG framework and the hybrid automata representation [16].

2.3 Alternate Model Representations

The bond graph can be used to systematically derive alternate model representations. Three representations are used to solve different sub-tasks in the diagnosis scheme: (i) state space equations, for tracking of continuous behavior, (ii) temporal causal graphs for qualitative fault isolation, and (iii) input output equations, for parameter estimation and refinement of the fault isolation results. A

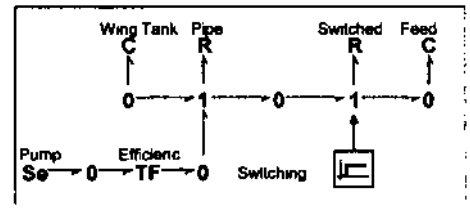

Figure 2: HBG of Fuel Transfer subsystem

detailed description of temporal causal graphs (TCG) can be found in[14, 15] and is not repeated here. Essentially the TCG captures causal and temporal relations between variables in the system. The vertices in the graph represent variables of the system and the edges (with labels) represent the types of relations between the variables.

3 MBD Architecture

Our diagnosis architecture implements a scheme to track the nominal system dynamics using an observer that is robust to model uncertainties and noise in the measurements. It uses a fault detection scheme to trigger the fault isolation scheme when discrepancies are detected between the observed and predicted measurements, tracks and analyzes the fault transients using fault signatures to isolate the fault, and then employs a quantitative parameter estimation scheme to determine the magnitude of the fault. Our work focuses on component parameter faults, which are multiplicative, i.e., faults directly affect the system dynamics models. As discussed earlier, the hybrid nature of the system complicates the tracking and diagnosis tasks, because mode transitions cause model switching, which has to be included in the online behavior tracking and fault isolation algorithms. For pragmatic reasons we simplify our algorithms by making the single fault assumption.

We have developed a novel approach that combines qualitative and quantitative algorithms for fault isolation. This extends our earlier work [12] on fault isolation in continuous systems. The qualitative approach overcomes limitations of quantitative schemes, such as convergence and accuracy problems in dealing with complex nonlinearities and lack of precision of parameter values in system models. It plays a significant role in cutting down on computational complexity, enabling online algorithms for fault isolation in the hybrid framework. The qualitative reasoning scheme is fast and effective, 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.

3.1 Tracking and Fault Detection

Our hybrid observer is implemented as a combination of an extended Kalman filter (EKF) and a hybrid automaton to track continuous behavior in individual modes of operation, and discrete mode changes (controlled and autonomous), respectively. At mode changes, the new state space model and the initial state of the system are recomputed. Model uncertainty and measurement noise are implemented as white, uncorrected Gaussian distributions with zero mean. The state space model in mode q is defined as:

$$x_{k+1} = F_q(x_k)x_k + G_q(x_k)u_{k+1} + w_k$$
$$y_{k+1} = C_q(x_k)x_{k+1} + D_q(x_k)u_{k+1} + v_{k+1}$$

where w is distributed $N(O,Q)$ and v is distributed $N(0,R)$, and Q and R are process and measurement noise covariance matrices. It is assumed that 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 [4].

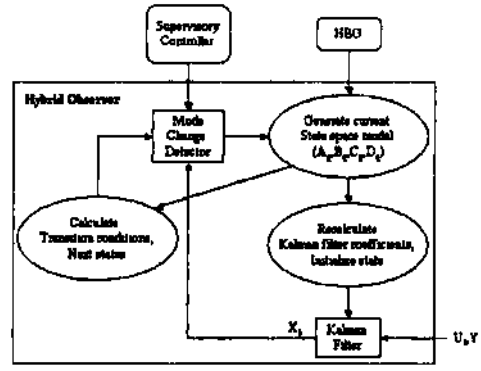

Figure 3: Hybrid Kalman Filter

Mode change calculations are based on the system mode at time step k, q_k, and the continuous state of the system, 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, x_k is used to compute the Boolean functions that signal mode transitions. Note that several transition functions may be triggered simultaneously. They are combined to derive the new system mode. A mode transition results in a new state equation model, i.e., the matrices F_q, G_q, C_v, and D_H are recalculated online. We have developed an efficient symbolic solver that can construct state equation models from equation fragments. The equation fragments correspond to constituent equations defining component behavior, and the junction relations. When switching occurs, sets of equation fragments are de-activated, and others are activated. The new state equations are then derived incrementally. 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 [15].

Fault detection is performed by first computing estimates of the output variables y_k from the state estimates x_k. We then compute a smoothed estimate y_k using an FIR filter. Finally we compute the residual $(y_k\text{-}o_k)$, where o_k are the observations at time step k. If this residual r_k is above a threshold e for a pre-defined number of time steps, then a fault is signaled.

3.2 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 is presented in Figure 4 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 at points at which discrete mode changes occur in the plant behavior. At all other time points the plant behavior is continuously differentiate. We take advantage of this fact for qualitative analyses of all measured variables, y_k. The residual (r_k) for any variable is defined as the difference between the measure plant output and the nominal expected plant output. Since both of these are continuously differentiable after the fault occurrence, and after each mode change, the residual can be approximated by the Taylor scries expansion:

[illegible] 1! [illegible] 2! [illegible] k! [illegible]

We can then represent the residual as the coefficients of the magnitude and higher order derivative terms of the residual. Instead of representing them in quantitative form, qualitative values (-, 0, and +) are used to indicate if the coefficient is below, at, or above zero.

The qualitative analyses that comprise of the roll-back and roll-forward steps work on these qualitative coefficients. After detection of the fault, the signal to symbol generator is responsible for converting the measured observations to symbolic form at each time step. The discussion of the computation of the residual and the converting it to symbolic form is presented in [11], and not repeated here.

The roll-back algorithm can be summarized as follows. Given the observer estimated mode trajectory $Q = \{q_{1\cdot q2}, \ldots, q_k\}$, we first use the back propagation algorithm [14] 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 $q_{k-1}, q_{k-2}, \ldots, q_{k-n+1}$, where n is a number determined externally by diagnosability studies. During the crossover from a mode to a previous mode, the symbols need to be propagated back across the mode change. This is done by using the inverse of the reset functions (y^1) associated with the mode transition. For example, the symbols to be propagated in mode q_{k-t} is obtained as $y^1(\alpha, q_k, q_{k-1})$. The hybrid hypotheses generation algorithm returns a hypotheses set, $H = \{h_1, h_2, \ldots, h_m\}$, where each hypothesis h_i is a three-tuple $\{q,p,k\}$, where q represents the mode in which the fault is hypothesized to have occurred, p is the parameter whose deviation corresponds to the fault, is the direction of deviation of parameter P.

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 estimated by the hybrid observer may not be the actual mode of system under hypothesized fault conditions. We need to estimate the current mode of the system for each hypothesis. However, we still do not have the quantitative value for the faulty parameter, implying that we cannot accurately determine the current mode of the system. To

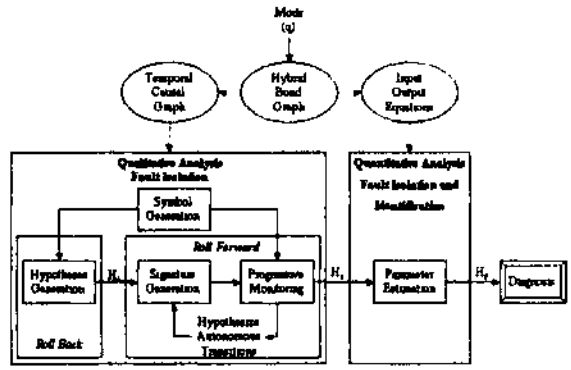

Fig. 4: Fault Isolation and Identification Architecture

solve this problem we take advantage of the fact that the same sequence of mode transitions in any order would lead the system to the same end state. This follows from the fact that each mode transition essentially changes the status of a switched junction in the hybrid bond graph representation. Based on this observation, for each hypothesis, we apply all controlled transitions (assuming that no autonomous transitions have occurred) that have occurred since the hypothesized fault mode to get an estimated current mode. This is known as the roll-forward process.

We can now use the model (TCG) to predict the expected qualitative values for the residuals (signatures) in the current estimated mode for each hypothesis. This is done through a forward propagation algorithm [14]. The fault hypotheses in the estimated current mode are compared with the symbols generated by the signal to symbol generator using a progressive monitoring scheme [14]. If there is a mismatch, the hypothesis cannot be dropped immediately, we assume that an autonomous mode transition may have caused this mismatch. We apply all possible autonomous transitions to the current estimated mode and derive m new estimated current modes for the hypothesis, where m is the number of possible autonomous transitions. For each of these m new modes, we can generate the qualitative signature using forward propagation and compare this against observations. In case of mismatch we hypothesize occurrence more autonomous transitions and repeat the process. Once the total number of transitions (controlled + hypothesized autonomous) exceeds the diagnosability limit, further mismatches in signatures and symbols eliminate hypotheses.

In other work [12], we have shown the limited discriminatory capabilities of the qualitative progressive monitoring scheme. This often leads to multiple fault hypotheses being reported as the diagnostic result. Even when we are left with only one hypothesis, determining the magnitude of the parameter associated with the hypothesis is essential to continue tracking the plant behavior in the faulty situation. We use a parameter estimation technique based on the least-squares estimation method for further fault isolation and identification. Applying a statistical hypothesis testing scheme to the error in the fault identification task leads to unique fault isolation,

and an estimate of the deviated parameter value. Consider the input output equation form:

$$Y = ((\Sigma_i \Sigma_j b_{ij}.q^j)/(\Sigma_k a_k.q^k)).U,$$

where *U* arc the inputs, *Y* are the outputs, *a's* and *b's* are constant coefficients and *q* is the forward delay operator. The typical parameter estimation task is to estimate the a's and *b's* using measurements, *u* and *y*. The optimal estimate is given by: $\theta_{est} = (\Phi^T \Phi)^{-1} \Phi^T Z$, where *0* is a vector of *a's* and b's, Φ is the regression vector, and Z is current output vector. The single fault assumption implies that only one parameter in unknown. So, for each remaining hypothesis, we rewrite θ in terms of the corresponding hypothesized fault parameter *p*. If we assume a first order polynomial relation (it is possible to extend this to arbitrary polynomial relations): $\theta = K_1 p + K_0$, where K_1 and K_0 are matrix constants. Now we can reformulate the estimation problem and obtain the optimal estimate for/7 as:

$$p = (K_1^T \Phi^T Z - K_1^T \Phi^T K_0 - K_1^T \Phi^T K_1 - Z^T \Phi K_1)/(2.K_1^T \Phi^T K_1)$$

Table 1: Trace of Fault Isolation and Estimation Left Wing Tank Pump Degradation

Time	Measured Deviation	Fault Set
433	Transfer Manifold Pressure (XMP)	13 fault candidates
434	XMP: discontinuous	10 candidates
439	*Mode Change: Left Feed Tank On*	
465	*Mode Change: Left Feed Tank Off*	
469	LeftWing Tank Pressure ++	LWT Pipe.R + LeftWingTank TF - LeftWingTank.R + Leg26.R -
471	*Mode Change: Left Wing Tank -- Second Pump On*	
489	Left Feed Tank Pressure --	LWT_Pipe.R + LeftWingTank.TF - LeftWingTank.R +
528	*Mode Change: Left Feed Tank On*	
528	Parameter estimation started	**LeftWingTank.TF fault coefficient: 0.658041**

For each remaining hypothesis, we compute the input output equations of the system in the estimated current mode from the bond graph. This can be achieved by computing the temporal causal graph from the bond graph. The TCG can then be used to compute the signal flow graph, which can be used to derive the input output equations. Please note that the signal flow graph still contains parameters in symbolic form as opposed to actual numeric values. This gives us a parameterized input output equation. We then calculate the K_1 and K_0 matrices. Finally we accumulate *u* and y values, and estimate *p* using the above expression. This may also be used for fault isolation by plugging back the estimated parameter in the state space equations and computing the predicted values for the outputs. Hypotheses whose predicted output values are statistically different from actual output values are eliminated. Table 1 illustrates an experimental run for a left wing tank pump degradation (33% loss) at time step 150. The initial change was observed in the transfer manifold pressure at time step = 433 (Fig. 5), but two mode changes occurred between the fault occurrence and its detection. The roll back process was employed, and this produced an initial list of 13 candidates. Detecting that the transfer manifold pressure was discontinuous reduced the candidate set to 10 faults. As other measurements deviated over time (first left wing tank pressure, then left feed tank pressure), the candidate set was further refined. Mode changes required the re-derivation of the system models, and the computation of new signatures to track system behavior. The qualitative scheme reduced the candidate set to 4, and then the parameter estimation scheme was invoked. This resulted in generating the correct fault hypothesis, and a correct estimation of the faulty parameter.

4 Experimental Results

We demonstrate the effectiveness of our diagnosis scheme on a real world example, the fuel system of fighter aircraft (Fig. 1). The pumps are modeled as a source of effort (pressure) with a transformation factor that defines its efficiency. The pipes are modeled as nonlinear resistances.

The diagnosis experiments used a controller sequence provided by Boeing. The performance of the hybrid observer in tracking the nominal data (with 3% noise) through mode transfers is illustrated in Figure 5 for the transfer manifold and left wing tank pressure measurements. The dots represent the measured data and the black line shows the observer estimates. A number of diagnosis experiments were run for different noise values and fault magnitudes (see [15] for details).

Table 2: Fuel System Diagnosability

	WTP	WTR	TTP	TTR	SPR	FTP	FTR
WTP	-	x	√	√	√	√	√
WTR	x	-	√	√	√	√	√
TTP	√	√	-	x	√	√	√
TTR	√	√	x	-	√	√	√
SPR	√	√	√	√	-	√	√
FTP	√	√	√	√	√	-	x
FTR	√	√	√	√	√	x	-

Table 2 summarizes the different fault classes that can be distinguished by our diagnosis algorithms. The fault classes are as follows: (i) Wing Tank Pump (WTP), (ii) Wing Tank Resistance (WTR), (iii) Fuselage Tank Pump (TTP), (iv) Fuselage Tank Resistance (TTR), (v) Switched Pipe Resistance (SPR), (vi) Feed Tank Pump (FTP), and (vii) Feed Tank Resistance (FTR). The V mark in row i and column j indicates that fault class i can be distinguished from fault class j. The x mark indicates that the current controller sequence and set of measurements are not sufficient to distinguish between the pair in question. From the table, we see that we cannot distinguish between tank pump faults and tank outlet pipe resistance faults. However, this is true only for a pump efficiency (T_F-) decrease and pipe resistance increase (R+). Since the pump efficiency cannot increase (no T_F+ **fault),** pipe resistance decreases, R-, (i.e., leaks) can be uniquely identified. All other classes of faults can be distinguished

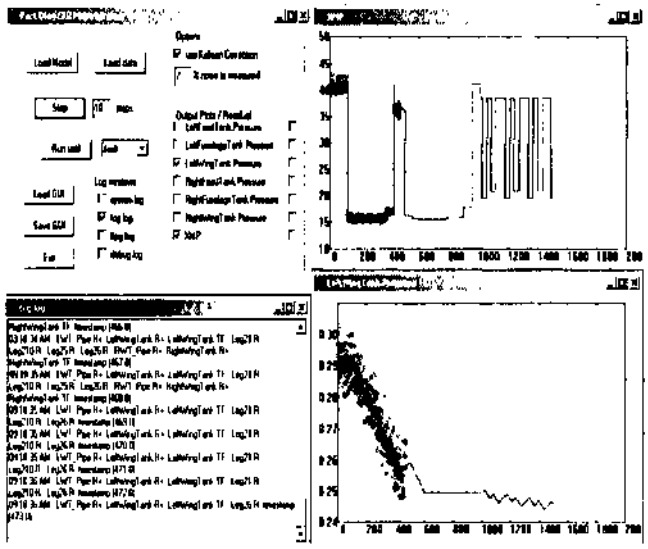

Figure 5: Transfer manifold and Left Wing Tank Pressure

from one another. In sonic cases, the isolation may be achieved only after mode changes occur.

5 Summary

In this paper we have presented an integrated approach to solving the tracking, fault detection, isolation, and identification tasks for hybrid systems. The novel contribution of the presented work is the extension of continuous system model-based diagnosis techniques to hybrid systems. These include the hybrid observer that combines an extended Kalman filter and hybrid automaton, hybrid fault isolation through roll back and roll forward using qualitative analysis, and the single parameter estimation for further fault isolation and identification. Our work is motivated by the requirements of 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 [15]. 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 like to thank Dr. Gabor Karsai, Tivadar Szemcthy, Nagabhushan Mahadevan, and Eric Manders for their help.

References

1. Alur, R., et al., *Hybrid Automata - An Algorithmic Approach to Specification and Verification of Hybrid Systems, Lecture Notes in Computer Science: Hybrid Systems 1.* 1994, Springer Verlag. p. 209-229.

2. Biswas, G. and X. Yu. *A Formal Modeling Scheme for Continuous Systems: Focus on Diagnosis,* in *Proc. IJCA1* 1993. Chambery, France, p 1474-1479.

3. Dearden, R. and D. Clancy. *Particle Filters for Real-Time Fault Detection in Planetary Rovers,* in *Thirteenth Intl Wkshp. on Principles of Diagnosis,* 2002. Semmering, Austria.

4. Gelb, A., *Applied Optimal Estimation.* 1979, Cambridge, MA, USA: MIT Press.

5. Gertler, J.., *Fault detection and Diagnosis in Engineering Systems.* 1998, New York: Marcel Dekker.

6. 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.

7. Karnopp, D., D.L. Margolis, and R.C. Rosenberg, *System dynamics : modeling and simulation of mechatronic systems.* 3rd ed. 2000, New York: Wiley.

8. Koutsoukos, X., et al. *Fault Modeling for Monitoring and Diagnosis of Sensor-Rich Hybrid Systems,* in *IEEE Conference on Decision and Control.* 2001. Orlando, FL.

9. Lunze, J. and J. Schroder, *Sensor and Actuator Fault Diagnosis of Systems with Discrete Inputs and Outputs.* in press, IEEE Transactions on Systems, Man, and Cybernetics, March, 2002.

10. Manders, E. and G. Biswas. *Transient Detection and Analysis for Diagnosis of Abrupt Faults in Continuous Dynamic Systems,* in *Intl Wkshp. on Intelligent Signal Processing (WISP '()!).* 2001. Budapest, Hungary.

11. Manders, E., P.J. Mosterman, and G. Biswas. *Signal to Symbol Transformation techniques for Robust Diagnosis in TRANSCEND,* in *Intl Wkshp on Principles of Diagnosis.* 1999. Loch Awe, Scotland.

12. Manders, E., et al. *A combined Qualitative/Quantitative approach to efficient Fault Isolation in complex dynamic systems,* in *Intl. Safe Process Symposium,* 2000. Budapest, Hungary, p.512-517.

13. Mosterman, P.J. and G. Biswas, *A Theory of discontinuities in Physical System Models.* Journal of Franklin Institute, 1998. 335(B): p. 401-439.

14. 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.

15. Narasimhan, S., *Model-based Diagnosis of Hybrid Systems,* in *Computer Science.* 2002, Vanderbilt University: Nashville.

16. Narasimhan, S. and G. Biswas. *An Approach to Model-Based Diagnosis of Hybrid Systems,* in *Hybrid Systems: Computation and Control (HSCC '02).* 2002. Stanford, CA: Springer Verlag.

17. Sampath, M., et al., *Failure Diagnosis using Discrete-Event Models.* IEEE Transactions on Control Systems Technology, 1996. 4: p. 105-124.