An Approach to Model-Based Diagnosis of Hybrid Systems

Sriram Narasimhan and Gautam Biswas

Vanderbilt University
Department of Electrical Engineering and Computer Science
Box 1679, Station B
Nashville, TN 37235.
{nsriram, biswas}@vuse.vanderbilt.edu

Abstract. The need for reliability and robustness in present day systems requires that they possess the capability for accommodating faults in the controlled plant. Fault accommodation requires tight integration of online fault detection, isolation, and identification with the system control loop. This paper develops an effective fault diagnosis scheme for plants that mainly exhibit continuous behavior, but include supervisory control making the overall dynamic behavior hybrid in nature. We use hybrid bond graph models to develop diagnosis algorithms that combine hybrid behavior tracking, mode estimation, and qualitative-quantitative reasoning techniques. The effectiveness of the approach is demonstrated with example scenarios generated from a three- and five-tank systems with control valves and sources that can be turned on and off by supervisory control.

1 Introduction

The need for reliability and robustness in present day complex systems requires that they possess the capability for accommodating faults in the controlled plant. Fault accommodation combines fault detection, isolation, and identification [1] with the determination of appropriate control actions to mitigate the effect of the faults and help maintain nominal system operation. The fault diagnosis task must be performed on line, and has to be tightly integrated with the system control loop. This motivates online model-based approaches to diagnosis that provide sufficient information for fault adaptations by the supervisory controller.

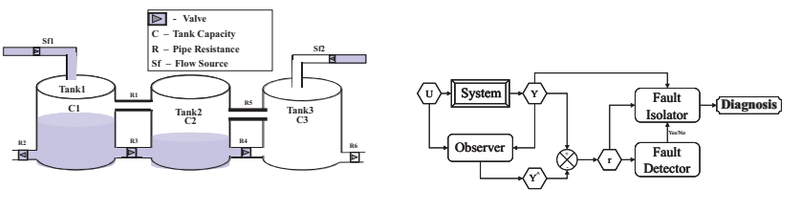
Aircraft subsystems, chemical processes, and manufacturing plants combine continuous dynamics with discrete supervisory control. Therefore, overall system behavior is necessarily *hybrid.*Discrete changes in the system can be attributed to supervisory controller input (*controlled jumps*) and changes defined by internal variables crossing threshold values (*autonomous jumps*)[1]. Fig. 1(a) presents a simple example of such a system: three tanks with flow sources, connecting

[1] Autonomous jumps are often attributed to modeling simplifications introduced to avoid complex non-linearities in system behavior [9].

C.J. Tomlin and M.R. Greenstreet (Eds.): HSCC 2002, LNCS 2289, pp. 308–322, 2002.

pipes, and outlet pipes. The valves on all the pipes and sources are commanded by signals from the supervisory controller. The upper connecting pipes between the tanks become active only when the fluid levels reach pre-defined heights. They represent autonomous transitions.

Model-based diagnosis starts with a framework that links system behavior to system components and parameters. Most real systems are equipped with a limited number of sensors to track system behavior, and analytic redundancy methods have to be applied to derive non-local interaction between potential faults and observations. These techniques have been applied to a variety of schemes used in the diagnosis of discrete [2], discrete event [3] and continuous systems [1,4,5]. They have also been applied to hybrid system diagnosis, using a single continuous model with complex non-linearities, or abstracting the continuous dynamics to a discrete event model. Complex nonlinearities complicate the analysis and they may introduce numerical convergence problems. Discrete event abstractions lead to loss of critical information, such as fault transient characteristics. Further, methods to identify the set of events that describe both nominal and faulty behavior is often a computationally challenging task bringing to question the scalability of such approaches [6]. Hybrid system analyses require the use of multiple models of the system. As a result, appropriate model selection and switching has to be performed at run time to execute tasks like simulation, monitoring, control, and fault isolation. This paper discusses a model-based diagnosis



(a) Three-Tank System

(b) Model-Based Diagnosis Architecture

Fig. 1.

methodology for hybrid systems that builds on our previous work in continuous system diagnosis [5]. Since the overall goal is fault accommodation, the diagnosis algorithm has to be:(i) online, (ii) employ analytic redundancy methods since the number of sensors are small, (iii) based on analysis of fault transients so that faults can be quickly isolated and identified.

2 The Hybrid Diagnosis System

A complete diagnosis scheme for the type of systems displayed in Fig. 1(a) must consider faults in the plant and the supervisory controller. A faulty controller issues commands that are in conflict with the desired functionality. We do not deal with these faults in this paper, and make the assumption that the controller has no faults. A lot of work in the FDI community has dealt with fault isolation filter design using structured and directional residual methods that mainly apply to additive faults [1]. When dealing with multiplicative faults (e.g., changes in the plant parameters) that affect the dynamic response of the system, researchers have resorted to parameter estimation techniques that are often computationally complex and hard to implement online. Our focus in this paper is on plant component faults. We assume that these faults can be parameterized, and a fault is represented as a persistent step change in a plant parameter value. In dynamic systems, a step change in a parameter value causes a *transient response* that is superimposed on the nominal plant dynamics. In previous work, we have developed qualitative schemes to characterize and analyze these transients to solve the fault isolation problem in a way that mitigates some of the problems of the numerical schemes [5,11]. We extend these schemes for continuous diagnosis to address the more complex problem of hybrid diagnosis.

We restrict ourselves to systems whose models can be described as piecewise linear hybrid dynamical models. These models are sufficient to describe a large class of engineered systems [6]. The discrete time state space parameterized model of a hybrid system is given by

$$\begin{aligned} x(t+1) &= A_{q(t)}(P)x(t) + B_{q(t)}(P)u(t) \\ q(t+1) &= \delta(q(t), \pi(x(t)), \sigma(t)) \\ y(t) &= C_{q(t)}(P)x(t) + D_{q(t)}(P)u(t), \end{aligned}$$

where $x(t) \in R^n$ is the state vector of the plant, $u(t) \in R^m$ is the input to the plant, and $q(t) \in I$ is the discrete mode that can take on a finite number of values. The system matrices$A$, B, C, and D are functions of the physical plant parameters, P. For such systems it has been shown that the state space can be divided into polyhedral regions, where each region corresponds to a mode of system operation [6].

In our work, we are interested in diagnosing *step changes* in a single deviated parameter $p_i \in P$. Our model-based approach (Fig. 1(b)) uses a *hybrid observer* to track nominal system behavior, a fault detection mechanism, and a fault isolation and identification unit. The observer uses a quantitative hybrid model of the plant to follow a hybrid trajectory. The mode of operation, q, and the continuous state vector in that mode, x defines the system state. Mode changes are defined by controlled and autonomous changes. For controlled transitions, it is assumed that the controller signals are input to the diagnosis system. Autonomous changes are defined by events that are functions of system variables. Mode identification is complicated by measurement noise and imperfections in the system models. The resulting inaccuracies and delay in the observer can

affect the accuracy and precision in predicting autonomous transitions. Mode transitions require computation of the new mode of system operation, and the initial state in the new mode using the *reset* function. We have adopted a combination of a hybrid automata and Kalman filtering approach to design our *hybrid observer* [7]. Small differences, attributed to minor imperfections in the model and noise in the measurements, are compensated for in the observer mechanism. When the differences become significant, the *fault detection unit* signals the presence of a fault, and this triggers the *fault isolation unit*, which generates candidate faults and refines them by analyzing subsequent measurements from the system.

3 Modeling Hybrid Systems

Our approach to modeling hybrid systems is based on an extended form of bond graphs [8], called *Hybrid Bond Graphs* (HBG). Bond graphs present a methodology for energetic modeling of physical systems. A bond graph is made up of capacitors and inertias, dissipative elements, and ideal sources of effort and flow, (ii) $0-$ *and* $1-$ *junctions* (correspond to parallel and series connections, respectively) that define the interconnectivity between components, and (iii) *bonds* that represent the energy transfer pathways in the system. System behavior is based on component characteristics and the principles of continuity and conservation of energy that are imposed by the junction relations. Extensions to hybrid systems require the introduction of discrete changes into the model configuration. In the HBG framework, discontinuities in behavior are dealt with at a *meta-model* level, where the energy model embodied in the bond graph scheme is frozen for a time instant, and discontinuous model configuration changes are executed as instantaneous junction switching. The switched junctions act as idealized switches that turn energy connections on and off, and do not violate the physical principles of energy conservation [9]. Hybrid Bond Graphs can be formally defined as:

$$HBG = \{BG, M\}, \tag{1}$$

where BG is a continuous bond graph model, $M = \{M_1, M_2, \ldots M_k\}$, are a set of finite state machines, with each $M_i \in M$ being assigned to a switched junction of the HBG. Each finite state machine M_i is formally defined as:

$M = (Q, \sum, \delta, \mu, q_0)$
$Q = \{q_1, q_2, \ldots, q_n\}$ is the finite set of states
$\sum = \sum_c \bigcup \sum_a$ is the set of discrete events,
$\sum_c$ is the finite set of controller events,
$\sum_a : \{f_a(x) > 0\}$, where x is the continuous state vector,
$\delta : Q \times \sum \rightarrow Q$ is the transition function,
$\mu : Q \rightarrow \{on, off\}$ is the output function, and
$q_0 \in Q$ is the initial state.

A system mode transition occurs when one or more automata M_i undergo a state transition. The sequence of transitions establish the on/off state of all the individual switched junctions, and defines a new mode q for the system. The switching conditions are specified so that each combination of junction states correspond to a polyhedron in the state space. If only one automata triggers at a point in time, each polyhedron is adjacent to at most n polyhedra, where n is the number of switched junctions. After the discrete mode transition, system behavior evolution is again continuous till another point in time when a M_i changes state. Assigning individual automata to each switched junction provides a compact representation of the system model across all its nominal modes of operation.

Fig. 2 represents the HBG model of the three-tank system (Fig. 1(a)). The three tanks are modeled as capacitors, and the pipes are modeled as simple resistances. Some junction states are determined by controlled signals. Others, such as junction 4 and 6, whose on-off transitions depend on the height of the liquid column, are autonomous. Hybrid automata models (e.g., [10]), traditional

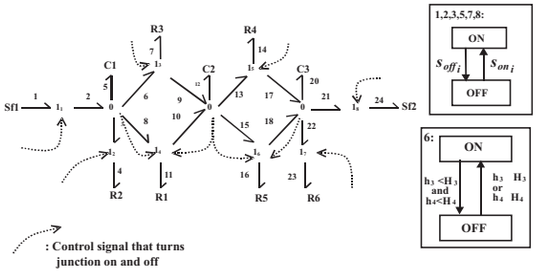


Fig. 2. Hybrid Bond Graph Model of Three-Tank System

computational models used in hybrid systems community are easily derived from HBG models of a system. Typically a linear hybrid automaton is defined as:

$$A = (X, V, flow, inv, init, E, jump, \Sigma, syn),$$

where X is the real-valued state vector, V is the finite set of system modes, *flow* describes the equations governing X in each mode, *inv* assigns an invariant condition to each mode, and *init* assigns initial conditions to each mode (a mode is initiated if and only if it's initial conditions are true), E describes a finite set of mode switches represented as a directed edge between two modes, *jump* represents the change in values of X after a mode switch, Σ describes the finite set of events, and *syn* assigns an event to each mode switch.

The hybrid automata model A is easily derived from a HBG model by setting X = {a subset of effort and flow variables in the bond graph BG}, V = $\{\mu(\mathrm{M}_1),\mu(\mathrm{M}_2), \ldots ,\mu(\mathrm{M}_k)\}$, where (μ) takes on two values on/off as described earlier. Hence the automata has a total of 2^k modes. For each mode, *flow* can be

obtained by deriving the bond graph for that mode and then deriving the state space equations from the bond graph. $inv = !\Sigma$. Events Σ force transitions in the FSM's, and hence the system can stay in a mode only when these events do not occur. $init = \Phi$. $E = \bigcup \delta(M_i) \forall M_i \in M$. $jump = previous(X)$. There is no change to the state as a result of a mode change. $\Sigma = \bigcup \Sigma(M_i) \forall M_i \in HBG$. $syn = \bigcup \delta(M_i) \forall M_i \in M$.

We use the HBG models of our hybrid system to derive hybrid automata models for the system. Within each state of the automata (i.e., mode of the system). we can derive (i) state space equations, (ii) temporal causal graphs, and input output equations from the corresponding BG model. Each of these models are employed for different tasks in our diagnosis engine.

4 Fault Isolation

In previous work, we have developed a qualitative reasoning scheme based on the analyses of *transients* caused by step changes in parameter values for fault isolation in continuous systems [5]. This methodology uses a framework developed by AI researchers, where the fault isolation task is broken into two parts: (i) *hypothesis generation* and (ii) *hypothesis refinement*. The hypothesis refinement algorithm uses a progressive monitoring technique that we describe in more detail later. Qualitative reasoning schemes provide efficient means for initial hypotheses generation and refinement, but lack precision for fault identification. To overcome this problem, we combine qualitative reasoning with least squares based parameter estimation techniques for fault isolation and identification [11].

The fault isolation task in hybrid systems is complex because the discrete mode transitions cause changes in the model and may cause discrete changes in variable values. The residual analysis scheme must, therefore, include mode identification techniques to achieve correct fault isolation. However, the occurrence of a fault invalidates the system model and, therefore, the mapping between the state space and the measured variables. Hence the state estimates made by the observer may no longer be correct. Abrupt faults (i.e., step changes) may produce a discontinuous jump in the values of the state and output variables. This abrupt change may result in a sudden transition from one mode to another. For example, an abrupt decrease in the capacitance of tank 1 (Fig. 1(a)) (caused by an object falling into the tank) would cause a sudden discontinuous increase in the pressure. The increase may render pipe R1 to become active changing the system mode, and therefore, the system model. In addition, the polyhedral boundary region definitions are functions of system parameter values. A fault, i.e., a change in a parameter value, can, therefore, cause a change in the polyhedral partition of the state space. Therefore, autonomous mode transitions may no longer be correctly predicted after the occurrence of a fault. Therefore, designing of fault observers becomes a nontrivial task. In our work, we mitigate this problem to some extent by using qualitative tracking schemes.

A second problem with hybrid fault isolation is that the fault may be detected only after mode transitions have occurred. Therefore, the current system model

does not provide the right set of constraints for initial hypothesis generation. To solve this problem, one has to introduce a fast *roll back* process to determine the possible modes in which the fault could have occurred. Solving the fault isolation problem requires determining the mode along with the parameter value change that explains the observed discrepancies in system behavior. The residual analysis task for hypothesis refinement requires a fast *roll forward* process to determine the current mode of the system. Multiple hypotheses may be generated, and the fault identification process then determines the true fault. The

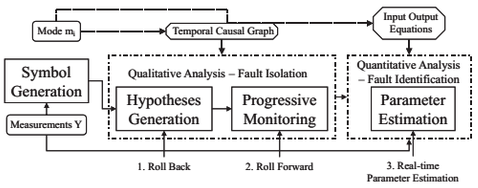


Fig. 3. Hybrid Fault Isolation and Identification

overall scheme for hybrid diagnosis is illustrated in Fig. 3. Like the continuous case, we overcome limitations of quantitative schemes by combining robust qualitative reasoning mechanisms with quantitative techniques. The fault isolation methodology for hybrid systems is broken down into three steps:

A fast *roll back process* using qualitative reasoning techniques to generate possible fault hypotheses. Since the fault could have occurred in a mode earlier than the current mode, fault hypotheses need to be characterized as a two-tuple *(mode, fault parameter)*, where mode indicates the mode in which the fault occurs, and fault parameter is the parameter of an implicated component whose deviation possibly explains the observed discrepancies in behavior.

A quick *roll forward process* using progressive monitoring techniques to refine the possible fault candidates. The goal is to retain only those candidates whose fault signatures are consistent with the current sequence of measurements. After the occurrence of a fault, the observer's predictions of autonomous mode transitions may no longer be correct, therefore, determining the consistency of fault hypotheses also requires the fault isolation unit to roll forward to the correct current mode of system operation.

A *real-time parameter estimation process* using quantitative parameter estimation schemes. The qualitative reasoning schemes are inherently imprecise. As a result, a number of fault hypotheses may still be active after Step 2. We employ least squares based estimation techniques on the input-output form of the system model to estimate consistent values of the fault parameter that is consistent with the sequence of measurements made on the system.

4.1 Hypotheses Generation Using Roll Back Process

Since the fault could have occurred in a mode that is different from the mode in which the fault is detected, we need to consider previous modes when generating hypotheses. Without knowledge of the current mode, it is not possible to determine, what modes the system could have been in previously (using the pre operator defined in [6]).

Lemma 1 (Fault Mode) *If we assume that the observer can accurately track the system under nominal conditions, then the mode in which the fault occurred has to be part of the observer mode trajectory.*

Lemma 1 implies that it is sufficient to generate hypotheses only in modes in the observer predicted trajectory. We can further restrict this using diagnosability studies to determine that any fault must manifest itself in the form of significant transients within k modes. This defines the roll back process for hybrid diagnosis.

Definition 1 (k Diagnosability) *A system is said to be k diagnosable if the effects of any fault manifest themselves within k mode transitions.*

Hypotheses generation within a mode is based on a back propagation algorithm performed on the temporal causal graph (TCG) derived from the bond graph model in that mode. In the TCG, we start at the node(s) corresponding to these discrepancy(s) expressed as + or $-$[2] and propagate them against the direction of arrows. When we traverse an edge that contains a parameter, we include that parameter as a fault candidate (See [5] for more details). Let *BackProp(q,δ)* denote the back propagation algorithm performed on the TCG model of the system in mode q with initial symbolic discrepancies δ.

Suppose $\{q_1, q_2, \ldots, q_n\}$ is the observed mode trajectory of the system. q_n is the mode in which the fault was detected. Let δ_n denote the symbolic discrepancies detected. *Reset*(q_1,q_2,δ_n) denotes the reset function when the system transitions from mode q_1 to mode q_2. Let *InverseReset* denote the inverse of the *Reset* function. Hypotheses generation (hybrid back propagation) algorithm is summarized as:
$\delta_{current} = \delta_n$
For mode = 1 to k
BackProp($q_{n-mode-1}$, $\delta_{current}$)
$\delta_{current}$ = *InverseReset*(q_{n-mode}, $q_{n-mode}-1$, $\delta_{current}$)

We first run back propagation in the mode in which the fault is detected using the observed discrepancies and generate candidates in that mode. Then we go back in the observer predicted mode trajectory and generate candidates in each of those modes. Note that the TCG in each of these modes is different. Discrepancies are propagated back across modes by applying constraints specified by the appropriate reset functions. Note that the sign of the propagation may be undetermined (both + and –) because of the ambiguity of qualitative arithmetic.

[2] The algorithm for generating symbolic discrepancies is described in [12]

In such cases we have to propagate both + and - values across the mode for hypotheses generation in the previous mode. Note that backward propagation is applied only once, immediately after a fault has been detected.

Definition 2 (Fault Hypotheses) *A fault hypothesis is defined as the pair* (q_i,p_j) *where* q_i *represents the mode in which the fault occurred and* p_j *represents the deviating parameter.*

4.2 Hypotheses Refinement Using Roll Forward Process

Qualitative hypothesis refinement compares signatures generated from the model against the symbolic representation of the measurement transient as it evolves in time. Inconsistent fault hypothesis are dropped as the tracking progresses. To generate signatures, we need to determine the current mode of the system and derive the system model in the mode using the hybrid bond graph. Since the system state is unknown after the occurrence of a fault, the hypotheses refinement needs to solve the mode identification problem. Using the assumption that the controller signals are known, the controlled transitions that have occurred between the hypothesized fault mode (for each candidate hypothesis) and current time are known. We apply these known mode changes to get a hypothesized current mode. We call this a quick *roll forward* process.

Lemma 2 (Sequence of Mode Transitions) *A sequence of k mode transitions occurring in any order would drive the system to the same final mode if the system starts in the same initial mode.*

Lemma 2 can be justified by looking at the representation of a mode in the hybrid bond graph. Since each junction state is dependent only on its local automata, the order in which the junction states are changed does not matter. Note some sequences may not be physically valid. For example, if the level of fluid in tank 1 is below the pipe R1 and is decreasing, we cannot assume that the autonomous transition that makes R1 active will occur next.

For each fault hypothesis (q_i, p_j) we need to determine what the current mode is. In order to do this we first determine all controlled transitions that occurred since the time the system was in mode q_i and current time. If this set is given by $\{\Sigma_1, \Sigma_2, \ldots, \Sigma_n\}$, then we can hypothesize the current mode as $q_{current} = \delta \ldots (\delta (\delta (q_i, \Sigma_1), \Sigma_2), \ldots, \Sigma_n)$.

The qualitative progressive monitoring techniques used for hypotheses refinement within a mode is discussed in section 4.2. Let ProgressiveMonitoring ((q_i, p_j), $q_{current}$) denote the results of progressive monitoring on fault candidate (q_i, p_j) in mode $q_{current}$. If the progressive monitoring indicates that the predictions based on the hypothesized fault do not match the observed output, we cannot drop the hypothesis because our hypothesized current mode may not be the right current mode. Since we have complete knowledge of controlled transitions, the implication is that some autonomous transition could have occurred in the system (due to the effects of the fault) that the observer could not predict.

In order to determine the possible autonomous transitions, we define a qualitative forward prediction operator (inverse of the pre operator defined in [6]) to identify which modes the system may have reached from the current hypothesized mode. The operator looks at the fault signature for different variables and determines if a possible autonomous transition is consistent with a predicted signature. This restricts the number of autonomous changes we need to consider. Let HypothesizeAutonomous(q) denote the possible autonomous transitions out of mode q.

We apply each of these selected autonomous transitions in turn to generate a new mode. In each of these new modes we repeat the qualitative hypotheses refinement process. We keep hypothesizing autonomous transitions till the total number of transitions from the hypothesized fault mode to the hypothesized current mode exceeds k, at which point we drop the candidate if there is still a discrepancy. As discussed before this k is a function of the diagnosability measure. This approach will eventually lead us to the correct current mode based on Lemma 2.

The hypotheses refinement algorithm for each fault hypothesis (q_i, p_j) is:
$q_{current} = \delta \ldots (\delta\ (\delta\ (q_i,\ \Sigma_1),\ \Sigma_2),\ \ldots,\ \Sigma_n)$
if !*ProgressiveMonitoring*($(q_i,\ p_j)$, $q_{current}$)
Σ_a = HypothesizeAutonomous ($q_{current}$)
For each $\Sigma \quad \varepsilon \quad \Sigma_a$
$q_{current} = \delta\ (q_{current},\ \Sigma)$
Repeat step 2.

We next discuss the qualitative hypotheses refinement process within a mode (progressive monitoring).

Qualitative Hypotheses Refinement in Continuous Regions As discussed, faults are represented as step changes in parameter values, and the occurrence of faults produces transients in system behavior. Previous work has shown that discontinuous changes in variable values can only occur at the point of failure, thus system behavior is continuously differentiable before and after the occurrence of a fault. Therefore, the transient response in a measurement after the time point of failure, t_0, can be approximated by the Taylor series expansion. If $r(t_0)$ is the value of the residual signal just after the occurrence of the fault, thek^{th} order Taylor series expansion for *r(t), $t \geq t_0$* is given by

$$r(t) = r(t_0) + r'(t_0)\frac{(t-t_0)}{1!} + r''(t_0)\frac{(t-t_0)^2}{2!} + \cdots + r^k(t_0)\frac{(t-t_0)^k}{k!} + R_k(t),$$

where $R_k(t)$ is a remainder term based on higher order derivatives of r and $r^k = \frac{\partial^k r}{\partial t^k}$

For most well behaved functions the series converges, therefore, the Taylor series is a good approximation of the true signal $r(t)$ when t is close to t_0. The analysis of transient dynamics by interpreting the signal as a Taylor series approximation is the basis for describing the fault transient signal as a *fault signature*.

Qualitative fault isolation is based on the comparison of the fault signatures with measurements made on the system. Performing this analysis quantitatively is an intractable problem. When a fault occurs, the exact magnitude of parameter value changes is unknown, so derivative values in the fault signature have to be computed from subsequent measurements. To address this problem, we use a *qualitative constraint analysis* scheme, developed for the fault isolation task.

In the qualitative framework, individual measurements are labeled as normal (0), above normal (+) and below normal (−). Similarly, derivatives take on values, increasing (+), steady (0), and decreasing(−). The fault signature in the qualitative framework then is the sequence of +,0, or − magnitude and k derivative values computed at the point of failure, t_0. This fault signature is the basis for qualitative transient analysis using the progressive monitoring scheme.

Lemma 3 (Qualitative Transient Analysis) *Transient dynamics are captured by evaluation of the direction of abrupt change at the point of failure (if it occurs), and the signs of the derivatives of the signal after the onset of a fault.*

Fault detection triggers the fault isolation mechanism. The hypothesis generation algorithms, implemented as a two-step process, fault hypothesis generation followed by fault signature generation for each hypothesis, is described in detail in [5] An observer, defined in terms of a set of fault signatures, one for each measurement, is designed for each fault hypothesis.

Comparing the fault signature with the feature vector obtained from the evolving transient data is the basis of a *progressive monitoring* scheme for tracking signal transients [5].

Lemma 4 (Progressive Monitoring) *Qualitative magnitude and slope of a fault transient are matched against a qualitative fault signature by starting in a sequence from a discontinuous magnitude and first order change to a succession of higher order derivatives.*

Comparing the i^{th} and $(i+1)^{th}$terms in the Taylor series, one can establish $|r^i(t_0)| \geq |r^{(i+1)}(t_0)|(t\text{-}t_o)/(i+1)$ for some period of time t. As t increases, the inequality reverses and the higher order derivative starts dominating the lower one. Lemma 4 provides the basis for progressive monitoring of signal dynamics using higher order derivatives. Starting from the point of failure, t_0, the signal magnitude in response to the fault, $r(t_0)$ determines the signal value. Immediately after that the first derivative of the signal dominates the dynamic behavior because small values of $(t-t_o)$ dominate higher powers $(t-t_o)^i$ in the Taylor series. As t increases, higher order derivatives in succession increasingly contribute to the dynamics of the signal.

4.3 Fault Identification Using Real-Time Parameter Estimation

In other work ([11]), we have derived limited discriminatory capabilities of the qualitative progressive monitoring scheme. This often leads to multiple fault

hypotheses as the diagnostic result. Even if the fault isolation is unique, we still need to identify the fault. For this we use real-time parameter estimation using the least squares estimation method. A fault observer is triggered for each fault hypothesis that is still valid. Note that each remaining hypothesis has a current mode associated with it. We symbolically derive the parameterized input output equation model of the system from the bond graph model in that mode using Mason's gain rule. We substitute nominal values for all component parameters except the implicated fault hypotheses . This gives us an input output model in terms of inputs (u), measurements (y), and the fault parameter p_i. The least squares estimate is run on each fault observer, to estimate p_i. If this estimation does not converge, we may drop that corresponding hypothesis. If it converges, then we have identified the fault. In case of transitions during the estimation, we can switch modes and derive the new input output models and continue the estimation in the new mode.

5 Experiments

Fig. 4 illustrates a sample run of our diagnosis engine on the three-tank system (Fig. 1(a)). The three pressure values at the bottom of each tank are measured. We keep all the controlled pipes open at all time steps. The sources are opened at time step 20. The source into tank1 is shut off at time step 60. The source into tank 3 is shut off at time step 70. We introduce a clog in the outlet pipe from tank 1 (called Drain West) at time step 30 by changing the resistance value R2 from 1 to 10.

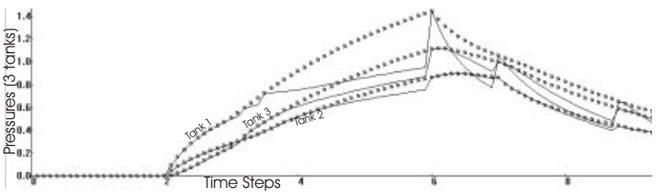


Fig. 4. Sample Run

The plot shows the actual system output as circles and the estimated observer output as solid lines. The observer tracks controlled and autonomous changes like the pipe R2 going active at time steps 32. Fault detection is triggered at time step 39. Note that the observer is unable to track system behavior after the occurrence of the fault.

Hypotheses generation and quick roll back (with $k = 1$) are initiated on the initial discrepancy: tank 1 pressure below nominal. This produces the following hypotheses in the current mode q_1(all open) and in the previous mode q_2 (all but

R5 open): C1-, R2+, C2-, R1-, C3-, R5+, R3+, R4+, and R6+ (Fig. 1(a)). The quick roll forward brings the predictions back to the current mode for all the candidates. Hypothesis refinement is initiated considering all measurement discrepancies. All the C candidates predict a discontinuous change in the pressures, which we do not see, so they are eliminated. To continue tracking, we have to hypothesize autonomous transitions for some candidates (R1 and R5) in q_1, and continue progressive monitoring. The signatures still do not match and so these hypotheses are also dropped. R2+ and R6+, the remaining candidates cannot be distinguished by qualitative analysis. The parameter estimation procedure is initiated, and the estimate converges for the left drain (R2+) to a new resistance value = 10.43. For R6+, parameter estimation does not converge.

Table 1. Experiments on Three-Tank amd Five-Tank Systems

System	Measurements	Fault (original value, new value)	Qualitative Diagnosis (Steps)	Parameter Estimated (Steps)
Three-Tank	h2,h3	C2- (1,0.2)	C2- (2)	0.21 (15)
Three-Tank	h2,h3	R1+ (1,5)	R1+ (8), R12- (8)	4.978 (16), NC
Three-Tank	h2,h3	R23- (1,0.2)	R23- (6)	0.195 (16)
Five-Tank	h2,h3,h4	C2- (1,0.2)	C2- (2)	0.203 (21)
Five-Tank	h2,h3,h4	C4+ (1,5)	C4+ (2)	4.967 (21)
Five-Tank	h2,h3,h4	R5- (1,0.2)	R5- (10), R12+ (10)	0.191 (22), NC
Five-Tank	h2,h3,h4	R34+ (1,5)	R34+ (7)	5.023 (20)

Table 1 summarizes the results of a set of experiments on a three-tank and five-tank system. The results indicate that the diagnosis algorithm scales well. For capacitance faults, the qualitative analysis can isolate the true fault using discontinuity detection. Signature orders of 3 and 5 were used for the three-tank and five-tank system respectively. The parameter estimation also identifies the correct fault magnitude. For resistive faults, the qualitative diagnosis takes longer but still converges to the correct fault hypothesis (NC implies Non-convergence for corresponding fault hypothesis). In some cases, parameter estimation is required to isolate the correct fault.

6 Discussions and Conclusion

This paper has developed a model-based approach to fault detection, isolation, and identification for hybrid systems. Our work is motivated by the fault accommodation task that requires that the diagnosis tasks be performed online while the system is in operation. We have described the complexities of tracking and analyzing hybrid behavior under faulty conditions: (i) changes in the model, (ii) the current system mode may be unknown, and (iii) since the model is faulty, the observer cannot reliably predict autonomous transitions. This would normally

lead to exhaustive search techniques to solve the fault isolation and identification problem. By introducing qualitative reasoning mechanisms, we have developed computationally simple operators that perform the equivalent of a pre and post operations defined for piecewise linear hybrid systems.

Our work contrasts the approach of Lunze [4], who maps hybrid behaviors into discrete spaces for fault diagnosis applications. This requires exhaustive pre-enumeration of trajectories to derive the automata for tracking system behavior, and does not easily scale up to large systems. On the other hand, Ferrari-Trecate et al. [13], use a mixed integer logic model formulation to solve the fault detection and isolation problem as an optimization problem. In their work, deriving mixed logic dynamic model for a general system is non-trivial. Also the computational complexity of their optimization methods makes it unsuitable for online applications.

In future work, we plan to formalize our approach in defining the pre and post operator for tracking hybrid behavior under a variety of fault conditions. We are currently working on more complex systems, like the aircraft fuel transfer system and NASA's bioplex system to demonstrate the effectiveness of our approach.

Acknowledgments: The DARPA/ITO SEC program (F30602-96-2-0227) and The Boeing Company have supported the activities described in this paper. We would like to thank Dr. Gabor Karsai, Tivadar Szemethy, and Eric Manders for their help.

References

1. Gertler, J., Fault detection and isolation using parity relations. Control Engineering Practice, 1997. 5(5): p. 653-661.
2. Kleer, J.D., An assumption-based truth maintenance system. Artificial Intelligence, 1987. 28: p. 197-224.
3. Sampath, M., et al., Failure diagnosis using discrete event models. IEEE Transactions on Control Systems Technology, 1996. 4: p. 105-124.
4. Lunze, J., A timed discrete event abstraction of continuous variable systems. International Journal of Control, 1999. 72(13): p. 1147-1164.
5. 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.
6. Koutsoukos, X. and P. Antsaklis, Hierarchical control of piecewise linear hybrid dynamical systems based on discrete abstractions, Technical Report, 2001.
7. Narasimhan, S., Biswas, G., Karsai, G., Pasternak, T., and Zhao, F., 2000. Building Observers to Handle Fault Isolation and Control Problems in Hybrid Systems, Proc. 2000 IEEE Intl. Conference on Systems, Man, and Cybernetics, Nashville, TN, pp. 2393-2398.
8. R.C. Rosenberg, and D.C. Karnopp. Introduction to physical system dynamics, McGraw-Hill, 1983.
9. Mosterman, P.J. and G. Biswas. Towards procedures for systematically deriving hybrid models of complex systems. Third International Workshop on Hybrid Systems: Computation and Control. 2000.

10. R. Alur, C. Courcoubetis, T.A. Henzinger, and P.-H. Ho. "Hybrid Automata – an algorithmic approach to specification and verification of hybrid systems", Hybrid Systems I, Lecture Notes in Computer Science 736, pp. 209-229, Springer-Verlag, 1994.
11. Manders E.J., S. Narasimhan, G. Biswas, and P.J. Mosterman. A combined qualitative/quantitative approach for efficient fault isolation in complex dynamic systems. 4th Symposium on Fault Detection, Supervision and Safety Processes, pp. 512-517, 2000.
12. Manders E.J., P.J. Mosterman, and G. Biswas. Signal to symbol transformation techniques for robust diagnosis in TRANSCEND, Tenth International Workshop on Principles of Diagnosis, Loch Awe, Scotland, pp. 155-165, 1999.
13. Ferrari-Trecate G., D. Mignone and M. Morari. Moving Horizon Estimation for Hybrid Systems, IEEE Transactions on Automatic Control, To Appear, 2002.