Fault Modeling for Monitoring and Diagnosis of Sensor-Rich Hybrid Systems

Xenofon Koutsoukos Feng Zhao Horst Haussecker Jim Reich
Patrick Cheung
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304, USA
{koutsouk,zhao,hhaussec,jreich,pcheung}@parc.xerox.com

Abstract

This paper presents a framework for modeling faults in hybrid systems that leads to an efficient approach for monitoring and diagnosis of real-time embedded systems. We describe a fault parameterization based on hybrid automata models and consider both abrupt failures and gradual degradation of system components. Our approach also addresses the computational problem of coping with large amount of sensor data by using a discrete event model of the system so as to focus distributed signal analysis on when and where to look for signatures of interest. The approach has been demonstrated for the on-line diagnosis of a hybrid system, the Xerox DC265 printer.

1 Introduction

Real-time monitoring and diagnosis of sensor-rich embedded systems such as networked printers or automotive vehicles face a number of significant challenges. Such systems are best modeled by hybrid systems that describe the continuous and discrete dynamics as well as their interactions. An important challenge addressed in this paper is modeling of faults in hybrid systems. Monitoring and diagnosis of complex embedded systems require the modeling of abrupt component failures as well as subtle component degradation. In a printer, for example, abrupt failures of components such as a broken transfer belt or a stalled motor cause the interruption of the printing operation. Component degradation can also cause the interruption of the operation. For example, paper jams are often caused by subtle component degradation such as roll slippage or timing variations of clutches, motors or solenoids due to wear, some of which is not directly observable with the system's built-in sensors and must be estimated using system behavioral models and additional sensor information.

Current model-based diagnosis techniques for hybrid systems are based on observers that estimate the state of the system at every time step. In practical systems such as printers, however, it may be difficult or very expensive to place sensors in order to obtain measurements of the complete state. For example, a printer contains a small number of photodiodes that can only detect the arrival of a sheet of paper. Measurements of the state of the components such as the speed of the motors are not available either because some of the components do not have appropriate sensors or because the measurements are used only locally, for example, by a PID controller. However, recent advances in micro-machined sensors and electronics enable us to embed a large number of sensors such as microphones or vibration sensors inside a machine. Monitoring and diagnosis in such a sensor-rich environment requires the collaborative processing of high-volume sensor data in an efficient manner while minimizing the communication cost. Traditional signature analysis and FDI techniques, although effective in a relatively sensor-poor environment, are unable to cope with the constant onslaught of sensory data, and existing model-based techniques do not scale up well to distributed, signal-rich diagnosis problems.

Our motivation for this work is the problem of monitoring and fault diagnosis in a document processing factory (or print shop) consisting of multiple printing, collating, and binding machines in proximity to each other. An example of such a machine is the Xerox Document Centre DC265 printer, a high-speed, high-capacity multi-function device that can make xerographic prints at 65 pages per minute. The DC265 printer is an embedded system consisting of a large number of mechanical components such as motors, solenoids, belts, gears whose operation is orchestrated by several distributed digital controllers. The behavior of such printing equipment can be conveniently described by hybrid systems that model the various physical phenomena using continuous dynamics and the interaction between the components using discrete events. Due to space limitations, in this paper we use a subsystem of the printer, the paper feed system, which consists of multiple components and exhibits behavior rich enough to illustrate our approach for monitoring and fault diagnosis of hybrid systems.

In this paper, we present a novel approach to monitoring and diagnosis of real-time embedded systems that integrates

model-based techniques using hybrid system models with distributed signature analysis. Faults affect the behavior of a hybrid system through both continuous and discrete dynamics as well as their interactions. Fault parameterization in hybrid systems at an appropriate level of abstraction is a challenging problem. Discrete faults lead to additional modes and increased computational cost, while continuous faults cannot be estimated efficiently. Here, we present a framework for fault parameterization based on hybrid automata models and we parameterize both abrupt failures and subtle degradation of components. Although, currently, we focus on hybrid automata with linear first-order dynamics, our printer example demonstrates that this class of hybrid systems can address realistic and important problems. We use the developed model to generate the fault symptom table for different fault hypotheses. For the rest of the paper, we assume single persistent faults. Note that while this is a simplifying assumption required by our on-line diagnostic system, the hybrid system model can also describe multiple simultameous faults. The fault symptom table is generated off-line by simulation and is compiled into a decision tree that is used as the on-line diagnoser.

Monitoring of hybrid systems has two components, discrete mode estimation and continuous-state tracking. Once a system is estimated to be in a particular mode, a continuous state estimator such as Kalman filter could be used to track the continuous state. This paper also addresses the problem of mode estimation. In our on-line diagnostic system, we simulate only the temporal discrete-event behavior of the hybrid system using a timed Petri net model while abstracting away the continuous dynamics. The architecture of the diagnostic system is shown in Fig. 1. Discrete-event data from built-in sensors and control commands of the printer are used to drive the model of the system. The model compares observed sensor events with their expected values. When a fault occurs, the deviation from the simulated behavior triggers the decision-tree diagnoser. The diagnoser either waits for the next sensor event or queries the mode estimator to search for a particular event, depending on the next test. The mode estimator requests a temporal prior from the model of the system, uses the prior to retrieve the segment of the signal from appropriate sensors, and computes the posterior of the event. The system model uses the event posterior to update model parameters, generate a deviation of the event parameter for the diagnoser, and the process iterates until there are no more sensor tests to perform and the diagnoser reports the current fault candidates.

Monitoring and diagnosis of hybrid systems is a challenging problem for designing real-time embedded system and has recently attracted considerable research efforts. Proposed models and techniques include Bayesian networks [6], timed discrete-event representations [7], particle filter methods for tracking system behavior [8, 9], Viterbi-like algorithms [2], and temporal causal graphs [10]. Our approach presents a fault parameterization that can model abrupt failure and subtle degradation of components

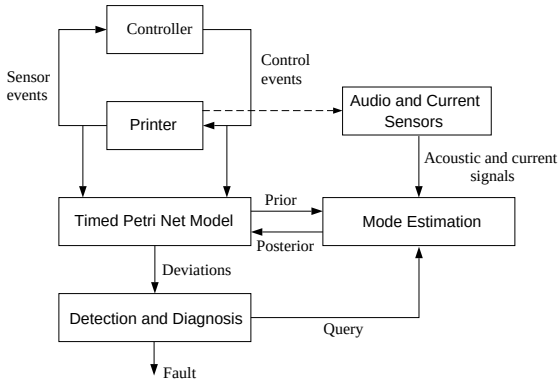

Figure 1: Architecture of the prototype diagnosis system.

by extending the well-defined hybrid automata modeling framework. Furthermore, existing approaches to hybrid system monitoring and diagnosis do not address the computational data association problem associated with distributed multi-sensor systems and assume that the sensor output has already been properly assembled to form likelihood functions of the system output. Moreover, they assume either no autonomous mode transition or autonomous transition without signal mixing. In contrast, our approach exploits model knowledge of control and discrete-event behaviors of hybrid systems to address the exponential blow-up in data association of multi-sensor observation, as well as the complexity due to multiple measurements over time.

The paper is organized as follows. Section 2 describes the paper feed system of the DC265 Xerox printer that is used to illustrate our approach. The fault parameterization for hybrid systems is described in Section 3. Section 4 presents our approach for monitoring and diagnosis of hybrid systems. Our prototype diagnostic system and experimental results are present in Section 5.

2 Motivating Example

The paper feed system shown in Fig. 2 is used to move sheets of paper from the tray to the xerographic module of the printer by orchestrating several electro-mechanical components. These components include the feed and elevator motors, the acquisition solenoid, the feed and the acquisition rolls, and the wait station and stack height sensors as shown in Fig. 2. The feed motor is a 24V DC motor that drives the feed and acquisition rolls. The acquisition solenoid is used to initiate the feeding of the paper by lowering the acquisition roll onto the top of the paper stack. The elevator motor is used to regulate the stack height at an appropriate level. The wait station sensor detects arrival of the leading or trailing edge of the paper at a fixed point of the paper path. The stack height sensor is used to detect the position of the paper stack and controls the operation of elevator motor. The paper feed system exemplifies important challenges similar to those that arise in the monitoring and diagnosis of more

complex systems such as the print shop.

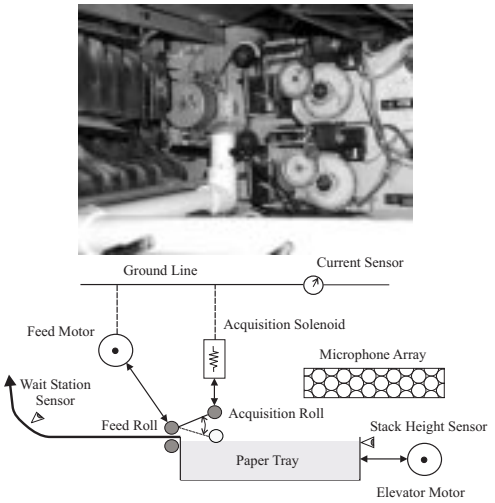

Figure 2: Paper feed system of the Xerox DC265 printer.

In our experimental setup, in addition to the system built-in sensors, audio and current sensors are deployed for estimating quantities not directly accessible. These *virtual sensors* [12] augment the diagnostic information using signature analysis techniques A 14-microphone array is placed next to the printer. These are omni-directional microphones with a flat frequency response up to 20KHz. The total ground return current of the feed system is acquired using a 0.22Ω resistor in series between the driveplate's return path and the analog ground. Sensor signals are acquired at 40,000 samples per second and 16-bit resolution by a 32-channel data acquisition system. The controller and the various components of the printer communicate with each other by sending control and sensor signals through a common bus. By using an interface card on a PC, these control and sensor signals can be accurately detected and mapped to the analog data acquired by the data acquisition system. The interface card is also used to systematically activate individual components while the rest of the printer was powered off in order to build signal templates for each component as required by our monitoring and diagnosis approach.

3 Modeling Faults in Hybrid Systems

In this section, we first review the hybrid automaton model [1] and then, introduce a fault modeling formalism for hybrid systems that is used for our monitoring and diagnosis tasks.

Definition 1: A hybrid system is described by $H = (Q, X, \Sigma, I, Inv, E, f)$ where Q is the set of discrete states or *modes* of the system, $X = \Re^n$ is the continuous state space, Σ is a finite set of transition labels or *events*, $I \subseteq Q \times X$ is the set of initial conditions, $Inv : Q \rightarrow 2^X$ is the invariant associated with each mode q, $E \subset Q \times X \times \Sigma \times Q \times X$ is the set of discrete transitions, and $f : \Re \times Q \times X \rightarrow X$ is the flow condition for every mode which is usually represented by a differential equation.

The state of the hybrid system is described by the pair (q, x). The state can change either by a discrete and instantaneous transition or by a time delay. A discrete (or mode) transition changes both the mode and the continuous state, while a time delay changes only the continuous state according to the flow condition. Mode transitions are induced by either control events or the evolution of the continuous dynamics using the guard conditions. For example, the transition from idle to ramp-up for a motor in the printer is caused by the control event "turn_motor_on" issued by the printer controller. However, the transition that represents the acquisition roll contacting the paper depends on the dynamic evolution of the system and is characterized as autonomous. Complex systems can be modeled by using the parallel composition of simple hybrid automata that represent the various components.

In the following, we introduce three types of system (or component) faults in the hybrid automata that model the components of the system. First, at every mode q we assume that the continuous dynamics are described by the parameterized system $\dot{x}(t) = f_q(x(t), \theta_q(t))$ where the system's behavior depends on the fault parameter $\theta_q \in \Theta_q$ and θ_{q0} represents the faultless system. Therefore, we use a finite set of subspaces Θ_q representing the different and possible multiple fault hypotheses $\theta_q \in \Theta_q, q \in Q$ to be tested. The set of fault parameters is partitioned as $\Theta_1 = \bigcup_{q \in Q} \Theta_q$. It should be noted that in the general case the parameterized dynamic system can be described by $\dot{x} = f_q(t, x, \theta, d, u)$ to model time-varying dynamics with continuous inputs u and disturbances d.

Second, we introduce discrete states corresponding to faulty modes of the system that cannot be described by small deviations in the fault parameters θ. This modeling assumption arises naturally by the need to represent abrupt component failures caused possibly by exogenous actions. These abrupt failures are modeled as unobservable events that drive the system to the faulty modes. Here, we assume that the set of modes of the hybrid system is partitioned as $Q = Q_N \cup Q_F$ where Q_N and Q_F are the set of normal modes and faulty modes respectively. Similarly, we partition the set of transition labels as $\Sigma = \Sigma_N \cup \Sigma_F$. The set of failure events Σ_F labels transitions to faulty modes. Note that if information about the continuous dynamics for the faulty modes is available then a flow condition can be associated with these modes.

Finally, we introduce possible faults in the guards of the mode transitions in order to model the case when system faults affect autonomous transitions. For example, the motor switches from the ramping up phase to the steady state phase when the angular velocity reaches its steady state value. The steady state value is usually a reference signal used in a lo-

cal PID controller. However, increased friction in the motor caused by aging may slow down the motor. Faults in the autonomous transitions are represented by considering a parameterized guard condition of the form $G(x, \theta_e) \subset X \times \Theta_e$ where $\theta_e \in \Theta_e$ is the fault parameter and θ_{e0} describes the faultless system. Therefore, we use a finite set of subspaces Θ_e representing the different and possible multiple fault hypotheses $\theta_e \in \Theta_e, e \in E$ to be tested so that the set of fault parameters is partitioned as $\Theta_2 = \bigcup_{e \in E} \Theta_e$.

Let ϵ denote the null event and $\Sigma' = \Sigma_F \cup \{\epsilon\}$. Then, the space of fault hypotheses for the hybrid system is defined as $H = \Sigma' \times \Theta_1 \times \Theta_2$ where the null event ϵ corresponds to the case when no discrete fault has occurred. Fault hypotheses for the hybrid system are described by the signal $h(t)$ where $h : \Re \rightarrow H$.

Complex systems like the DC265 printer consist of multiple potentially malfunctioning components. Consider the set of components $COMPS = \{c_1, \ldots, c_m\}$ and assume without loss of generality that each component can be modeled by a hybrid automaton. Then, the hybrid model of the printer is computed using the parallel composition of the simple hybrid automata [1]. In this case, the set of modes Q can be understood as the product of individual component modes, i.e. $q = [q^{(1)}, \ldots, q^{(m)}]^T \in Q$. Similarly, for the continuous state we have $x = [x^{(1)}, \ldots, x^{(m)}]^T \in X$. The set of events of the hybrid system can be written as $\Sigma = \bigcup_{i=1}^{m} \Sigma^{(i)}$. The set of multiple fault hypotheses can be partitioned with respect to the components of the system as $H = H^{(1)} \times \ldots \times H^{(m)}$ where $H^{(i)} = \Sigma'^{(i)} \times \Theta_1^{(i)} \times \Theta_2^{(i)}$.

Example We present a simplified hybrid model of the paper feed system. We consider only the feed motor, the acquisition solenoid, and a sheet of paper. Note that for our diagnostic system in Section 5 we also model the elevator motor that is used to place the paper stack at the correct position during the printing operation.

Reliability studies have shown that the most common faults for the feed motor are the following: the motor does not energize, the nominal speed is not reached, and it takes longer to ramp up. The feed motor is a DC brushless motor locally controlled by a PID controller. In our experiments, an external optical sensor was instrumented to measure the angular velocity of the motor to obtain "ground-truth". Note that this sensor is not used in our diagnosis approach. The measurements obtained by this optical sensor show clearly that the behavior of the motor and the local PID controller can be approximated by an integrator system with three distinct modes, ramp-up, steady-state, and ramping down. The behavior of interest for the feed motor is captured in the hybrid automaton shown in Fig. 3.

Initially, the feed motor is idle and the angular velocity is $\omega = 0$. Upon receiving the control command "motor_on", the feed motor is ramping up according to the equation $\dot{\omega} = K_{ru} + \theta_{ru}$. The nominal behavior for the motor of the paper feed system is described by $K_{ru} = 550$rad/sec^2 and $\theta_{ru} = 0$, where $\theta_{ru} \in \Theta_{ru} \subset \Re$ parameterizes the accelaration, and therefore the "ramping-up" time of the motor. The range Θ_{ru} is determined using the specifications of the motor so as to simulate the fault of interest. The transition from the ramp-up to the steady-state mode is labeled by the guard $\omega = \omega_{ss} + \theta_{ss}$. The nominal steady state speed of the motor is $\omega_{ss} = 16.5$rad/sec (and $\theta_{ss} = 0$). Upon receiving a "motor_off" control command, the motor is ramping-down ($\dot{\omega} = K_{rd}, K_{rd} < 0$) and returns to the idle position. Note that it is possible that the controller will issue a control event "motor_on" before the motor has completely stopped. Finally, the case when the motor does not energize is modeled by the fault mode q_4.

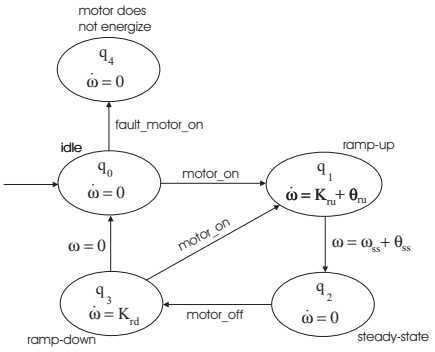

Figure 3: Hybrid model for the feed motor.

The most common faults for the acquisition solenoid are the following: solenoid does not energize, and solenoid energizes slowly. The hybrid automaton model shown in Fig. 4 captures the behavior and the possible faults for the acquisition solenoid. This model describes the behavior of the solenoid using the relative displacement y of the acquisition roll that is attached to the solenoid. Let $y = 0$ be the initial condition for the displacement of the acquisition roll. After receiving a "solenoid_on" event from the controller the solenoid energizes to drop the acquisition roll onto the paper. The dynamics of the system in the pull-in mode of the solenoid are approximated by the equation $\dot{y} = K_{pi} + \theta_{pi}$. For the nominal behavior, we have $K_{pi} = -0.88235m/sec$ and $\theta_{pi} = 0$. The acquisition roll contacts the paper at $y = K_h + \theta_h$ with $K_h = -15mm$ and $\theta_h = 0$ for the nominal behavior. The parameter θ_h describes the deviation for the height of the paper stack which is control by the elevator motor. A "solenoid_off" event deenergizes the solenoid to lift the acquisition roll. The faulty mode q_4 is introduced to model the failure when the solenoid does not energize.

A set of gears, belts, and clutches is used to transfer the drive from the feed motor to the feed and acquisition rolls that drive the paper. The motion of a sheet of paper in the paper path of the printer is described by the hybrid system shown in Fig. 5 where the continuous state x_{le} represents the speed of the leading edge of the paper. The modes for the paper motion correspond to the paper being stationary,

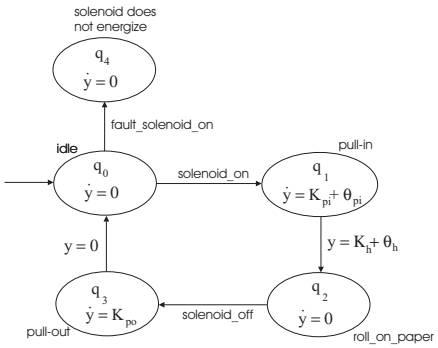

Figure 4: Hybrid model for the acquisition solenoid.

and the paper being driven by the acquisition roll or the feed roll. When the acquisition roll contacts the paper stack ($y = K_h + \theta_h$), the top sheet starts moving towards the feed roll. As soon as the leading edge of the paper reaches the nip created by the feed roll ($x_{le} = r_f$), the acquisition roll is lifted and the paper is driven by the feed roll. In the case when the paper is driven by the feed roll, we consider a simplified model for the paper motion described by $\dot{x}_{le} = \theta_1\theta_2 R_{fr}\omega$. The parameter θ_1 models the drive transfer from the feed motor to the feed roll through a set of belts, gears, and clutches and has nominal value $\theta_{1_0} = 1$. A common failure for the system is the degradation of the gears which affects the speed of the moving sheet and may result in paper jams. Such a degradation is represented in our framework by $\theta_1 < 1$. The parameter θ_2 represents the friction between the feed roll and the paper with nominal parameter $\theta_{2_0} = 1$. A roll that is worn will cause the paper to slip and may also lead to paper jams. Finally, R_{fr} is a constant that depends on the geometrical characteristics of the belt, the gears, and the rolls. Similarly, for the case when the paper is driven by the acquisition roll we have $\dot{x}_{le} = \theta_1\theta_3\theta_4 R_{ar}\omega$. Note that the acquisition roll is driven by the feed motor through the feed roll. Here, θ_3 represents the drive transfer from the feed roll to the acquisition roll, and θ_4 the friction between the acquistion roll and the paper. When the leading edge of the paper reaches the wait station sensor ($x_{le} = s_1$) the feed motor is turned off and the paper stops.

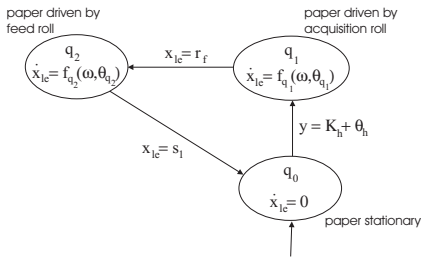

Figure 5: Hybrid model describing the paper motion.

The hybrid model of the paper feed system is derived using the parallel composition of the hybrid automata that model the motor, the solenoid, and the paper motion. The mode q of the overall system is the product of the component modes and the continuous state is $x = [\omega, y, x_{le}]^T$. The space of fault hypotheses for the paper feed system is the product of the fault hypotheses for the components. Several fault hypotheses $h(t)$ for the paper feed system have been simulated using the hybrid system model described above.

4 Design of the Fault Diagnoser

In this section, we present the design method for the fault diagnoser. One of the advantages of our hybrid model is that it can be used to automatically generate the fault symptom table which in turn, is compiled to a decision-tree that can be used efficiently as the on-line diagnoser.

4.1 Generation of the fault symptom table

The problem of hybrid system diagnosis is to find the most likely fault hypothesis $h(t)$ for the observation history $\mathbf{y}(t)$. This paper presents an approach for monitoring and diagnosis of hybrid systems based on a qualitative representation of the fault hypotheses. The abrupt fault events are represented by the binary values "Y" and "N" (Yes, No) and the fault parameters θ are labeled as normal (0), above normal $(+)$, below normal $(-)$, maximum value $(\max)$, and minimum value $(\min)$. The sensor variables y_k are also discretized and are represented approprietaly either by qualitative values or binary values. The qualitative values were selected so as to be able to distinguish among the frequent faults described by our reliability studies. The $(+)$ and maximum values are used to distinguish, for example, between the paper arriving late at the sensor and no paper at the sensor respectively. In the case when the continuous dynamics of the system are described by first-order integrators as in the paper feed system, a partition of the hypotheses space using thresholds can be used to generated a fault symptom table where the qualitative sensor values depend deterministically in the qualitative fault hypotheses. For more complex dynamics, the partition of the fault hypothesis space can be determined based on the continuous dynamics using methods like those used in the supervisory control of hybrid systems [5]. However, in this case it is possible that the qualitative sensor values will depend on the qualitative fault hypotheses in a nondeterministic manner and the fault symptom table would contain multiple rows for the same fault. Diagnostic inference for such cases may be still valuable and it is the subject of future research.

Our diagnostic process consists of two steps. In the first step, a fault symptom table is generated offline by simulation of the hybrid system model. In the second step, a decision tree is compiled from the fault symptom table and it is used as the on-line diagnoser. The behavior of the system is monitored in order to detect deviations from the nominal behavior predicted by the model. Upon detection of abnormal behavior, the decision tree generates qualitative candidate models for the fault hypotheses. The diagnostic task is to determine the most likely path in the decision tree by taking into consid-

eration current and future measurements. It should be noted that for diagnosis of many physical systems, qualitative estimates for the fault parameters are sufficient. For example, diagnosing a slow motor in the printer may allow a simple adjustment in the controller to prevent paper jams.

Example The fault symptom table for the paper feed system is shown in Table 1. The columns of the table correspond to the deviations of the sensor outputs from the nominal values. The diagnosability (discrimination between the faults) of the approach can be assessed using existing methods based on fault symptom tables [3, 13]. In the fault symptom table of Table 1, the selected sensor outputs are the following: y_1 is the time the leading edge of the paper is detected by the wait station sensor and in the hybrid model of the system is associated with the firing time of the transition labeled by $x_{le} = s_1$ in Fig. 5, y_2 is the pull-in time of the acquisition solenoid that is associated with the firing time of the transition labeled by $y = h + \theta_h$ in Fig. 4, y_3 takes the value "Y" if the elevator motor energizes and "N" otherwise, y_4 is the speed of the elevator motor (the hybrid model of the elevator motor is similar to the hybrid automaton shown in Fig. 3 and it is not included due to space limitations), y_5 takes the value "Y" if the feed motor energizes and "N" otherwise, y_6 is the ramp-up time of the feed motor and is associated with the firing time of the transition labeled by $\omega = \omega_{ss} + \theta_{ss}$ in Fig. 3, and y_7 is the angular velocity ω of the feed motor. It should be noted that in the on-line diagnostic system described in Section 5 faults that affect the arrival of the trailing edge of the paper in the wait station sensor are taken into consideration; the corresponding part of the fault symptom table is omitted due to space limitations.

In order to illustrate the generation of the fault symptom table consider the second row of Table 1 corresponding to the case when the feed motor has high ramp-up time. To simulate the fault, we set the parameter $\theta_{ru} = -150 rad/sec^2$ and we monitor the state of the hybrid system. The values in the fault symptom table represent the deviations of the sensor outputs from the nominal values. For example, the ramp-up time of the motor in the nominal operation is approximately $30ms$ while for the simulated fault is $41ms$ and the qualitative deviation for the sensor output y_6 is $(+)$.

4.2 Decision-tree diagnoser

For real-time, embedded applications, the fault symptom table can be compactly represented by a corresponding decision tree using, for example, the ID3 algorithm [11]. In our diagnosis system we have two types of sensors, built-in sensors that are always accessible with a low cost and virtual sensors that cannot be used directly in the diagnoser but require the invocation of the mode estimation algorithm (see Section 5). Thus, the built-in sensors can be used for fault detection and trigger the diagnosis algorithm. The diagnoser will try to isolate the fault using only the built-in sensors. If this is not possible, then it will use virtual sensors. In order to take into consideration the sensor characteristics, we associate with the built-in sensors a cost equal to 0 and with the virtual sensors a cost equal to $K > 0$. The objective of the decision tree generation algorithm is to minimize the weighted cost of the tree $\sum_{L \in \text{leaves}} P(L) \sum_{X \in path(L)} C(X)$, where $P(L)$ is the probability of a fault or faults corresponding to leaf L of the tree and $C(X)$ is the cost of sensor test at node X of the path to L. A decision tree minimizing the weighted cost is generated by applying the ID3 algorithm in two phases. First, ID3 builds a tree using only the built-in sensors. Next, ID3 is applied to leaf nodes of the tree with more than one faults, and generates subtrees for those leaves using the virtual sensors (see Fig. 6).

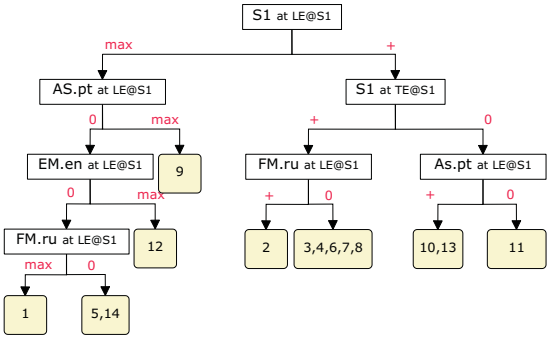

Figure 6: Decision tree for the paper feed system.

5 On-line Diagnostic System

In this section, we describe our on-line diagnostic system and we illustrate our approach using experimental results for one fault scenario. We have prototyped a diagnosis system comprising three main components: system model, mode estimator, and decision-tree diagnoser (Fig. 1).

5.1 On-line temporal discrete-event simulation

In order to satisfy the real-time requirement for our diagnostic system, we selected to abstract away the continuous dynamics of the hybrid model and simulate on-line only the event-driven dynamics. For the simulation of the temporal discrete-event behavior of the paper feed system, it is necessary to include in the model the control logic of the system. This is the case in most embedded systems where a high-level controller orchestrates various components in order to perform a complex operation. In this paper, we use a timed Petri net model of the supervisory controller of the DC265 focusing only on the paper feed system. Petri nets have been used extensively as a high level control specification model [4] and can adequately describe the control logic of the printer for the purpose of this paper. We use a timed Petri net to model temporal discrete-event behavior of the supervisory controller instead of timed automata for the following reasons. First, Petri nets offer significant computational advantages over concurrent finite automata when the system to be modeled contains multiple moving objects. For example, it is desirable for the printer model to compactly describe

Id	Failure	Fault parameter	y_1	y_2	y_3	y_4	y_5	y_6	y_7
1	Feed motor does not energize	$\sigma_{f_{motor}}$	MAX	0	Y	0	N	MAX	MIN
2	Feed motor has high ramp up time	$\theta_{ru} = (-)$	+	0	Y	0	Y	+	0
3	Feed motor is slow	$\theta_{ss} = (-)$	+	0	Y	0	Y	0	-
4	Clutch has broken gears	$\theta_1 = (-)$	+	0	Y	0	Y	0	0
5	Belt is broken	$\theta_1 = (-)$	MAX	0	Y	0	Y	0	0
6	Belt is worn	$\theta_1 = (-)$	+	0	Y	0	Y	0	0
7	Gears are worn	$\theta_1 = (-)$	+	0	Y	0	Y	0	0
8	Feed roll is slipping	$\theta_2 = (-)$	+	0	Y	0	Y	0	0
9	Acquisition solenoid does not energize	$\sigma_{f_{solenoid}}$	MAX	MAX	Y	0	Y	0	0
10	Acquisition solenoid energizes slowly	$\theta_{pi} = (-)$	+	+	Y	0	Y	0	0
11	Acquisition roll is worn and slips	$\theta_4 := (-)$	+	0	Y	0	Y	0	0
12	Elevator motor does not energize	$\sigma_{f_{elev}}$	MAX	0	N	0	Y	0	0
13	Elevator motor is slow	$\theta_{el} = (-)$	+	0	Y	-	Y	0	0
14	No paper	$\sigma_{f_{paper}}$	MAX	0	Y	0	Y	0	0

Table 1: Faults for the paper feed system

a variable number of multiple sheets of paper in a printing operation. Second, Petri nets can be used to model concurrency and synchronization in distributed systems very efficiently without incurring state-space explosion.

The dynamics of a Petri net is characterized by the evolution of a marking vector referred to as the state of the net. The marking is updated upon firing of transitions. In a timed Petri net, transition firings can be expressed as functions of time. In addition, firing some of the transitions can be synchronized with external events. In this case, a transition is associated with an external event that corresponds to a change in state of the system. The firing of the transition will occur when the associated event occurs and the transition has been enabled. We associate with each transition a firing time domain $[\tau_{\min}, \tau_{\max}]$. The transition is enabled when all its input places are marked, but the firing of the transition occurs at a specific time instant within the time domain. The advantage of this formalism is that it takes into consideration stochastic fluctuations in the time duration of physical activities in the system. If statistical information for the firings of the transition is provided, then the firing time domain can be augmented with a probability distribution characterizing the time instant the transition fires after it has been enabled. In our diagnostic system, it is assumed that a normal distribution is associated with the firing time domain of each autonomous transition. The timed Petri net model of the supervisory controller is used to generate temporal prior probability distributions for the occurrence of autonomous events so as to focus the signal processing algorithms when and where to look for signatures of interest.

Example The Petri net of Fig. 7 models the control logic of the paper feed system and can capture concurrent behavior for multiple sheets and multiple components in an efficient manner. Control commands issued by the controller and outputs of built-in sensors are output and input events respectively for the Petri net. For example, the transition labeled by "Ac_sl_on" corresponds to the event "acquisition solenoid on". The transition labeled by "Dr_ac_rl" corresponds to the autonomous event "drop acquisition roll" that for the normal operation of the system should occur within a specified time interval $[\tau_{\min}, \tau_{\max}]$ from the time t was enabled and additionally, the occurrence is descibed by a normal distribution. The transition labeled by "LE@S1" corresponds to the event the wait station sensor detects the leading edge of the paper.

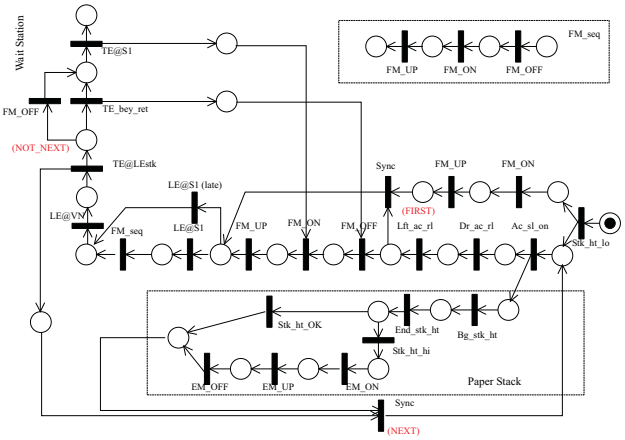

Figure 7: Petri net model of the paper feed system.

5.2 An Online Mode Estimation Algorithm

Current model-based diagnosis techniques for hybrid systems are based on observers that measure the continuous state [9] $x(t)$ (the quantized state in [7]) at every time t. In a multi-sensor environment like that described in our motivating example, the additional problem of data association must be addressed. Consider the experimental test-bed of the Xerox DC265 described in Section 2 and assume there exist l sensors. The sensor output vector is $\mathbf{y} = [y_1, \ldots, y_l]^T \in Y$, where y_k is the output of sensor k. Each y_k could be a mea-

surement of a signal from the i^{th} component at mode $q_j^{(i)}$ or a composite signal of multiple components at mode $\mathbf{q} = [q^{(1)}, \ldots, q^{(m)}]$. In order to monitor the mode transitions of the hybrid system, the sensor output must be associated not only with the i^{th} component but also with the mode $q_j^{(i)}$ of the component.

First, consider the case when there is no signal mixing and each sensor y_k measures a signal $s_i \in S$ from system component i only. The number of possible associations of y_k's with $q_j^{(i)}$'s is $(\sum_i |Q^{(i)}|)^l$ where $|Q^{(i)}|$ is the number modes for the i^{th} component, that is, is exponential in the number of sensors at every time step. In the more general case, each sensor signal y_k measures a composite of signals $s_i, i \in I$ through a mixing function. Without prior knowledge about the mixing function, any combination of component signals could be present in the sensor signals y_k and the total number of data associations is exponential in the number of both the sensors and signal sources. For applications such as diagnosis, it is usually necessary to reason across multiple time steps and examine the history of mode transitions in order to identify a component fault occurred in an earlier mode. Each pairing of the observations with the mode vector in the single-step mode estimation creates a hypothesis of the system mode transition sequence. As more observations are made over time, the total number of possible mode transition sequences is exponential in the numbers of sensors *and* measurements over time.

The objective of mode estimation is to estimate the mode transition sequence of a hybrid system: $\mathbf{q}_0 \stackrel{\tau_1}{\rightarrow} \mathbf{q}_1 \stackrel{\tau_2}{\rightarrow} \ldots \stackrel{\tau_n}{\rightarrow} \mathbf{q}_n$. Each transition is caused by one or more mode transitions of components of $\mathbf{q}$. Assuming each sensor output y_i is a linear superposition[1] of s_j's

$$y_i(t) = \sum_{j=1}^{n} \alpha_{ij} s_j(t - \tau_{ij}), \;\; i = 1, \ldots, l \qquad (1)$$

or more compactly, $\mathbf{y}^t = D(\alpha_{ij}, \tau_{ij}) * \mathbf{s}^t$ where $D(\alpha_{ij}, \tau_{ij})$ is an $l \times n$ mixing matrix with elements $d_{ij} = \alpha_{ij}\delta(t - \tau_{ij})$ and $\delta(t - \tau_{ij})$ is the sampling function. The operator $*$ denotes element-wise convolution in the same way matrix-vector multiplication is performed. In particular, when s_j represents the signal event characteristic of a mode transition of the j^{th} component, the mode estimation problem is then to determine τ_{ij}, the onset of the signal event s_j, and α_{ij}, the contribution of s_j to the composite sensor output y_i. A common physical interpretation for the mixing parameters τ and α is that τ characterizes signal arrival time at each sensor, and α sensor gain for each sensor.

The mode estimation algorithm computes $P(D(\alpha, \tau)|\mathbf{y}^t)$, the posterior distribution of τ and α given observation $\mathbf{y}^t$, iterating through the following three steps: (1) Use a model of system behaviors to generate a temporal prior $P(D(\alpha, \tau))$ of transition events within the time window associated with the current time step; (2) Decompose sensor observation as a sum of component signal events $\mathbf{y}^t = D(\alpha, \tau) * \mathbf{s}^t$, and compute the likelihood function $P(\mathbf{y}^t|D(\alpha, \tau))$; (3) Compute the posterior distribution of the mode transition $P(D(\alpha, \tau)|\mathbf{y}^t)$ using Bayesian estimation and update the mode vector. For simplicity, the likelihood functions are assumed to be Gaussian. Details for the mode estimation algorithm can be found in [14].

[1] When the signals are nonlinearly mixed, then a nonlinear source separation method must be used.

5.3 Experimental Results

The diagnosis system of Fig. 1 has been demonstrated on four test fault scenarios. The system, implemented in MATLAB running on a Win2000 PC, sequentially scans prerecorded data streams at real-time data rates to emulate on-line monitoring. The four test cases involve a feed roll worn fault (labeled as "8" in the decision tree of Fig. 6)), a feeder motor belt broken fault ("5"), an acquisition roll worn fault ("11"), and a motor slow ramp-up fault ("2"), and cover an interesting subset of system-level faults of the printer. These faults may cause a delayed paper or no paper at subsequent sensors. Note the two "worn" cases are not directly observable. Our algorithm isolates the faults by reasoning across several sensor tests to rule out competing hypotheses using the decision tree. The motor slow ramp-up fault could be directly observed by the corresponding virtual sensor test only at the cost of substantial signature analysis. Instead, our algorithm uses less expensive system built-in sensors to monitor and detect faults and only invokes virtual sensor tests on a when-needed basis.

Let's examine the trace of the diagnosis output for one of the fault scenarios. The paper arrives late at wait station sensor LE@S1. The arrival time is compared with the expected time to generate a qualitative deviation "+", which triggers the diagnosis. Since the paper arrived at the sensor, hypotheses such as belt broken are ruled out. Reading off the decision tree, the next test TE@S1, trailing edge arrival time, is then invoked and returns normal ("0"). This rules out feed roll worn and motor slow ramp-up faults since both would cause the trailing edge to be late. Next on the decision tree, the more expensive acquisition solenoid pull-in time test (AS.pt) is invoked. This calls the mode estimation algorithm to determine the transition time at which the acquisition roll contacts the paper (or equivalently, solenoid pull-in), an autonomous transition event. The composite signal of one-page printing is shown in Fig. 8. The estimation uses acoustic and current signal templates of solenoid and motor to compute a posterior probability distribution of the pull-in event. Using the Petri net model prediction [495ms,505ms] to localize the event search, the estimation algorithm determines that the event is 2.5 ms later than the nominal value, well within the permissible range (see the peak location of posterior in Fig. 9). Therefore, AS.pt returns "0", and the only candidate remaining is the acquisition roll worn fault, which is the correct diagnosis. Physically, the reduced friction between the worn acquisition roll and paper causes the

leading edge of the paper late at LE@S1. But this does not affect the trailing edge arrival time since the paper stops momentarily when the sensor detects the leading edge, and moves again without using the acquisition roll. In contrast a worn feed roll would cause the trailing edge to be late.

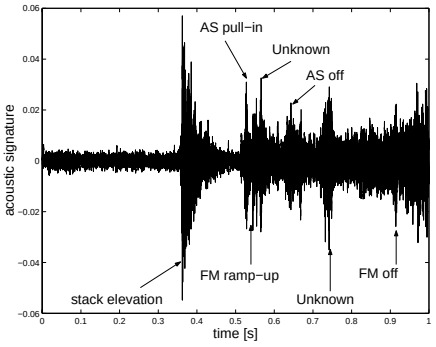

Figure 8: Acoustic signal for a one-page printing operation of DC265 printer.

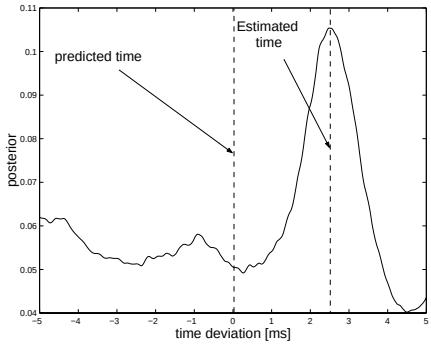

Figure 9: Posterior distribution of AS_pull_in time.

6 Conclusions

In this paper, we present a framework for fault parameterization based on hybrid automata models and we parameterize both abrupt failures and subtle degradation of components. We describe our on-line diagnostic system for the Xerox DC265 printer and we illustrate our approach using experimental results. In addition, this work has demonstrated that knowledge of the temporal discrete-event behavior of the system can address the computational problem of data association in sensor-rich hybrid systems. Currently, we research methods that combine qualitative and quantitative techniques in order to investigate the applicability of our approach to additional classes of faults.

Acknowledgment This work is supported in part by the Defense Advanced Research Projects Agency (DARPA) under contract number F33615-99-C3611. We thank Claudia Picardi for her assistance in implementing the decision tree algorithms during an internship at Xerox PARC and Jim Kurien for comments on drafts of the paper.

References

[1] R. Alur, C. Courcoubetis, N. Halbwachs, T. Henzinger, P.-H. Ho, X. Nicollin, A. Oliveiro, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical and Computer Science*, 138:3–34, 1995.

[2] M. Basseville, A. Benveniste, and L. Tromp. Diagnosing hybrid dynamical systems: Fault graphs, statistical residuals and Viterbi algorithms. In *Proc. of the 37th IEEE CDC*, pp 3757–3762, 1998.

[3] L. Console, C. Picardi, and M. Ribando. Diagnosis and diagnosability analysis using process algebra. In *Proc. 11th Int. Workshop on Principles of Diagnosis (DX'2000)*, 2000.

[4] A. Desrochers and P. Al-Jaar. *Applications of Petri Nets in Manufacturing Systems.* IEEE Press, 1995.

[5] X. Koutsoukos, P. Antsaklis, J. Stiver, and M. Lemmon. Supervisory control of hybrid systems. *Proc. of IEEE*, 88(7):1026–1049, July 2000.

[6] U. Lerner, R. Parr, D. Koller, and G. Biswas. Bayesian fault detection and diagnosis in dynamic systems. In *Proc. AAAI'2000*, 2000.

[7] J. Lunze. Diagnosis of quantised systems by means of timed discrete-event representations. In N. Lynch and B. Krogh, editors, *Hybrid Systems: Computation and Control*, vol. 1790 of *LNSC*, pp. 258–271. Springer, 2000.

[8] S. McIlraith. Diagnosing hybrid systems: a Bayesian model selection problem. In *Proc. 11th Int. Workshop on Principles of Diagnosis (DX'2000)*, 2000.

[9] S. McIlraith, G. Biswas, D. Clancy, and V. Gupta. Hybrid systems diagnosis. In N. Lynch and B. Krogh, editors, *Hybrid Systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, pages 282–295. Springer, 2000.

[10] S. Narasimhan, F. Zhao, G. Biswas, and E. Hung. Fault isolation in hybrid systems combining model based diagnosis and signal processing. In *Proc. 4th IFAC Symp. SAFEPROCESS*, pages 1074–1079, 2000.

[11] J. Quinlan. Combining instance-based and model-based learning. In *Proc. 10th Int. Conf. on Machine Learning*, 1993.

[12] M. Sampath, A. Godambe, E. Jackson, and E. Mallow. Combining qualitative & quantitative reasoning - a hybrid approach to failure diagnosis of industrial systems. In *Proc. 4th IFAC Symp. SAFEPROCESS*, pages 494–501, 2000.

[13] L. Trave-Massuyes, T. Escobet, and R. Milne. Model-based diagnosability and sensor placement application to a frame 6 gas turbine subsystem. In *Proc. IJCAI'2001*, pp. 551–556, Seattle, WA, 2001.

[14] F. Zhao, X. Koutsoukos, H. Haussecker, J. Reich, P. Cheung, and C. Picardi. Distributed monitoring of hybrid systems: A model-directed approach. In *Proc. IJCAI'2001*, pp. 557–564, Seattle, WA, 2001.