A Comprehensive Diagnosis Methodology for Complex Hybrid Systems: A Case Study on Spacecraft Power Distribution Systems

Matthew J. Daigle, *Member, IEEE*, Indranil Roychoudhury, *Member, IEEE*, Gautam Biswas, *Senior Member, IEEE*, Xenofon D. Koutsoukos, *Senior Member, IEEE*, Ann Patterson-Hine, *Senior Member, IEEE*, and Scott Poll

***Abstract*—The application of model-based diagnosis schemes to real systems introduces many significant challenges, such as building accurate system models for heterogeneous systems with complex behaviors, dealing with noisy measurements and disturbances, and producing valuable results in a timely manner with limited information and computational resources. The Advanced Diagnostics and Prognostics Testbed (ADAPT), which was deployed at the NASA Ames Research Center, is a representative spacecraft electrical power distribution system that embodies a number of these challenges. ADAPT contains a large number of interconnected components, and a set of circuit breakers and relays that enable a number of distinct power distribution configurations. The system includes electrical dc and ac loads, mechanical subsystems (such as motors), and fluid systems (such as pumps). The system components are susceptible to different types of faults, i.e., unexpected changes in parameter values, discrete faults in switching elements, and sensor faults. This paper presents Hybrid TRANSCEND, which is a comprehensive model-based diagnosis scheme to address these challenges. The scheme uses the hybrid bond graph modeling language to systematically develop computational models and algorithms for hybrid state estimation, robust fault detection, and efficient fault isolation. The computational methods are implemented as a suite of software tools that enable diagnostic analysis and testing through simulation, diagnosability studies, and deployment on the experimental testbed. Simulation and experimental results demonstrate the effectiveness of the methodology.**

***Index Terms*—Distributed diagnosis, electrical power distribution systems, hybrid bond graphs (HBGs), hybrid systems, model-based diagnosis.**

I. INTRODUCTION

THE INCREASING complexity of modern engineering systems has necessitated the deployment of online health monitoring and diagnosis schemes to ensure their safe, reliable, and efficient operation. Model-based diagnosis schemes are the preferred approach, because they allow for more general and robust diagnosis solutions [1]–[6]. However, deployment of these schemes on real systems presents significant challenges in *model development*, *system monitoring*, and *fault isolation*.

Model-based diagnosis requires accurate and reliable models of real physical processes that encompass multiple domains (e.g., hydraulic, electrical, and mechanical). Behaviors can be nonlinear, and the interactions between components and between the system and the environment can be difficult to capture. Furthermore, real-world systems are multimodal, i.e., they operate in many different configurations. Modeling their dynamics in a concise and efficient framework is a key challenge. In practice, balancing the details incorporated into the model to ensure diagnosability, while keeping the model complexity manageable, is an additional challenge.

The problem of monitoring complex systems to detect faulty behavior also presents a number of challenges. In model-based diagnosis, a model of the system is used to predict nominal behavior, and deviations between observed and predicted behaviors signal the presence of faults. However, system monitoring is often performed with incomplete information due to lack of sensors or with sensors that only provide data at rates slower than what is required to accurately estimate the system state. In addition, uncertainty in both the measurements and the system model may degrade the estimation accuracy. In spite of these difficulties, fault detection must be robust to minimize false alarms, missed detections, and detection delays.

Challenges also arise in the fault isolation task. Different types of faults (abrupt, incipient, and discrete) can manifest in system components, sensors, and actuators. Interactions among components may make it hard to distinguish between faults. Furthermore, fault isolation is impacted by the granularity of the model and the measurements that are available to the diagnosis system. Even with these issues, diagnosis algorithms must provide robust, accurate, and precise results in a timely manner. Computational issues arise in accomplishing this goal, particularly with large-scale nonlinear multimodal systems. Efficiency and scalability thus become key concerns.

Manuscript received April 29, 2008; revised June 27, 2009. Date of publication July 8, 2010; date of current version August 18, 2010. This work was supported in part by the National Science Foundation under Grant CNS-0615214 and in part by NASA NRA under Grant NNX07AD12A. This paper was recommended by Guest Editor J. De Kleer.

M. J. Daigle is with the University of California, Santa Cruz, CA 95064 USA at the NASA Ames Research Center, Moffett Field, CA 94035 USA (e-mail: matthew.j.daigle@nasa.gov).

I. Roychoudhury is with SGT, Inc., Greenbelt, MD 20770-6521 USA at the NASA Ames Research Center, Moffett Field, CA 94035 USA (e-mail: indranil.roychoudhury@nasa.gov).

G. Biswas and X. D. Koutsoukos are with the Institute for Software Integrated Systems and the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN 37235 USA (e-mail: gautam.biswas@vanderbilt.edu; xenofon.koutsoukos@vanderbilt.edu).

A. Patterson-Hine and S. Poll are with the NASA Ames Research Center, Moffett Field, CA 94035 USA (e-mail: ann.patterson-hine@nasa.gov; scott.poll@nasa.gov).

Digital Object Identifier 10.1109/TSMCA.2010.2052038

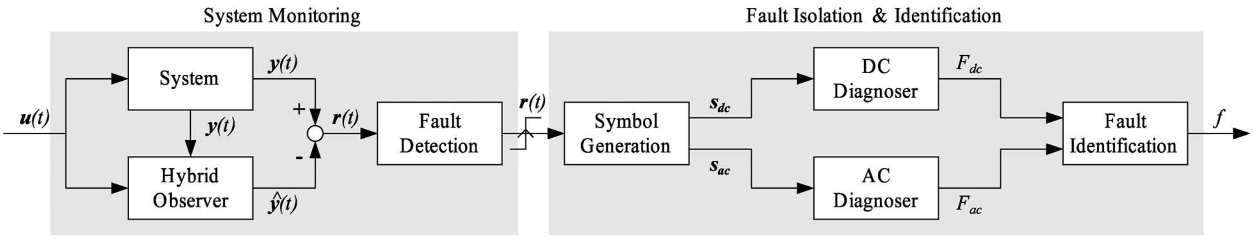

Fig. 1. Run-time diagnosis architecture.

The Advanced Diagnostics and Prognostics Testbed (ADAPT), which was developed at the NASA Ames Research Center, emulates a spacecraft power storage and distribution system [7]. It is designed to provide an environment where researchers and practitioners can tackle the challenges of diagnosis and prognosis in a realistic environment. The testbed is multimodal and can be commanded into many different configurations through the use of relays and circuit breakers. The system is multidomain, and the heterogeneous components exhibit diverse behavior characteristics with contrasting time constants. Therefore, faults in the system take on multiple forms, and the faulty behaviors can evolve at widely differing rates. The system includes a number of sensors that measure system variables, such as voltages, currents, and temperatures. However, only limited information and data are available to estimate and validate the parameters of the dynamic models of the system. In addition, the system is limited by the data collection rate at which these behaviors are monitored.

In order to address the challenges of diagnosis of real-world systems, we have developed the Fault-Adaptive Control Technology (FACT) tool suite [8], which encompasses a comprehensive modeling and diagnosis approach for hybrid systems based on the Hybrid TRANSCEND methodology [9]. To apply the framework to ADAPT, we devise a number of innovative and novel extensions. First, we extend our hybrid diagnosis scheme for parametric faults to a combined parametric and discrete fault scheme. Second, to handle the limited sensors and the fast transients in the ac subsystem, we develop a new model-driven approach for deriving parametric and discrete fault signatures for ac measurements. Third, we develop a comprehensive methodology for combined diagnosis of hybrid systems with dc and ac subsystems, using an extension of our previous diagnosability-based distributed diagnosis methods from continuous systems [10] to hybrid systems. We illustrate the effectiveness of our extended approach with experimental studies conducted on the hardware testbed and in a fully developed simulation environment called VIRTUAL ADAPT [7]. Much of FACT has been presented in previous papers (e.g., [8]–[13]), so we briefly cover the previously developed aspects of FACT and only provide details that are pertinent to the specifics of ADAPT. Our major technical focus is on the new methods developed to address the challenges specific to ADAPT, including extensions to discrete faults, comprehensive diagnosis of dc and ac subsystems, and distributed diagnosis for hybrid systems.

This paper is organized as follows: Section II presents the challenges that arise in diagnosing faults in ADAPT and how we approach them with FACT. Section III describes our modeling scheme. Section IV discusses our approach to monitoring complex hybrid system behaviors and online fault detection. Section V describes our integrated framework for diagnosis of the heterogeneous components of the ADAPT testbed, and Section VI discusses the details of our online fault isolation scheme. Section VII discusses our experimental results, and Section VIII provides the conclusions and our directions for future work on real-world diagnosis applications.

II. FAULT-ADAPTIVE CONTROL TECHNOLOGY TOOL SUITE

The FACT tool suite uses a model-integrated computing approach to automatically synthesize simulation models, hybrid observers, and diagnoser code from hierarchical component-based system models [8]. In this paper, we present a particular instantiation of the FACT architecture for diagnosis in ADAPT. Although the tool suite has been developed for general engineering systems, we have customized particular features and developed new methodologies to address specific challenges. The run-time computational architecture of FACT implemented for ADAPT is shown in Fig. 1. We assume that $\mathbf{u}(t)$ represents the inputs (controlled or otherwise) to the system under diagnosis and $\mathbf{y}(t)$ represents the system outputs. A nonlinear observer built using the component-based nominal system model is used to generate the residual signals for the fault detection process by comparing the actual and predicted behaviors. Statistically significant nonzero residuals $\mathbf{r}(t)$ trigger the symbol generator. Symbols $\mathbf{s}_{\mathrm{dc}}$ (corresponding to dc measurements) and $\mathbf{s}_{\mathrm{ac}}$ (for ac measurements) are input to the qualitative fault isolation processes for the dc and ac subsystems implemented as a distributed isolation scheme. Parameterized fault candidates may be fed into a fault identification unit to determine fault magnitude and for further hypothesis refinement [9], [14].

A. Model Development

The ADAPT system schematic, as shown in Fig. 2, illustrates a typical functional representation of the *power generation* (two battery chargers), *power storage* (three sets of lead-acid batteries), and *power distribution* components (two inverters; a number of relays and circuit breakers; and a variety of dc and ac loads, including fans, lights, and pumps) of a spacecraft's electrical power system. Sensors measure voltages, currents, temperatures, and frequencies (which are denoted in Fig. 2 using circles). The system includes elements from the electrical, chemical, mechanical, and hydraulic domains. Furthermore, the system contains more than 50 switching elements, which

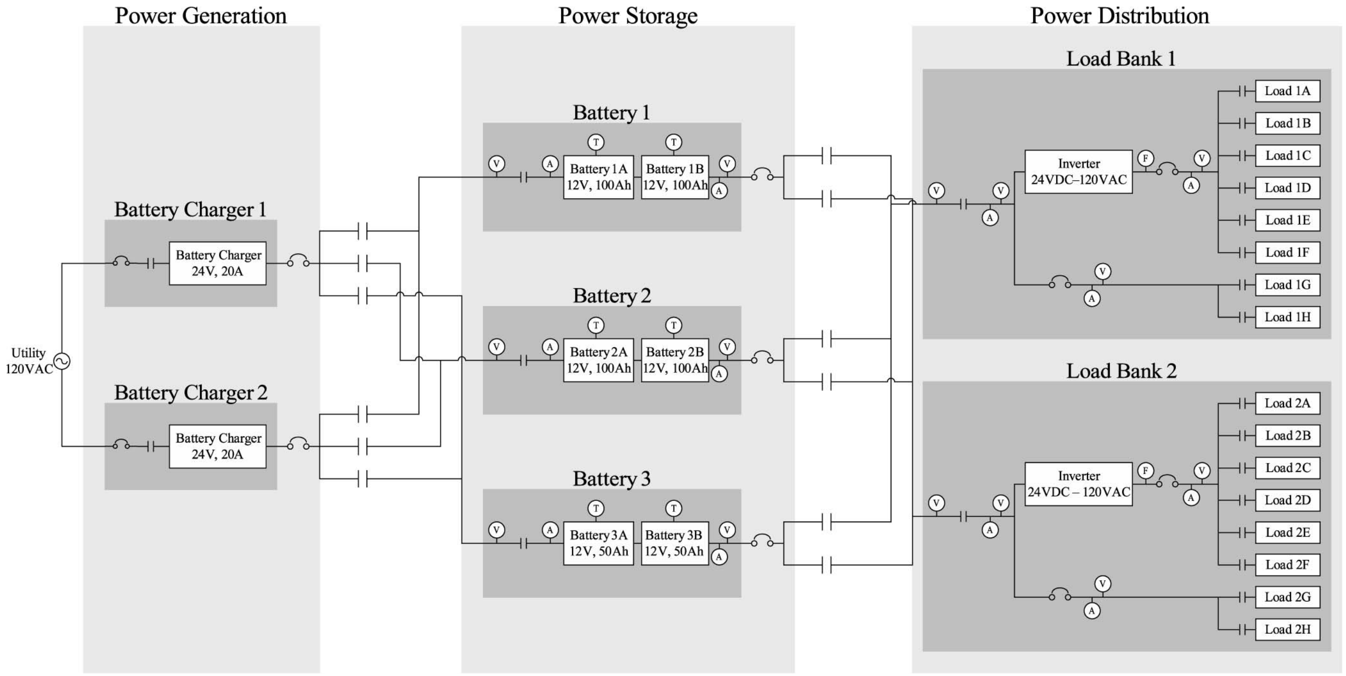

Fig. 2. Schematic diagram of ADAPT.

implies that it can potentially operate in more than 2^{50} distinct modes and that the system behavior is inherently hybrid. Therefore, we require a modeling framework that can seamlessly integrate models from various physical domains and concisely capture all possible switching behaviors.

With large-scale systems such as ADAPT, which contains more than 170 components and more than 220 possible individual faults, model development is a difficult, time-consuming task. We alleviate this burden by utilizing a component-based modeling approach that includes a library of parameterized component models, which are then composed to form the complete system model. Our modeling approach builds on the hybrid bond graph (HBG) language [15]. The HBG language, as described in Section III, supports energy-based topological modeling of physical processes in multiple domains using generic elements, such as dissipators of energy (resistances), energy storage elements (capacitors and inertias), and source elements (efforts and flows) that represent inputs to the system. System components are modeled as HBG fragments, which are connected through energy and signal ports to define the complete system behavior.

In HBGs, switching is defined at the component level. The use of localized switching functions avoids the state explosion in model building for hybrid systems. Pre-enumeration of the complete set of system modes is not required. For a particular mode, the system equations can automatically be derived from the model configuration and the constituent model for each component based on *causality*, i.e., the preferred order for computing the effort and flow variable values [16]. A large number of discrete modes can efficiently be handled, because we can systematically update the current models to those for a new mode by exploiting efficient causality reassignment procedures in HBGs [17].

Our modeling framework is implemented within a model-integrated computing paradigm using the Generic Modeling Environment (GME), which is a meta-modeling framework for specifying domain-specific modeling languages [18]. We construct system models using graphical interfaces provided in GME and design model transformations for automatically synthesizing code for the components of the run-time application. This approach greatly simplifies the entire development process, from creating and testing the initial prototypes to generating the diagnosers for the run-time environment. Fig. 3 overviews the set of model transformations. The graphical model is transformed into a simulation model [17], which can be used throughout the development and testing cycles. Another transformation process generates a model file that serves as input to the run-time application. At run time, the HBG model is reconstructed from the model file and is automatically transformed to 1) a set of state and output equations for the hybrid observer [9] and 2) the qualitative diagnosis model, which is known as the temporal causal graph (TCG) [11]. The system developer needs only to supply the HBG model of the system. From this model, FACT can derive all the models it requires for tracking and diagnosis, and synthesize the run-time code. Customization and tuning are supported through parameters associated with the software modules.

B. System Monitoring

A complete model-based approach to online diagnosis requires methods for accurately tracking the dynamic system behavior in the presence of modeling errors, measurement noise, and disturbances in the system. A standard approach for accomplishing this task is to use an observer, which can accommodate model errors and measurement noise to provide

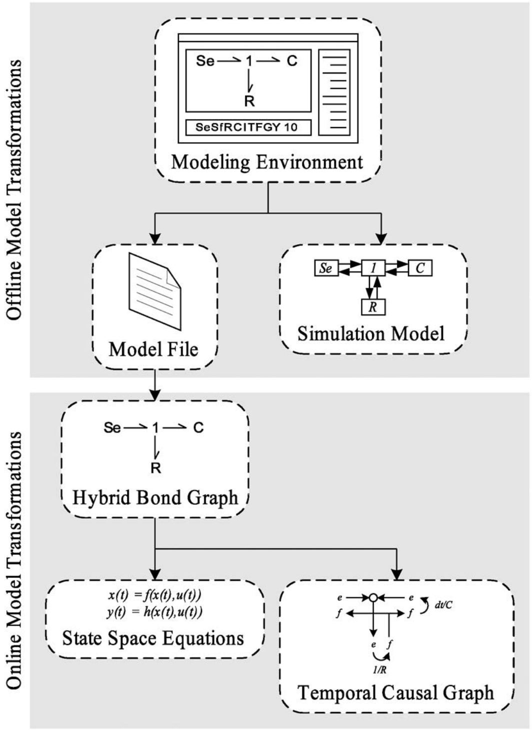

Fig. 3. Model transformations in FACT.

robust estimates of true system behavior. Along with uncertainty and noise, the monitoring behavior in ADAPT must overcome additional difficulties, because the ac components operate at 60 Hz, whereas the data acquisition system samples the data at 2 Hz. Furthermore, the sensors only measure rms and phase values of the corresponding ac signals. To overcome this, we design an extended observer that tracks the fast ac behavior with relatively infrequent measurement updates. We do this using a scheme that updates the state estimates when new measurements are made available but simulates the ac behavior in between updates at the required high rate. The scheme is described in detail in Section IV.

Since ADAPT is hybrid, the observer must also handle mode changes when tracking system behavior. We accomplish this by using an extended Kalman filter (EKF) in conjunction with an automaton to track the system modes. This hybrid observer, which is constructed from the HBG model, is automatically reconfigured when mode changes occur. This method has been described in detail in previous work [9] and is briefly described in Section IV.

Faults can be detected by comparing the actual and estimated system outputs, but measurement noise complicates this task. In ADAPT, the measurements can be noisy, as shown in Table I and the plots in Section VII. The amount of noise may also vary in time, particularly the battery voltage sensors exhibit greater noise when the battery output voltage becomes low. Noise in the signals requires the design of robust statistical tests to determine if a measurement has truly deviated from its nominal value. We implement fault detection as a test of statistical significance using the Z-test [19] coupled with a sliding window technique [13]. Systematic analysis is required to achieve the proper tradeoff between sensitivity of detection and false alarm generation. Fault detectors are tuned to adjust sensitivity in order to minimize false alarms and missed detections. For ADAPT, we customize our fault detectors to have the highest sensitivity to faults without producing false alarms. Implementation details of our fault detection scheme have been described in [13] and are presented in Section IV for completeness.

TABLE I
MEASUREMENT NOISE IN ADAPT

Sensor Type	Standard Deviation of Noise
DC Voltage	0.0190 V
DC Current	0.0261 A
AC Voltage	0.0334 V rms
AC Current	0.0114 A rms
Phase	0.0016 rad
Rotational Speed	29.1370 RPM
Temperature	0.2904 °C

C. Fault Isolation

The faults considered for ADAPT cover a large subset of faults observed in spacecraft power storage and distribution systems [20]. Faults in ADAPT can manifest as abrupt faults, i.e., unexpected abrupt changes in system parameter values, and discrete faults, i.e., unexpected changes in the operating mode. Incipient faults may also occur, but we do not consider them for this work. Faults may occur in sensors (e.g., additive sensor bias), the process (e.g., a change in a resistance value), or the actuators (e.g., stuck-at faults in relays). FACT implements the Hybrid TRANSCEND methodology that combines qualitative and quantitative diagnosis for hybrid systems [9], [11]. The approach originally addressed parametric faults, and in recent work, motivated by ADAPT, it has been extended to incorporate discrete faults [12]. These previously developed methods, as presented in Section V, can directly be applied to the dc components of ADAPT.

However, transient analysis of fault signatures cannot directly be applied to diagnosing faults in the ac components of ADAPT, because a sampling rate of 2 Hz is too slow to capture ac transients. In addition, the available ac sensors measure steady-state rms and phase values of the ac signals. In this paper, we extend our fault signature generation scheme to derive steady-state fault signatures for ac signals, given parametric and discrete faults in the ac components of the system. In addition, we develop a methodology where the dc and ac diagnosers independently operate in a distributed manner.

A distributed diagnosis scheme without centralized coordination provides additional advantages in reducing the computational complexity and, therefore, improving the overall scalability of the diagnosis process. In [10], we discuss our approach to the distributed diagnosis of continuous systems. In this paper, we extend the approach to hybrid systems and apply it to the ADAPT system, based on the diagnosability analysis of the hybrid system model. Section V discusses the details. The extended diagnosis schemes and the distributed diagnosis approach provide an innovative framework for developing a comprehensive model-based diagnosis methodology for spacecraft power distribution systems and allow us to perform dc and ac diagnosis using two independent diagnosers.

III. Model Development

Our component-based models of hybrid physical systems are based on the HBG modeling language [15]. HBGs extend bond graphs (BGs) [16] and are particularly suitable for diagnosis, because they incorporate causal and temporal information, along with the mode change information required for deriving and analyzing fault transients. In BGs, components are vertices, and bonds, which are drawn as half arrows, capture ideal energy connections between the components. Associated with each bond are two variables: *effort* e and *flow* f, the product of which defines the rate of energy transfer through the bond. In the electrical domain, effort and flow map to voltage and current, respectively. 1-junctions are analogous to series connections (f values on incident bonds are equal and $\sum e = 0$), and 0-junctions are analogous to parallel connections (e values on incident bonds are equal and $\sum f = 0$). Component behaviors are modeled as resistances R, which capture energy dissipation in the system $(e = Rf)$; capacitances C $(\dot{e} = (1/C)f)$ and inductances I $(\dot{f} = (1/I)e)$, which capture energy storage functions; and sources of flow Sf and effort Se, which model the flow of energy into and out of the system. Nonlinearities are modeled by expressing system parameters as functions of system variables using modulating elements. The constituent equations of the BG elements define a set of differential algebraic equations describing the continuous system behavior.

HBGs introduce *switching junctions*, which act as ideal switches in the model, enabling a junction to be in either the on or the off mode of operation [15]. Off 1-junctions behave as sources of zero flow. Similarly, off 0-junctions act as sources of zero effort. When on, switching junctions behave as normal junctions. The switching behavior is defined by a *control specification* (CSPEC), which is modeled as a finite automaton, whose state determines whether the junction is on or off [9], [15]. The overall system mode is implicitly defined by the individual states of all the CSPECs, and this provides a concise representation of the hybrid system model.

Consider the example electrical circuit shown in Fig. 4. The circuit consists of an ac source Se with voltage $v(t)$, resistors R_1 and R_2, inductor L_1, and capacitor C_1. The series and parallel connections in the circuit are captured using the 1- and 0-junctions, respectively. The switch Sw_1 is modeled by an ideal switching 1-junction, representing a series connection that can be on or off. The switching junction is denoted by the dashed arrow in Fig. 4(b). The corresponding CSPEC determines the state of the switching junction and is nominally controlled by events Sw_1 and $\neg Sw_1$.

In this work, we focus on the diagnosis of single persistent faults in hybrid systems. We classify faults into two categories: 1) *parametric faults* and 2) *discrete faults*. Parametric faults, which represent partial failures or degradations in system components, manifest as abrupt changes in the HBG model parameter values. Discrete faults correspond to differences between the actual and expected states of a switching component in the HBG model and are modeled using unobservable fault events in the CSPECs that cause unexpected changes in junction state [12]. For example, the Sw_1^{on} and Sw_1^{off} events in Fig. 4(b) correspond to stuck-on and stuck-off faults of Sw_1 and, unexpectedly, change the state of the 1-junction associated with the CSPEC.

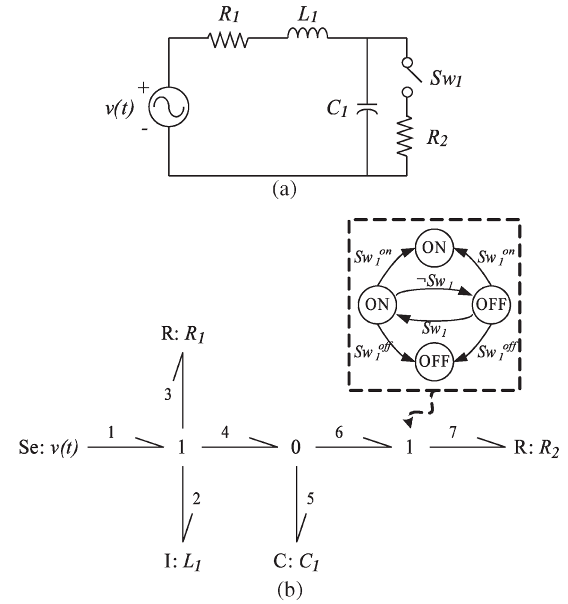

Fig. 4. Switched circuit example. (a) Circuit schematic. (b) HBG.

The ADAPT model is a composition of component models of the batteries; inverters; relays; circuit breakers; dc loads that include simple circuits; and ac loads that include fans, pumps, and light bulbs. The states of the various CSPECs establish the different configurations of the system. From the HBG models, we can derive a hybrid state-space formulation that forms the basis for the hybrid observer and the parameter estimation scheme, a reconfigurable block diagram that forms the basis of our simulation models, and the TCG, which forms the basis for performing qualitative fault isolation from transients.

A. Generating Simulation Models

We use the HBG to automatically generate simulation models of the system for offline diagnosis experiments. Each mode of the HBG corresponds to a BG model that defines the continuous behavior within a mode. The computational model for each mode (e.g., state-space equations, block diagrams, or signal flow graphs) can systematically be derived from each BG model using well-defined methods [16]. We have developed efficient methods for incrementally regenerating the computational model after a mode change occurs [21], which offer significant advantages for large hybrid systems such as ADAPT, because it avoids unnecessary preenumeration of all system modes. Instead, the computational model is locally reconfigured to the new mode. This scheme has been used to develop the Virtual ADAPT simulation testbed.

Parametric faults can be introduced into component simulation models by specifying the time of fault occurrence, the fault profile (in our case, abrupt), and the magnitude change in the parameter value. Discrete faults are introduced by specifying a particular discrete fault profile (e.g., uncontrolled switching or stuck faults) at specific points in time. This provides us with mechanisms for generating fault data sets for experimental

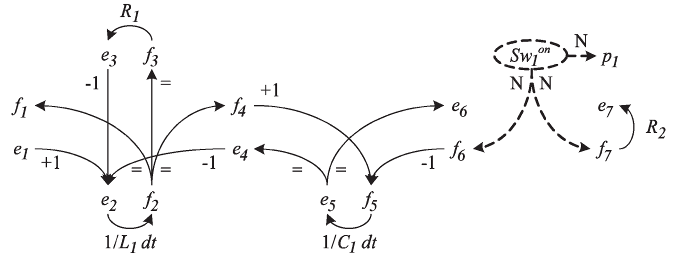

Fig. 5. Example TCG for the nominal mode, where the switch is off.

studies. In particular, faults that are dangerous or impossible to inject in the actual hardware can be studied using the simulation.

B. Temporal Causal Graphs

Our model for qualitative fault diagnosis, the TCG is derived from the HBG model of the system for a given mode. The model explicitly captures the propagation of both parametric and discrete fault effects on other system variables, including measured variables [11], [12]. The TCG is essentially a signal flow graph whose nodes are system variables or discrete fault events. The labeled edges represent the qualitative relationships between the variables, i.e., equality $(=)$, direct $(+1)$, inverse (-1) proportionality, integral (dt), and parametric dependencies (e.g., $1/R_1$), where changes in the involved parameters reflect parametric faults. The algebraic relations imply instantaneous propagation effects, whereas the integral edges imply a delay in the propagation that manifest as higher order effects (e.g., changes in slope). Links from discrete fault events to variables may have ± 1 labels and additional N and Z labels, if the fault causes the variable value to go from zero to nonzero (N) or from nonzero to zero (Z). The directionality of the edges is determined by causality, where the causal directions are derived from the BG model [16].

The TCG for the circuit example is given in Fig. 5 for the mode where the switch is off. It contains the system flow and effort values in addition to an explicit value for the on/off position of Sw_1 and p_1. In this mode, the TCG must include the discrete fault where the switch unexpectedly turns on (represented by fault event Sw_1^{on}). If this fault occurs, then the flow of current through the switch will go from zero to a nonzero value, which then affects the values of other variables in the system. A change in a parameter value caused by a fault, e.g., R_1^+, cannot cause discrete changes between zero and nonzero values.

IV. System Monitoring

As discussed earlier and illustrated in the architecture of Fig. 1, the fault detector triggers the fault isolation and identification modules. The robust fault detection scheme combines a hybrid observer for tracking nominal system behavior and a statistical hypothesis testing scheme for robust fault detection.

A. Hybrid Observer

The hybrid observer combines the use of an EKF for tracking continuous system behavior and automata for tracking the on/off mode of every switching junction in the HBG model and transitioning to the new mode when indicated by the CSPECs [9]. We assume that both controlled and autonomous mode changes can accurately be tracked under nominal system operation. Mode changes produce a reconfiguration in the HBG model. As a result, the state-space equations are automatically recomputed, the EKF equations are updated, and the tracking of continuous behavior resumes. The EKF scheme assumes the modeling errors, and measurement noise are uncorrelated Gaussian with zero mean. Therefore, the two covariance matrices that represent the modeling error and measurement noise are assumed to have known variance values.

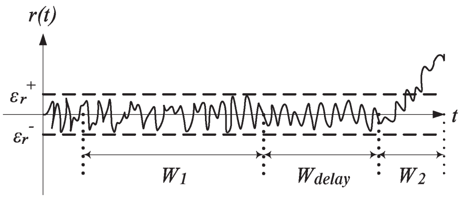

Fig. 6. Sliding windows in the fault detection scheme.

The observer receives updated measurements at a rate of 2 Hz. However, the system equations must be run at faster rates to accurately simulate ac system behavior, due to the controlled fast-switching behavior of the inverter [17]. Since instantaneous ac current and voltage measurements are not available at such rates, the observer can only update at the rate of 2 Hz. To address this issue, we run the observer at the rate required by the ac equations but only perform the EKF update at the 2-Hz rate, i.e., whenever new observations are made available. All instantaneous measurements contribute to the state update function in the EKF. For the ac subsystems, the rms and phase sensor readings are based on computations that take place over a window of samples, so they cannot directly be used in the EKF update functions. Since EKF updates are performed whenever new data are available, this approach has the most utility. In addition, faults occurring between EKF updates that produce observable changes in rms and phase measurements will still be detected.

B. Fault Detection

Our fault detection scheme employs independent fault detectors for each sensor. This allows each detector to be individually tuned to achieve maximum sensitivity for a given signal and allows the fault detection task to be easily distributed across large systems. For each measurement $y(t)$, we define the residual as $r(t) = y(t) - \hat{y}(t)$, where $\hat{y}(t)$ is the estimated output signal generated by the hybrid observer. The fault detection scheme employs the Z-test to look for nonzero residual signals [13].

The Z-test requires that the sample mean and standard deviation of a given population be known [19]. We estimate the population standard deviation and sample mean using a sliding window technique illustrated in Fig. 6. A small sliding window (e.g., five samples) W_2 is used to estimate the current mean $\mu_r(t)$ of a residual signal

$$\mu_r(t) = \frac{1}{W_2} \sum_{i=t-W_2+1}^{t} r(i).$$

The variance of the nominal residual signal $\sigma_r^2(t)$ is computed using a window W_1 preceding W_2, where $W_1 \gg W_2$ (e.g., 100 samples). W_1 is offset by W_2 by a buffer W_{delay} (e.g., 50 samples) to ensure that W_1 does not contain any samples after fault occurrence. The variance is computed as

$$\sigma_r^2(t) = \frac{1}{W_1} \sum_{i=t-W_2-W_{\text{delay}}-W_1+1}^{t-W_2-W_{\text{delay}}} \left(r(i) - \mu_r'(t)\right)^2$$

where

$$\mu_r'(t) = \frac{1}{W_1} \sum_{i=t-W_2-W_{\text{delay}}-W_1+1}^{t-W_2-W_{\text{delay}}} r(i).$$

Given a prespecified confidence level α (e.g., 95%), the tables provide the bounds z^- and z^+ for a two-sided Z-test. The thresholds for the fault and no-fault decisions $\varepsilon_r^-(t)$ and $\varepsilon_r^+(t)$, respectively, are computed as

$$\varepsilon_r^-(t) = z^- \frac{\sigma_r(t)}{\sqrt{W_2}} + E$$

$$\varepsilon_r^+(t) = z^+ \frac{\sigma_r(t)}{\sqrt{W_2}} - E$$

where E is a modeling error term. A computed mean value $\mu_r(t)$ that lies outside of the thresholds at time t implies a fault. In practice, parameters W_1, W_2, and W_{delay}, confidence level α, and the modeling error term E of the fault detector have to be experimentally tuned to optimize performance, i.e., minimize false alarms while keeping detection sensitivity high [13].

V. Diagnoser Design

Our approach to diagnosing faults in power distribution systems, such as ADAPT, combines schemes for diagnosis from transients for dc measurements and changes in steady-state values for ac measurements. We develop the system diagnoser as two distributed diagnosers: 1) the dc subsystem diagnoser and 2) the ac subsystem diagnoser. The diagnoser design is based on deriving fault signatures for the dc and ac components of the system and then performing diagnosability analysis using the fault signatures.

A. Fault Signatures for DC Measurements

For the dc measurements, the fault signatures are derived from the transients generated at the point of fault occurrence t_f. Assuming that the system output is continuous and continuously differentiable, except at the points of fault occurrence and mode changes, the transient response after abrupt fault occurrence can be approximated by a Taylor series expansion, which is defined by the changes in magnitude and higher order derivatives in the signal at t_f [11], [14]. In TRANSCEND, the fault signatures are expressed in a qualitative form: $+$ (increase), $-$ (decrease), and 0 (no change) in the magnitude and derivatives of the residual signal. If a fault produces an immediate change in the residual, i.e., a discontinuity at t_f, then the magnitude symbol is $+$ or $-$; otherwise, it is 0. Ambiguity in a signature is denoted by the $*$ symbol. In previous work, we have shown that the first change and subsequent slope provide all of the discriminatory evidence for qualitative fault isolation in dynamic systems [14]. Therefore, our fault signatures include two symbols: 1) the magnitude change and 2) the slope of the residual signal.

TABLE II
Fault Signatures for DC Measurements for the Circuit With the Switch On

Fault	V_{R1}	I_{R_2}
C_1^+	`0+,X`	`-+,X`
C_1^-	`0-,X`	`+-,X`
L_1^+	`-+,X`	`0-,X`
L_1^-	`+-,X`	`0+,X`
R_1^+	`-+,X`	`0-,X`
R_1^-	`+-,X`	`0+,X`
R_2^+	`0-,X`	`-+,X`
R_2^-	`0+,X`	`+-,X`
Sw_1^{off}	`0-,X`	`-*,Z`

For discrete fault analyses, fault signatures have been extended to include a third symbol that indicates if a fault causes a zero-to-nonzero or nonzero-to-zero value change in measured values from estimated values. Discrete faults cause mode changes at junctions, and as a result, variable values linked to this junction may abruptly go from nonzero to zero (for a junction turning off) or from zero to nonzero (for a junction turning on). The symbols `N`, `Z`, and `X` represent zero-to-nonzero, nonzero-to-zero, or no discrete change behavior in the measurement from the estimate [12].

Fault signatures representing the transient behavior due to parametric and discrete faults are defined as follows for our three-symbol representation:

Definition 1 (Fault Signature From Transients): A *fault signature* for a fault f in a system mode q defines the qualitative effect in magnitude, slope, and discrete change in measurement m due to the occurrence of f.

Fault signatures are derived for each hypothesized fault f in mode q by performing a forward propagation function on the TCG [11], [12]. In the circuit example, we denote the mode where the switch is off as q_0 and the mode where the switch is on as q_1. Signatures for mode q_1 are given in Table II, assuming that the voltage source is dc, instead of ac, and variable values are nominally positive, where the measurements are the voltage across R_1 and V_{R_1}, and the current through R_2 and I_{R_2}. For example, an abrupt increase in the value of C_1, which is denoted as C_1^+, will cause a smooth increase in V_{R1} and a transient characterized by an abrupt increase and subsequent smooth decrease in I_{R2}. The table shows that the system is not diagnosable with the selected measurements, because faults in L_1 and R_1 cannot be distinguished in this mode.

B. Fault Signatures for AC Measurements

Analyzing fault transients in the ac domain would require sampling at rates that are much faster than 2 Hz, which would make the diagnoser computationally infeasible. In addition, as discussed earlier, we can only measure the rms and phase values of the ac voltages and currents with a 2-Hz sampling frequency. Therefore, from practical considerations, the ac fault signatures represent steady-state deviations in the measurements. These fault signatures can be derived by computing the partial derivative of the steady-state expression for a measurement with respect to a given fault variable to determine the sign of the measurement value change. In general, steady-state signatures may result in large delays in detection and isolation; however, in ADAPT, changes in rms and phase occur within the 2-Hz sampling window, so there are no delays relative to transient analysis for the dc measurements.

This analysis starts by deriving the symbolic expressions relating faults to the measurements using the HBG model of the system. The parameters for the R, C, and I elements are replaced by their complex impedance representations in the ac domain. Given the frequency ω in radians, the impedance of a resistance R, a capacitor C, and an inductor L is $Z_R = R$, $Z_C = (1/j\omega C)$, and $Z_L = j\omega L$, respectively. By combining the constitutive relations of the elements and the junction equations derived from the HBG, we can generate the voltage and current variable relations in symbolic form. By algebraic manipulation, we get the symbolic form of the expressions for the ac measurements as a function of a given fault. After substituting nominal values of all other parameters, if the sign of this partial derivative is always positive (negative) for the considered fault magnitudes, then the corresponding fault signature is defined to be a $+$ $(-)$. If the sign cannot be uniquely determined, the ambiguity is represented using the $*$ symbol. Since discrete faults represent changes in system mode, we determine the signatures by simply computing the rms and phase values for the different fault configurations and then comparing them to nominal configurations to compute the fault signatures for the discrete faults.

Definition 2 (Fault Signature by Steady-State Analysis): A *fault signature* for a fault f in a system mode q defines the qualitative effect in magnitude and discrete change in measurement m due to the occurrence of f.

To illustrate the approach, we consider the circuit of Fig. 4(a). The measured signals are the voltage across R_1 and V_{R_1}, and the current through R_2 and I_{R_2}. The measurements include both rms values and phase difference relative to the source voltage for both measured signals. We assume that the source voltage $v(t)$ is 120 Vrms at 60 Hz, and the parameters have nominal values of $C_1 = 0.005$ F, $L_1 = 0.03$ H, $R_1 = 1\ \Omega$, and $R_2 = 2\ \Omega$. We need to analyze the effects of faults in both system modes q_0, where the switch is off, and q_1, where the switch is on. Using the HBG as previously described, we derive the symbolic expressions describing the measurements as a function of the inputs and the impedances of the four components

$$V_{R_1} = \frac{vR_1}{Z_{\text{eq}}}$$

$$I_{R_2} = \begin{cases} 0, & \text{for mode } q_0 \\ \frac{vZ_{C_1,R_2}}{Z_{\text{eq}}R_2}, & \text{for mode } q_1 \end{cases}$$

TABLE III
FAULT SIGNATURES FOR AC MEASUREMENTS FOR THE CIRCUIT WITH THE SWITCH ON

Fault	V_{R_1}	$\phi_{V_{R_1}}$	I_{R_2}	$\phi_{I_{R_2}}$
C_1^+	−,X	−,X	−,X	−,X
C_1^-	⋆,X	+,X	+,X	+,X
L_1^+	−,X	−,X	−,X	−,X
L_1^-	+,X	+,X	+,X	+,X
R_1^+	+,X	+,X	−,X	+,X
R_1^-	−,X	−,X	+,X	−,X
R_2^+	+,X	−,X	−,X	−,X
R_2^-	−,X	⋆,X	+,X	+,X
Sw_1^{off}	−,X	+,X	+,Z	−,Z

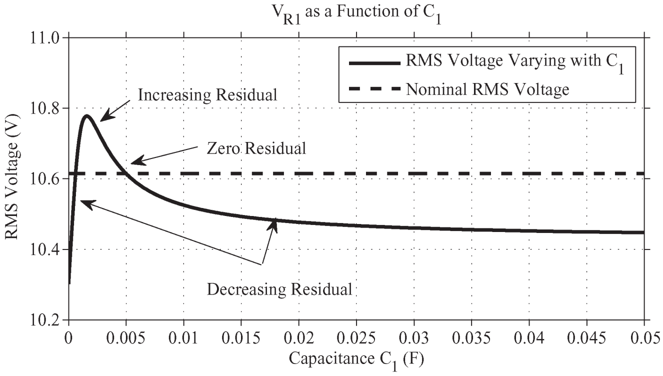

Fig. 7. V_{R_1} rms value as a function of C_1 magnitude in mode q_1.

where

$$Z_{C_1,R_2} = \left(j\omega C_1 + \frac{1}{R_2}\right)^{-1}$$

$$Z_{\text{eq}} = \begin{cases} j\omega L_1 + R_1 + \frac{1}{j\omega C_1}, & \text{for mode } q_0 \\ j\omega L_1 + R_1 + Z_{C_1,R_2}, & \text{for mode } q_1. \end{cases}$$

These symbolic expressions for impedances are used to compute the fault signature matrix for each mode.

The steady-state signatures for mode q_1 are shown in Table III. In some cases, the direction of change in measurement values depends on fault magnitude. For example, C_1^+ will always cause a decrease in the rms value of V_{R_1}, but C_1^- may cause either an increase or decrease in V_{R_1}, depending on its magnitude, as shown in Fig. 7. For its nominal value of 0.005 F, with an increase in C_1, the measurement value always decreases, but for a decrease in magnitude, the measurement value may go above or below the nominal measurement value, so we represent the signature in this case as a $*$ (see Table III). Discrete faults do not produce ambiguous signatures, because we can always compute steady-state values in two separate modes and determine the qualitative difference. For example, when the switch is on, the rms value of I_{R_2} is 2.83 A, and when off, it is zero; therefore, when unexpectedly going from q_1 to q_0, we will observe a decrease $(-)$ in I_{R_2}, and it will go to zero (Z). This is represented by the fault signature $-$, Z.

C. Distributed Diagnoser Design

Distributed diagnosers partition the diagnosis task into smaller subtasks, thus reducing the computational complexity of the diagnosis algorithm [22]. In [10], we presented an approach for designing distributed diagnosers for continuous systems whose subsystem structure is given (Algorithm 1 in [10]). In this paper, we extend this approach to hybrid systems, which allows us to decouple the diagnosers for the dc and ac subsystems of ADAPT. Our objective is to decompose the overall diagnosis task into smaller subtasks performed by local diagnosers such that the local diagnosers generate globally correct diagnosis results while minimizing the number of measurements communicated among the local diagnosers.

To generate distributed diagnosers for hybrid systems, we require the fault signatures for each mode of the system, and these are generated using the techniques discussed in the previous section. Mode changes may occur during fault isolation, so fault signatures for one mode interleaved with fault signatures for another mode are possible. These cases must be accounted for, because they affect the diagnosability of the system [12] and, therefore, the diagnoser design process. The traces formed by measurement deviations and mode transitions can be represented as a finite automaton that maps states to consistent fault hypotheses [23]; its construction is omitted here for space but is directly formed from the fault signatures and mode change structure of the system. We denote this finite automaton as $\mathcal{D}_{F,M,Q}$, where F is the set of all possible faults, M is the set of all available measurements, and Q is the set of all system modes.

We define a subsystem $S_i = (F_i, M_i)$, where F_i is the set of faults in S_i and M_i is the set of measured variables in S_i. The separate F_i and M_i form partitions of the set of faults F and measurements M, respectively. Given κ subsystems, $S_i = (F_i, M_i)$ $(1 \leq i \leq \kappa)$, and $\mathcal{D}_{F,M,Q}$, our design problem is to construct, for each subsystem, a measurement set $\widetilde{M_i} \subseteq M$ such that 1) $\widetilde{M_i} \supseteq M_i$ is minimal and 2) all single faults in F_i are *globally diagnosable* by measurements in $\widetilde{M_i}$. We define global diagnosability as follows:

Definition 3 (Global Diagnosability): A set of faults $F_i \subseteq F$ is *globally diagnosable* by $\widetilde{M_i} \subseteq M$ if $\widetilde{M_i}$ can uniquely isolate every fault $f \in F_i$ from all other faults in F for every possible sequence of mode transitions.

We apply this concept to the diagnoser design process as follows: Each local diagnoser is characterized by a set of faults F_i that it must diagnose. For its fault set to be globally diagnosable, given a set of measurements, the fault signatures for these measurements must uniquely distinguish each fault in F_i from each fault in the *complete* fault set F. If this condition is satisfied for each local diagnoser, then this guarantees that local diagnoses will be globally correct [10].

Given the set of available measurements, global diagnosability is not always attainable in real-world systems, and in fact, we will show in Section VII that ADAPT is not globally diagnosable. We first analyze the diagnosability of the system. If the system is not globally diagnosable for a set of measurements, we define the notion of *aggregate faults*. An aggregate fault includes all single faults that are not distinguishable from one other. Our diagnosis methodology treats aggregate faults as single faults, and as a result, the reduced fault set is guaranteed to be globally diagnosable.

Given F_i and $\widetilde{M_i}$, we construct a *local diagnoser* [10] $\mathcal{D}_{F_i,\widetilde{M_i},Q}$ for each subsystem. By ensuring that each $\widetilde{M_i}$ is minimal, the local diagnosers share minimal information with one another.

The procedure for designing diagnosers for a partitioned hybrid system is presented as Algorithm 1. For each subsystem S_i, we assign to F_i^* the faults in F_i that are not globally diagnosable using measurements in M_i. The search for additional measurements is simplified by defining a notion of proximity among subsystems, which is used to prioritize the measurement selection process. We represent the system S as a graph of connected subsystems. The proximity d between subsystems S_i and S_j is defined as the minimum path length from S_i to S_j in the graph. If F_i^* is nonempty, we start with a working measurement set $\widetilde{M_i}$ that is initially equal to M_i. The proximity bound δ starts at 1. We select additional measurements from subsystems within this bound to reduce the number of faults in F_i^*. The number is selected as to be minimal while making the maximum number of faults in F_i^* globally diagnosable. The set $\widetilde{M_i}$ is expanded with these measurements, and F_i^* is reduced to a smaller set. If F_i^* remains nonempty, δ is incremented by 1, and the procedure is repeated until F_i^* is empty, expanding the search to farther subsystems. At this point, we have the local diagnoser $\mathcal{D}_{F_i,\widetilde{M_i},Q}$. We will present the results of this algorithm on ADAPT in Section VII, where we partition the system into dc and ac subsystems.

The worst-case size of $\mathcal{D}_{F,M,Q}$ is $O((|M| + |Q|)!)$, where Q is the set of all modes [23]. Diagnosability can be checked with a single pass over this structure, thus taking $O((|M|+|Q|)!)$ time. In the worst case, all measurement combinations must be considered, which is $O(2^{|M|})$ [10], where, for each combination, diagnosability is checked, resulting in a total worst-case complexity of $O(2^{|M|}(|M| + |Q|)!)$. Since the diagnoser design is performed offline, the high complexity is acceptable.

Algorithm 1 Partitioned System Diagnoser Design

```
Input: $\kappa$ local subsystems, $S_i = (F_i, M_i)$, and $\mathcal{D}_{F,M,Q}$
for each $S_i$ do
    identify $F_i^* \subseteq F_i$ that are not globally diagnosable in $\mathcal{D}_{F,M_i,Q}$
    $\delta \leftarrow 1$
    $\widetilde{M_i} \leftarrow M_i$
    while $F_i^* \neq \varnothing$ do
        identify measurement set $\widehat{M_i}$ from measurements of subsystems $S_i$ at a distance $d \leq \delta$ that isolates maximal $F_i' \in F_i^*$, and $\widetilde{M_i} - \widehat{M_i}$ is minimal
        $\widetilde{M_i} \leftarrow \widetilde{M_i} \cup \widehat{M_i}$
        $F_i^* \leftarrow F_i^* - F_i'$
        if $F_i^* \neq \varnothing$ then
            $\delta \leftarrow \delta + 1$
    construct $\mathcal{D}_{F_i,\widetilde{M_i},Q}$
```

VI. Online Fault Isolation

In this section, we describe the online fault isolation, which consists of the symbol generation method and the online signature matching scheme for qualitative fault isolation.

A. Symbol Generation

We independently define symbol generators for each sensor, so that, as with fault detection, they can be individually tuned and easily distributed. For each dc measurement, we extract the magnitude and slope of the deviation, as well as the discrete change feature. For each ac measurement, we use only the first change and the discrete change behavior. The changes are symbolically abstracted to $+$, 0, $-$, N, Z, and X symbols, and the computed symbols form the observed fault signatures that are matched to predicted signatures during fault isolation.

A robust method based on the Z-test is used for computing the symbolic features of the residual signal. If the measurement residual $r(t)$ is greater than $\varepsilon_r^+(t)$ (or less than $\varepsilon_r^-(t)$), we assign a $+$ (or $-$) to the magnitude value for the residual.

The calculation of the slope of a measurement deviation starts with the estimation of the initial residual value $\mu_{r_0}(t_d)$ at the time of fault detection t_d by computing the average of the residual samples over a small window W_3, i.e.,

$$\mu_{r_0}(t_d) = \frac{1}{W_3} \sum_{i=t_d}^{t_d+W_3-1} r(t_d+i).$$

Again using the Z-test, the slope of the residual is determined over another small window W_n (e.g., 15 samples) after the end of W_3 [13]. The mean value of the residual after fault detection is given by

$$\mu_{r_d}(t_d+t) = \begin{cases} \dfrac{\left(\sum\limits_{i=t_d}^{t_d+W_n-1} r(t_d+i)\right)}{W_n} - \mu_{r_0}, & W_n > W_3 \\ 0, & W_n \leq W_3. \end{cases}$$

It is assumed that the variance of the residual does not change due to the occurrence of the fault, i.e., $\sigma_r^2(t) = \sigma_r^2(t_d)$ for all $t \geq t_d$. The variance of μ_{r_d} is $\sigma_{r_d}^2(t_d+t) \approx \sigma_r^2/W_n$, whereas the variance of μ_{r_0} is $\sigma_{r_0}^2 \approx \sigma_r^2/W_3$. That is, the uncertainty of the initial residual value depends on the noise and W_3, whereas the uncertainty of the mean estimate depends on the noise and the number of samples used in the calculations. Using a confidence value α and the corresponding z^+ and z^- values, the $+$ slope symbol is generated when

$$\mu_{r_d} > z^+ \sigma_r \left(\frac{1}{\sqrt{W_3}} + \frac{1}{\sqrt{W_n}} \right) + E_s$$

where E_s is a modeling error term. Similarly, the $-$ slope symbol is generated when

$$\mu_{r_d} < -z^- \sigma_r \left(\frac{1}{\sqrt{W_3}} + \frac{1}{\sqrt{W_n}} \right) - E_s.$$

The size of the window used to calculate the mean W_n is increased until the symbol is successfully generated, or W_n becomes larger than a prespecified limit, at which the slope is reported as 0, implying that the true slope is either zero or unknown but very small.

The generated symbols must be translated to observed fault signatures, which requires discontinuity detection to determine whether the generated magnitude symbol represents a discontinuity or not. We assume that a discontinuity has occurred only if the generated magnitude and slope symbols are different, e.g., a generated magnitude symbol of $+$ and a generated slope symbol of $-$ will be interpreted as a $+-$ signature. In contrast, if, e.g., a $+$ symbol is generated for both magnitude and slope, we interpret this as a smooth increase, i.e., a $0+$ signature. This methodology of discontinuity detection is sufficient if the signatures $++$ and $--$ cannot be observed. This is typically the case, as these signatures imply unstable systems.

To compute the discrete change symbol, we do not use the residual but use the observed and estimated values of the signal. We compute the mean of the measured signal $y(t)$ and the mean of the estimate $\hat{y}(t)$ over a small window W_c, i.e.,

$$\mu_y(t_d) = \frac{1}{W_c} \sum_{i=t_d}^{t_d+W_c-1} y(i)$$

$$\mu_{\hat{y}}(t_d) = \frac{1}{W_c} \sum_{i=t_d}^{t_d+W_c-1} \hat{y}(i)$$

where t_d is the time of fault detection. We wish to determine whether each signal belongs to a population with zero mean and choose the variance of the population to be the variance of the residual defined by $y(t) - \hat{y}(t)$, $\sigma_r^2(t)$, as a good approximation of the true variance of the zero-mean distribution. Here, the thresholds are computed as

$$\varepsilon_{y_d}^+ = \varepsilon_{\hat{y}_d}^+ = z^+ \frac{\sigma_r(t_d)}{\sqrt{W_c}} + E_c$$

$$\varepsilon_{y_d}^- = \varepsilon_{\hat{y}_d}^- = z^- \frac{\sigma_r(t_d)}{\sqrt{W_c}} - E_c$$

where E_c is a modeling error term. These thresholds are the same as for fault detection, only they are computed for $y(t)$ and $\hat{y}(t)$ rather than $r(t)$. If $\mu_y(t_d)$ is outside its bounds, we say that it is nonzero; otherwise, we say that it is zero. Similarly, if $\mu_{\hat{y}}(t_d)$ is outside its bounds, we say that it is nonzero; otherwise, we say that it is zero. If the estimate is nonzero and the measurement is zero, we report Z. If the estimate is zero and the measurement is nonzero, we report N; else, we report X.

B. Distributed Fault Isolation

Observed fault signatures computed using symbol generation are matched to predicted fault signatures to isolate faults. Each local diagnoser, e.g., the dc and ac diagnosers, obtains the symbols for its own sensors. Inconsistent faults are eliminated, and consistent faults are retained. A globally correct diagnosis result is reached when 1) *all* measurements for a local diagnoser have deviated and the fault hypothesis set is reduced to a singleton fault set or 2) a local diagnoser's hypothesis set is reduced to a singleton but all of its measurements have not deviated, and all other diagnosers produce a *null hypothesis*, i.e., their candidate sets are empty [10].

Mode changes are handled using the approach presented in [9]. If a controlled mode change occurs, such as a relay turning on or off, the fault signatures for the new mode are used, and consistent faults must match future measurement

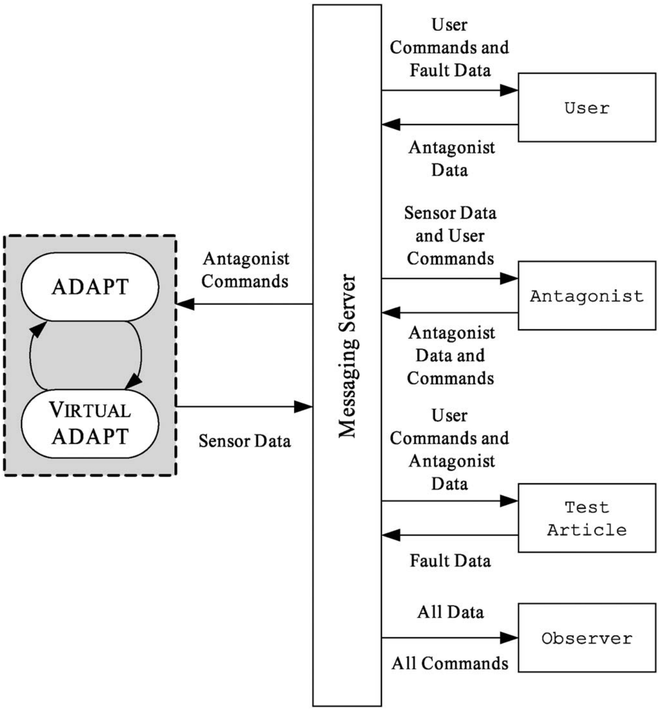

Fig. 8. Messaging architecture for ADAPT.

deviations for the current mode. If an inconsistency is obtained, autonomous mode changes are hypothesized, such as circuit breaker tripping, and consistent faults in the hypothesized modes must match the observed measurement deviations.

VII. Experimental Results

In this section, we present the experimental setup for ADAPT and demonstrate our diagnosis approach using experiments performed in simulation and experiments performed using real testbed data.

A. Experimental Setup

The current testbed operational infrastructure, as shown in Fig. 8, contains a `User` component, which simulates a crew member and provides commands to the testbed; an `Antagonist` component, which injects faults and spoofs sensor data sent to the `User`; and a `TestArticle` component, such as a diagnoser, which receives the data and commands issued by the `User` and determines the health of the system. The `Observer` component logs all system data in order to evaluate the performance of the test articles. A common communication interface between the testbed and the various components is supported through a publish/subscribe messaging server that operates at 2 Hz.

The `Antagonist` can inject discrete faults by blocking or changing user commands to the testbed and sensor faults by spoofing sensor data. Only a subset of the faults can be injected into the system, so the remainder of the faults is synthesized using Virtual ADAPT [7]. The `Antagonist` can use the simulator to realistically spoof sensor data based on simulated faulty scenarios. As indicated in Fig. 8, the simulation testbed, which is implemented in Matlab Simulink, uses external wrappers to communicate to the messaging server of ADAPT. By supporting the same interfaces as ADAPT, it functions as a portable virtual version of the actual testbed that can be used for diagnoser design and diagnosis experiments.

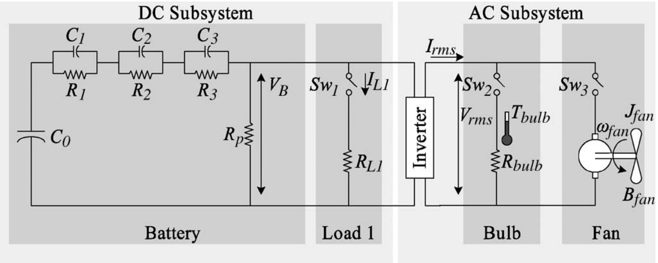

Fig. 9. Selected subset of ADAPT.

We choose a subset of ADAPT to demonstrate our approach with both the simulation and the testbed. This subset includes one of the lead-acid batteries, one dc load, an inverter, and two ac loads. The models of the dc components can be found in [23], and the models of the ac components can be found in [17]. A schematic of the subsystem is given in Fig. 9. The battery acts as a direct nonideal voltage source for the dc load. The inverter connected to the battery produces a constant 120-Vrms 60-Hz sinusoidal ac output when the input voltage is in the range of 21–32 V. When the voltage falls below 21 V, the inverter automatically shuts off. The dc load connected to the battery is purely electrical, whereas the ac loads include a light bulb and a large fan. In addition, we also consider three relays, one of which connects the dc load to the battery, whereas the remaining two connect the ac loads to the inverter. The available measurements include the rms values of the inverter voltage and current V_{rms} and I_{rms}, respectively; the phase difference between the inverter voltage and current ϕ; the temperature of the light bulb T_{bulb}; the rotational speed of the fan, ω_{fan}; the current through the dc load I_{L1}; and the battery voltage and current V_B and I_B, respectively.

We consider two subsystems (see Fig. 9): 1) the dc subsystem, which contains the battery, the dc loads, and Sw_1; and 2) the ac subsystem, which contains the inverter, the ac loads, and Sw_2 and Sw_3. The dc subsystem fault list F_{dc} includes changes in the dc load resistance R_{L1}, battery capacitance C_0, and internal battery resistance R_1, and faults in Sw_1. The dc measurements M_{dc} include I_{L1}, V_B, and I_B. The ac subsystem fault list F_{ac} includes faults in the inertia and resistance of the fan J_{fan} and B_{fan}, respectively; the resistance of the light bulb R_{bulb}; and faults Sw_2 and Sw_3. The ac measurements M_{ac} include V_{rms}, I_{rms}, ϕ, T_{bulb}, and ω_{fan}.

Fault signatures for the mode with all loads online are given in Table IV. We can see that the system is not globally diagnosable, because Sw_2^{off} and R_{bulb}^{+} cannot be distinguished. We form an aggregate fault from these two faults to apply the diagnoser design algorithm described in Section V. Using Algorithm 1, we obtain distributed diagnosers for the selected subsystems, which naturally falls out of the decoupling of the subsystems introduced by the inverter. The distributed diagnoser for the ac subsystem does not require any additional measurements from the dc subsystem to isolate its faults, i.e.,

TABLE IV
FAULT SIGNATURES FOR THE MODE WITH ALL LOADS ON

Fault	DC Measurements			AC Measurements				
	V_B	I_B	I_{L1}	V_{rms}	I_{rms}	ϕ	T_{bulb}	ω_{fan}
C_0^-	+*,X	+*,X	+*,X	0,X	0,X	0,X	00,X	00,X
R_1^+	0-,X	0-,X	0-,X	0,X	0,X	0,X	00,X	00,X
R_{L1}^+	0*,X	-*,X	-*,X	0,X	0,X	0,X	00,X	00,X
R_{L1}^-	0*,X	+*,X	+*,X	0,X	0,X	0,X	00,X	00,X
Sw_1^{off}	0*,X	-*,X	-*,Z	0,X	0,X	0,X	00,X	00,X
R_{bulb}^+	0*,X	-*,X	0*,X	0,X	-,X	+,X	0-,X	00,X
R_{bulb}^-	0*,X	+*,X	0*,X	0,X	+,X	-,X	0+,X	00,X
J_{fan}^-	0*,X	+*,X	0*,X	0,X	0,X	-,X	00,X	-+,X
B_{fan}^+	0*,X	+*,X	0*,X	0,X	0,X	+,X	00,X	0-,X
Sw_2^{off}	0*,X	-*,X	0*,X	0,X	-,X	+,X	0-,X	00,X
Sw_3^{off}	0*,X	+*,X	0*,X	0,X	-,X	-,Z	00,X	0-,X

$\widetilde{M}_{\rm ac} = \{V_{\rm rms}, I_{\rm rms}, \phi, T_{\rm bulb}, \omega_{\rm fan}\}$. This is clear from the signatures given in Table IV. If a dc fault occurs, no deviations will be observed on any of the ac measurements; therefore, the ac diagnoser will not isolate any dc faults.

The dc subsystem, on the other hand, does require ac measurements to achieve unique isolation. Faults in the ac subsystem also cause the dc measurements to deviate. To overcome this ambiguity, the distributed diagnosis design communicates the $I_{\rm rms}$ measurement to the dc diagnoser. Since dc faults do not change $I_{\rm rms}$ (due to the controlled behavior of the inverter), the dc diagnoser eliminates all local faults and determines the fault to be in the ac subsystem when $I_{\rm rms}$ deviates. If it does not deviate, the dc diagnoser will isolate a dc fault, but the ac diagnoser will not since it will not observe any deviations. Due to the autonomous mode change behavior of the inverter, the dc diagnoser also requires $V_{\rm rms}$, because the ac measurements are affected by a dc fault, if the fault is such that it causes the inverter to shut off. Hence, $\widetilde{M}_{\rm dc} = \{V_B, I_B, I_{L1}, V_{\rm rms}, I_{\rm rms}\}$. If a change occurs in $V_{\rm rms}$, then a subsequent change in $I_{\rm rms}$ is explained by the inverter shutting off and not an ac fault.

B. Simulation Results

We first present diagnosis results obtained on the simulation testbed Virtual ADAPT. We used the simulation model to provide the nominal reference for fault detection and symbol generation. For this set of experiments, we inject faults into the configuration where all loads are online. For the fault detectors, we selected $W_1 = 5$, $W_2 = 100$, $W_{\rm delay} = 50$, $W_3 = 3$, $W_n = 20$, and $\alpha = 99.97\%$. We chose $E = 0$ for all sensors, except I_B, where $E = 0.2$, and ϕ, where $E = 0.0001$.

The results are summarized in Table V. In the table, t_d is the time taken to detect a fault, and t_i is the time to isolate the fault, which is given as the point at which a diagnoser last reduces its fault set. All times in Table V are expressed in seconds. In all cases, the correct fault was isolated. In some cases, i.e., for C_0^- and R_1^+, the slope had to be calculated, which took an additional amount of time. Note that the fault $R_{\rm bulb}^+$, which is an increase in the bulb resistance, and $Sw_2^{\rm off}$, which is a fault where Sw_2 is stuck off, could not be distinguished, which was predicted using diagnosability analysis. In the following, we step through the reasoning of the distributed diagnosers for two interesting scenarios. Note that, in each scenario, different loads are turned on in sequence before the fault is injected.

TABLE V
SIMULATION DIAGNOSIS RESULTS

Fault	DC Diagnoser			AC Diagnoser		
	t_d	t_i	Result	t_d	t_i	Result
$C_0^-, -1\%$	0.5	12.0	$\{C_0^-\}$	N/A	N/A	$\varnothing$
$R_1^+, +200\%$	1.5	10.5	$\{R_1^+\}$	N/A	N/A	$\varnothing$
$R_{L1}^+, +50\%$	0.0	4.5	$\{R_{L1}^+\}$	N/A	N/A	$\varnothing$
$R_{L1}^-, -50\%$	0.0	4.0	$\{R_{L1}^-\}$	N/A	N/A	$\varnothing$
Sw_1^{off}	0.0	3.0	$\{Sw_1^{off}\}$	N/A	N/A	$\varnothing$
$R_{bulb}^+, +50\%$	0.5	0.5	$\varnothing$	0.5	0.5	$\{R_{bulb}^+, Sw_2^{off}\}$
$R_{bulb}^-, -5\%$	N/A	N/A	$\varnothing$	1.5	1.5	$\{R_{bulb}^-\}$
$J_{fan}^-, -50\%$	N/A	N/A	$\varnothing$	0.0	0.0	$\{J_{fan}^-\}$
$B_{fan}^+, +50\%$	N/A	N/A	$\varnothing$	0.5	4.0	$\{B_{fan}^+\}$
Sw_2^{off}	0.5	0.5	$\varnothing$	0.5	0.5	$\{R_{bulb}^+, Sw_2^{off}\}$
Sw_3^{off}	0.5	0.5	$\varnothing$	0.5	0.5	$\{Sw_3^{off}\}$

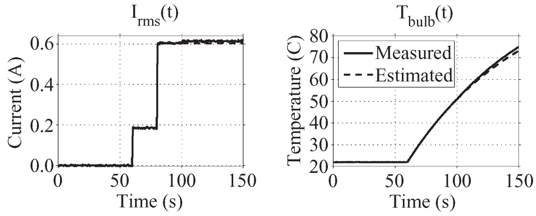

Fig. 10. $R_{\rm bulb}^-$ fault, where $R_{\rm bulb}$ decreases by 5%.

The first scenario consists of a 5% decrease in the bulb resistance $R_{\rm bulb}^-$ at 100 s. The relevant measurement plots corresponding to this scenario are shown in Fig. 10. This change results in an increase in the $T_{\rm bulb}$ at 101.5 s. Since only $R_{\rm bulb}^-$ is consistent with the observed increase in $T_{\rm bulb}$ (see Table IV), all other candidates are dropped by the ac diagnoser, and a unique candidate is obtained. The dc diagnoser later observes the increase in $I_{\rm rms}$, and since no faults in the dc subsystem can cause an increase in the rms inverter current, it eliminates all faults.

Next, we consider a 50% decrease in Load 1 resistance R_{L1}^- injected at 100 s. As shown in Fig. 11, this fault causes the Load 1 and battery currents to discontinuously increase. Both changes are detected at 100.0 s, resulting in the dc diagnoser generating $\{C_0^-, R_{L1}^-\}$ as the fault candidates. At 103.0 s, it is determined that neither measurement exhibited any discrete change behavior, which does not affect the current candidate list. At 104.0 s, it is determined that the change in I_B is a discontinuity and that V_B decreased. The fault C_0^- is dropped since it would cause instead a battery voltage increase, so R_{L1}^- is isolated as the true fault.

We have also studied in simulation the effect of fault magnitude and sensor noise on fault detection times and the fault isolation results. With R_{L1}^-, e.g., the fault was detected in less than 0.5 s, on average, for magnitudes of at least 5% with the selected levels of noise. Full details for faults in an extended dc subsystem can be found in [23].

C. Testbed Results

We have also performed experiments on the ADAPT testbed. Due to model uncertainty, the E values for some fault detectors had to be increased, resulting in slower detection and isolation times, compared with the simulation. For I_{L1} and $I_{\rm rms}$, $E = 0.1$; for $T_{\rm bulb}$ and ϕ, $E = 1.3$; and for $V_{\rm rms}$, $E = 1.0$.

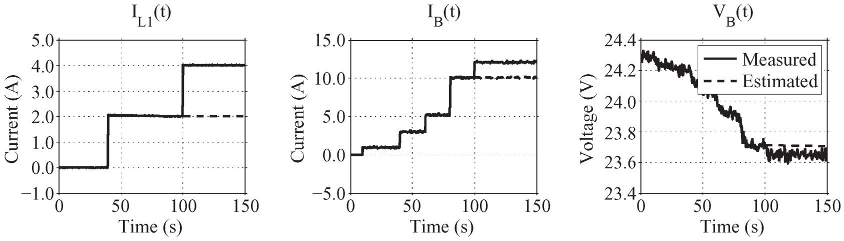

Fig. 11. R_{L1}^{-} fault, where R_{L1} decreases by 50%.

TABLE VI
TESTBED DIAGNOSIS RESULTS

Fault	DC Diagnoser			AC Diagnoser		
	t_d	t_i	Result	t_d	t_i	Result
$R_{L1}^{+}, +100\%$	0.5	8.0	$\{R_{L1}^{+}\}$	N/A	N/A	$\varnothing$
$R_{L1}^{-}, -33\%$	0.5	3.5	$\{R_{L1}^{-}\}$	N/A	N/A	$\varnothing$
$R_{bulb}^{+}, +50\%$	1.0	1.0	$\varnothing$	1.0	11.0	$\{R_{bulb}^{+}, Sw_2^{off}\}$
$R_{bulb}^{-}, -50\%$	2.5	2.5	$\varnothing$	2.5	2.5	$\{R_{bulb}^{-}, B_{fan}^{+}\}$
Sw_2^{off}	0.5	0.5	$\varnothing$	0.5	2.0	$\{R_{bulb}^{+}, Sw_2^{off}\}$
Sw_3^{off}	0.5	0.5	$\varnothing$	0.5	1.5	$\{Sw_3^{off}\}$

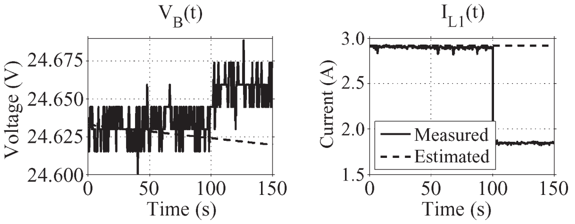

Fig. 12. R_{L1}^{+} fault, where R_{L1} increases by 100%.

Full results are provided in Table VI. Additional experiments for only dc components are provided in [23]. In most of the experiments, we achieved unique isolation when possible. The one exception is R_{bulb}^{-}, in which the changes in I_{rms} and T_{bulb} were too small to be detected. Future experiments will include our fault identification methods to resolve ambiguities remaining from the qualitative fault isolation stage. To demonstrate the diagnosis approach, we describe two scenarios: 1) a load fault and 2) a switch fault.

First, we consider a 100% increase in Load 1 resistance R_{L1}^{+} manually injected at 100.0 s in the mode with all loads on. The measured and estimated outputs are shown in Fig. 12. The increase in resistance causes a discontinuous drop in the current detected at 100.5 s. Since the slope has not yet been computed, the possible fault candidates are $\{R_1^{+}, R_{L1}^{+}, Sw_1^{\text{off}}\}$. At 102.5 s, the increase in V_B is detected, thus eliminating R_1^{+}. At 103.5 s, it is determined that I_{L1} did not go to zero, thus eliminating Sw_1^{off} and isolating R_{L1}^{+} as the true fault. None of the measurements in the ac subsystem deviate, so the ac diagnoser does not generate any candidates.

We next consider a discrete fault where Sw_2 turns off at 100.0 s. The relevant measured and estimated outputs are shown in Fig. 13. At 100.5 s, an increase in V_B is detected, so the dc diagnoser generates its initial candidates as $\{C_0^{-}, R_{L1}^{+}, R_{L_1}^{-}, Sw_1^{\text{off}}\}$. In addition, at 100.5 s, a decrease in I_{rms} is detected, so the initial candidates of the ac diagnoser are $\{R_{\text{bulb}}^{+}, Sw_3^{\text{off}}, Sw_2^{\text{off}}\}$. Because this measurement is known to the dc diagnoser, it can eliminate all of its faults and conclude that the fault must be in the ac subsystem. At 101.0 s, it is determined that the change in V_B was not a discontinuity, but the ac diagnosis remains unchanged. At 102.0 s, an increase in ϕ is detected, which reduces the fault set to $\{Sw_2^{\text{off}}, R_{\text{bulb}}^{+}\}$, which cannot further be distinguished, as explained earlier.

VIII. CONCLUSIONS

Applying model-based diagnosis techniques to real-world systems engenders many challenges, especially those associated with model development, system monitoring, and fault isolation. These challenges were faced when applying FACT to ADAPT. The modeling task is complicated, because details of component models are often unavailable, interactions between components are not fully documented, and sufficient data may not be available to estimate the parameters of the model. We faced these issues when modeling a number of components of the ADAPT system, especially the battery, inverter, fan, and pump. These modeling issues translate to challenges in system monitoring due to model uncertainty, as well as sensor noise and a lack of certain sensors that would simplify the diagnosis task. With a limited sensor set, fault isolation is also difficult, especially since ac and dc subsystems behave at vastly different time scales.

Other diagnosis approaches have also been applied to ADAPT. A convex optimization approach is employed in [24] but considers only faults in the dc subsystem. In [25], ADAPT is modeled using Bayesian networks on a quantized state space of low granularity. Methods using such quantizations may be unable to detect subtle faults, such as changes in battery capacitance, that a more detailed model would provide. More general approaches to hybrid systems diagnosis may also be applied, although most of them consider only discrete faults, such as the estimation-based approaches of [26] and [27], whereas our approach addresses both parametric and discrete faults. Notable exceptions that do address combined parametric and discrete fault diagnosis are the application-specific approach of [28], which models systems using hybrid automata, and the parity relations approach of [29], which does not easily extend to nonlinear systems with multiplicative faults. A related approach to [29] is that of [30], which also uses parity relations and incorporates discrete-event system techniques.

Perhaps most importantly, our FACT tools greatly facilitate synthesizing the different modules of the diagnosis system. However, setting the parameters of the observer and fault detectors is also a critical task for accurate system monitoring, avoiding false alarms, and correct symbol generation. Coming up with the right parameter values involves running a number

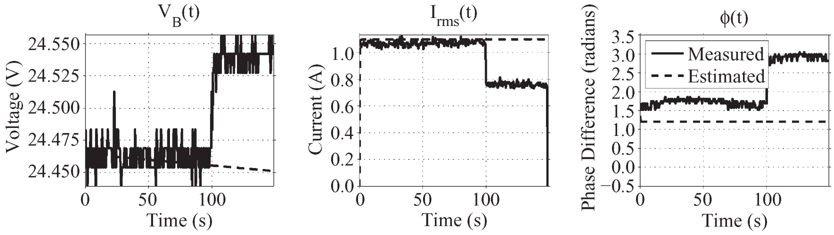

Fig. 13. Sw_2^{off} fault.

of systematic experiments. In some cases, it is hard to guarantee false alarm rates, because the nature of the modeling errors and measurement noise may be unknown. In our work, assuming Gaussian distributions and estimating the measurement noise variance online has worked well.

To manage specific challenges of ADAPT, we extended our traditional hybrid diagnosis approach to include steady-state analysis for ac systems, which provided us with fault signatures for ac and dc sensors. Based on the signatures, we performed diagnosability analysis of the system and designed distributed diagnosers for the heterogeneous dc and ac subsystems. In future work, we will perform additional online experiments to test our fault detection and symbol generation strategy for sensitivity to a variety of fault magnitudes under multiple sensor noise profiles. We are also improving our fault identification scheme for use on ADAPT and would like to provide confidence estimates when multiple candidates are retained after fault isolation. As part of ongoing work, we are also further extending our methods to deal with incipient faults [31] and multiple faults [23].

References

[1] P. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy—A survey and some new results," *Automatica*, vol. 26, no. 3, pp. 459–474, May 1990.

[2] J. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. New York: Marcel Dekker, 1998.

[3] R. J. Patton and J. Chen, "Observer-based fault detection and isolation: Robustness and applications," *Control Eng. Pract.*, vol. 5, no. 5, pp. 671–682, May 1997.

[4] R. Reiter, "A theory of diagnosis from first principles," in *Readings in Model-Based Diagnosis*. San Mateo, CA: Morgan Kaufmann, 1992, pp. 29–48.

[5] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Failure diagnosis using discrete-event models," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 2, pp. 105–124, Mar. 1996.

[6] M. Krysander, J. Aslund, and M. Nyberg, "An efficient algorithm for finding minimal overconstrained subsystems for model-based diagnosis," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 1, pp. 197–206, Jan. 2008.

[7] S. Poll, A. Patterson-Hine, J. Camisa, D. Nishikawa, L. Spirkovska, D. Garcia, D. Hall, C. Neukom, A. Sweet, S. Yentus, C. Lee, J. Ossenfort, I. Roychoudhury, M. Daigle, G. Biswas, X. Koutsoukos, and R. Lutz, "Evaluation, selection, and application of model-based diagnosis tools and approaches," in *Proc. AIAA Infotech@ Aerospace Conf. Exhib.*, Rohnert Park, CA, May 2007.

[8] E.-J. Manders, G. Biswas, J. Ramirez, N. Mahadevan, J. Wu, and S. Abdelwahed, "A model-integrated computing tool-suite for fault adaptive control," in *Proc. 15th Int. Workshop Principles Diagnosis*, Jun. 2004, pp. 137–142.

[9] S. Narasimhan and G. Biswas, "Model-based diagnosis of hybrid systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 3, pp. 348–361, May 2007.

[10] I. Roychoudhury, G. Biswas, and X. Koutsoukos, "Designing distributed diagnosers for complex continuous systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 2, pp. 277–290, Apr. 2009.

[11] P. Mosterman and G. Biswas, "Diagnosis of continuous valued systems in transient operating regions," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 29, no. 6, pp. 554–565, Nov. 1999.

[12] M. Daigle, X. Koutsoukos, and G. Biswas, "An event-based approach to integrated parametric and discrete fault diagnosis in hybrid systems," *Trans. Inst. Meas. Control*, DOI: 10.1177/0142331208097840. [Online]. Available: http://tim.sagepub.com/pap.dtl

[13] G. Biswas, G. Simon, N. Mahadevan, S. Narasimhan, J. Ramirez, and G. Karsai, "A robust method for hybrid diagnosis of complex systems," in *Proc. 5th Symp. Fault Detection, Supervision Safety Tech. Process.*, Jun. 2003, pp. 1125–1131.

[14] E.-J. Manders, S. Narasimhan, G. Biswas, and P. Mosterman, "A combined qualitative/quantitative approach for fault isolation in continuous dynamic systems," in *Proc. SafeProcess*, Budapest, Hungary, Jun. 2000, vol. 1, pp. 1074–1079.

[15] P. J. Mosterman and G. Biswas, "A theory of discontinuities in physical system models," *J. Franklin Inst.*, vol. 335, no. 3, pp. 401–439, Jan. 1998.

[16] D. C. Karnopp, D. L. Margolis, and R. C. Rosenberg, *Systems Dynamics: Modeling and Simulation of Mechatronic Systems*. New York: Wiley, 2000.

[17] I. Roychoudhury, M. Daigle, G. Biswas, and X. Koutsoukos, "Efficient simulation of hybrid systems: An application to electrical power distribution systems," in *Proc. 22nd Eur. Conf. Model. Simul.*, 2008, pp. 471–477.

[18] G. Karsai, J. Sztipanovits, A. Ledeczi, and T. Bapty, "Model-integrated development of embedded software," *Proc. IEEE*, vol. 91, no. 1, pp. 145–164, Jan. 2003.

[19] R. Kirk, *Statistics: An Introduction*. Fort Worth, TX: Harcourt Brace, 1999.

[20] R. Button and A. Chicatelli, "Electrical power system health management," in *Proc. 1st Int. Forum Integr. Syst. Health Eng. Manage. Aerosp.*, NAPA, CA, 2005.

[21] I. Roychoudhury, M. Daigle, G. Biswas, X. Koutsoukos, and P. J. Mosterman, "A method for efficient simulation of hybrid bond graphs," in *Proc. Int. Conf. Bond Graph Model. Simul.*, Jan. 2007, pp. 177–184.

[22] R. Debouk, S. Lafortune, and D. Teneketzis, "Coordinated decentralized protocols for failure diagnosis of discrete event systems," *Discr. Event Dyn. Syst.*, vol. 10, no. 1/2, pp. 33–86, Jan. 2000.

[23] M. Daigle, "A qualitative event-based approach to fault diagnosis of hybrid systems," Ph.D. dissertation, Vanderbilt Univ., Nashville, TN, 2008.

[24] D. Gorinevsky, S. Boyd, and S. Poll, "Estimation of faults in dc electrical power system," in *Proc. Amer. Control Conf.*, Jun. 2009, pp. 4334–4339.

[25] O. Mengshoel, A. Darwiche, K. Cascio, M. Chavira, S. Poll, and S. Uckun, "Diagnosing faults in electrical power systems of spacecraft and aircraft," in *Proc. 20th IAAI*, Jul. 2008, pp. 1699–1705.

[26] M. W. Hofbaur and B. C. Williams, "Hybrid estimation of complex systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 5, pp. 2178–2191, Oct. 2004.

[27] W. Wang, L. Li, D. Zhou, and K. Liu, "Robust state estimation and fault diagnosis for uncertain hybrid nonlinear systems," *Nonlinear Anal., Hybrid Syst.*, vol. 1, no. 1, pp. 2–15, Mar. 2007.

[28] F. Zhao, X. Koutsoukos, H. Haussecker, J. Reich, and P. Cheung, "Monitoring and fault diagnosis of hybrid systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 6, pp. 1225–1240, Dec. 2005.

[29] V. Cocquempot, T. El Mezyani, and M. Staroswiecki, "Fault detection and isolation for hybrid systems using structured parity residuals," in *Proc. 5th Asian Control Conf.*, 2004, pp. 1204–1212.

[30] M. Bayoudh, L. Travé-Massuyès, and X. Olive, "Hybrid systems diagnosis by coupling continuous and discrete event techniques," in *Proc. IFAC World Congr.*, 2008, pp. 7265–7270.

[31] I. Roychoudhury, G. Biswas, and X. Koutsoukos, "Distributed diagnosis of dynamic systems using dynamic Bayesian networks," in *Proc. 20th Int. Workshop Principles Diagnosis*, Jun. 2009, pp. 329–336.

Matthew J. Daigle (S'07–M'08) received the B.S. degree in computer science and computer and systems engineering from Rensselaer Polytechnic Institute, Troy, NY, in 2004 and the M.S. and Ph.D. degrees in computer science from Vanderbilt University, Nashville, TN, in 2006 and 2008, respectively.

From September 2004 to May 2008, he was a Graduate Research Assistant with the Institute for Software Integrated Systems and the Department of Electrical Engineering and Computer Science, Vanderbilt University. During the summers of 2006 and 2007, he was an intern with Mission Critical Technologies, Inc., at the NASA Ames Research Center, Moffett Field, CA. Since June 2008, he has been with the University of California, Santa Cruz, at the NASA Ames Research Center, where he is currently an Associate Research Scientist in the Intelligent Systems Division. His current research interests include physics-based modeling, model-based diagnosis and prognosis, and hybrid systems.

Dr. Daigle is a recipient of the 4.0 Award and Ricketts Prize from Rensselaer Polytechnic Institute, a University Graduate Fellowship from Vanderbilt University, and a Staff Recognition and Development Award from the University of California, Santa Cruz.

Indranil Roychoudhury (S'07–M'09) received the B.E. (Honors) degree in electrical and electronics engineering from the Birla Institute of Technology and Science, Pilani, Rajasthan, India, in 2004 and the M.S. and Ph.D. degrees in computer science from Vanderbilt University, Nashville, TN, in 2006 and 2009, respectively.

From September 2004 to July 2009, he was a Graduate Research Assistant with the Institute for Software Integrated Systems, Department of Electrical Engineering and Computer Science, Vanderbilt University. During the summers of 2006 and 2007, he was an intern with Mission Critical Technologies, Inc., at the NASA Ames Research Center, Moffett Field, CA. Since August 2009, he has been with SGT, Inc., Greenbelt, MD, at the NASA Ames Research Center, as a Computer Scientist and Postdoctor. His research interests include hybrid systems modeling, model-based diagnosis, distributed diagnosis, and Bayesian diagnosis of complex physical systems.

Gautam Biswas (S'78–M'82–SM'91) received the Ph.D. degree in computer science from Michigan State University, East Lansing, MI.

He is currently with Vanderbilt University, Nashville, TN, as a Professor of computer science and computer engineering with the Department of Electrical Engineering and Computer Science and a Senior Research Scientist with the Institute for Software Integrated Systems. He conducts research on intelligent systems with primary interests in hybrid modeling, simulation, and analysis of complex embedded systems, and their applications to diagnosis and fault-adaptive control. As part of this work, he has worked on fault-adaptive control of fuel transfer systems for aircraft and Advanced Life Support systems for NASA. He has also initiated new projects in distributed monitoring, diagnosis, and prognosis, and health management of complex systems. In other research projects, he is involved in developing simulation-based environments for learning and instruction, and planning and scheduling algorithms for distributed real-time environments. His research has been supported by funding from NASA, the National Science Foundation, the Defense Advanced Research Projects Agency, and the Office of Naval Research.

Dr. Biswas is a Senior Member of the IEEE Computer Society, Association for Computing Machinery, Association for the Advancement of Artificial Intelligence, and Sigma Xi Research Society. He is an Associate Editor for the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A. He has served on the program committee of a number of conferences.

Xenofon D. Koutsoukos (S'95–M'00–SM'07) received the Diploma in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, in 1993 and the M.S. degrees in electrical engineering and applied mathematics and the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 1998 and 2000, respectively.

From 2000 to 2002, he was a Member of Research Staff with the Xerox Palo Alto Research Center, Palo Alto, CA, working in the Embedded Collaborative Computing Area. Since 2002, he has been with Vanderbilt University, Nashville, TN, where he is currently an Associate Professor with the Department of Electrical Engineering and Computer Science and a Senior Research Scientist with the Institute for Software Integrated Systems. He currently serves as an Associate Editor for the *ACM Transactions on Sensor Networks* and *Modelling Simulation Practice and Theory*. His research interests include hybrid systems, real-time embedded systems, sensor networks, and cyber-physical systems.

Dr. Koutsoukos is a member of the Association for Computing Machinery. He was the recipient of the National Science Foundation CAREER Award in 2004.

Ann Patterson-Hine (M'88–SM'02) received the B.S. degree in mechanical engineering from The University of Alabama, Tuscaloosa, in 1981 and the M.S. and Ph.D. degrees in mechanical engineering from the University of Texas, Austin, in 1983 and 1988, respectively.

Since July 1988, she has been with NASA Ames Research Center, Moffett Field, CA, where she is currently the Tech Area Lead for Discovery and Systems Health in the Intelligent Systems Division. She has been the project leader for advanced technology demonstrations under the Next Generation Launch Technology Program and many of the program's predecessors including Reusable Launch Vehicle and Space Launch Initiative programs. She participated in the Shuttle Independent Assessment Team and Wire Integrity Pilot Study at Ames. Her research interests include the use of engineering models for model-based reasoning in advanced monitoring and diagnostic systems.

Dr. Patterson-Hine is an Associate Fellow of the Association for the Advancement of Artificial Intelligence. She is a registered Professional Engineer in the state of California.

Scott Poll received the B.S.E. degree in aerospace engineering from the University of Michigan, Ann Arbor, in 1994 and the M.S. degree in aeronautical engineering from the California Institute of Technology, Pasadena, in 1995.

He is currently a Research Engineer with the National Aeronautics and Space Administration (NASA) Ames Research Center, Moffett Field, CA, where he is the Deputy Lead for the Diagnostics and Prognostics Group in the Intelligent Systems Division. He is co-leading the evolution of a laboratory designed to enable the development, maturation, and benchmarking of diagnostic, prognostic, and decision technologies for system health management applications. He was previously the Associate Principal Investigator for Prognostics in the Integrated Vehicle Health Management Project in NASA's Aviation Safety Program.