A Discrete Event Approach to Diagnosis of Continuous Systems

Matthew Daigle and **Xenofon Koutsoukos** and **Gautam Biswas**
Institute for Software Integrated Systems (ISIS)
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, TN 37235, USA
matthew.j.daigle,xenofon.koutsoukos,gautam.biswas@vanderbilt.edu

Abstract

Fault detection and isolation is a key component of any safety-critical system. Although diagnosis methods based on discrete event systems have been recognized as a promising framework, they cannot be easily applied to systems with complex continuous dynamics. This paper presents a novel approach for discrete event system diagnosis of continuous systems based on a qualitative abstraction of the measurement deviations from the nominal behavior. We systematically derive a diagnosis model, provide diagnosability analysis, and design a diagnoser. Our results show that the proposed approach is easily applicable and can be used for online diagnosis of abrupt faults in continuous systems.

Introduction

Fault detection and isolation (FDI) is a key component of safety-critical systems. Faults and degradations need to be quickly identified so corrective actions can be taken and catastrophic situations can be avoided. FDI methodologies can be categorized along several dimensions, such as model-based vs. signal-driven, online vs. offline, and continuous vs. discrete. Discrete event system (DES) methods are an important framework for diagnosis because of the significance of event-driven models in safety-critical systems, a well-developed theory that allows for systematic construction of diagnostic systems, and the computational efficiency it provides to enable online diagnosis for large systems.

Existing DES diagnosis methods (Sampath *et al.* 1996; 1995; Zad, Kwong, & Wonham 2003; Jiang & Kumar 2004) are based on detailed, automata-based models capturing both the nominal and faulty behavior traces of the system. Discrete event models have formed the basis for developing many practical diagnosis applications (Sampath *et al.* 1996; Kurien, Koutsoukos, & Zhao 2002; Benveniste *et al.* 2003; Chandra, Huang, & Kumar 2003), however, it is not clear how to systematically apply them to develop diagnosers for systems with complex continuous dynamics. Abstracting continuous dynamics requires quantization of the state space, resulting in large, nondeterministic models (Lunze 2000; Koutsoukos *et al.* 2000). Further, even if it is reasonable to abstract the nominal continuous behavior, developing the event-based behavior trajectories for fault conditions is very challenging. Faults in continuous dynamic systems are represented by changes in the system parameters, and therefore, quantization techniques must consider a high-dimensional state space and often complex nonlinear dynamics.

This paper presents a novel approach for DES diagnosis of continuous systems based on a qualitative abstraction of the measurement deviations from the nominal behavior. We describe a systematic method for generating a discrete event model of the system representing the faulty behaviors. The approach is derived from the TRANSCEND (Mosterman & Biswas 1999) methodology, a model-based approach to qualitative fault diagnosis in continuous systems. Starting from the continuous system model and the TRANSCEND approach to diagnosis, we use the concept of fault signatures combined with measurement orderings to build a discrete event diagnoser for isolating single faults in the system.

Specifically, the contribution of the paper is threefold: (*i*) we systematically construct a labeled transition system capturing the *fault language*, which, for each fault, describes all possible sequences of measurement deviations, (*ii*) we analyze the diagnosability of the system and design an event-based diagnoser, and (*iii*) we describe an implementation that improves the computational efficiency of the diagnoser. Diagnosis of component faults in an electric circuit is used throughout the paper to illustrate the approach.

Our approach to continuous systems diagnosis exploits the qualitative form of the fault transient created by abrupt deviations in component parameter values as well as the temporal ordering of measurement deviations, thereby generating event sequences (Daigle, Koutsoukos, & Biswas 2005; 2006; 2007). Since DES methods diagnose system failures based on sequences of observed events, there is a direct link between our diagnosis approach and DES approaches. We clarify this connection by first describing related work. We then present our modeling and analysis approach, the design of the diagnoser, and a comparison of our approach to DES methods.

Related Work

We formulate our approach to diagnosis of continuous systems into an event-based framework. DES diagnosis methods are based on observing system events and making inferences about the system state. The basic idea is that the occurrence of a fault will generate a unique sequence of observable events that will establish the presence of the fault.

(Sampath *et al.* 1996; 1995) describes a modeling and diagnosis framework for systems in the DES framework. A diagnoser based on the system model functions as an extended observer that provides estimates of the system state under nonfaulty and faulty conditions. (Zad, Kwong, & Wonham 2003) use a state-based approach, so the diagnoser determines system condition, rather than which failure events have occurred. (Jiang & Kumar 2004) present diagnosis of DES based on linear-time temporal logic (LTL) specifications. In this method, an individual diagnoser is designed for each fault specification, rather than constructing a single diagnoser for the global system.

To apply DES diagnosis approaches to continuous systems, the system models must be abstracted in some way. One method is to create a timed DES model. Such models typically include an additional observable event representing the tick of a global clock (Chen & Provan 1997; Zad, Kwong, & Wonham 1999). Diagnosis of timed DES has been investigated in (Chen & Provan 1997) as an extension of (Sampath *et al.* 1996) and in (Zad, Kwong, & Wonham 1999) as an extension of (Zad, Kwong, & Wonham 2003). Alternatively, a timed automaton model of the system can be used for diagnosis (Tripakis 2002). The approach of (Lunze 2000) develops the abstracted timed DES model through quantization. The continuous state space is partitioned and events defined for crossings of those partitions.

Chronicles are another method of modeling timed event traces in systems. Chronicles are patterns of event traces that include temporal constraints. They represent the possible timed evolutions of the system behaviors. Chronicles represent direct symptom to fault knowledge, so are therefore very efficient for online diagnosis (Bibas *et al.* 1996; Cordier & Dousson 2000). As events occur in the system, they are matched against known chronicles to determine which faults are present.

A different event-based abstraction for continuous systems can take measurement deviations as events. (Puig *et al.* 2005a) describes different methods for including timing information for fault isolation. One method is to set time bounds for symptom appearance as in (Kościelny 1995; Kościelny & Zakroczymski 2000). Another method is to consider the order of symptom appearance in what is called a *dynamic fault signature matrix*. In (Puig *et al.* 2005b), a fault diagnosis algorithm is described which uses this type of information. Results illustrate that the diagnostic precision is improved over methods that do not use timing information, and diagnostic error is improved over other methods like (Kościelny 1995). (Bayoudh, Travеé-Massuyès, & Olive 2006) take an alternative approach where the events are taken to be changes in binary residual values. This information is used to reconfigure the system in a way that increases diagnostic precision, because in some modes a residual would be 0 due to a fault, an in others it would be 1.

Background

Our diagnosis approach is based on the TRANSCEND methodology (Mosterman & Biswas 1999), a model-based approach to continuous systems diagnosis. Faults are represented as persistent, abrupt parameter changes in the system, modeled as a bond graph (Karnopp, Margolis, & Rosenberg 2000). We assume only single faults are likely to occur. We use an observer based on the system model to track the system and produce estimates of nominal behavior. When faults occur, they produce transients causing measurements to deviate from nominal behavior. To achieve robustness to sensor noise and model uncertainty, these deviations are analyzed to determine if they are statistically significant (Biswas *et al.* 2003). Significant deviations are used as they occur to isolate faults in the system. The diagnosis model, the temporal causal graph (TCG), is derived from the system model. It captures the propagation of fault effects on measurements and, therefore, is used to compute predicted effects of faults on measurements. By comparing predicted and observed effects on measurements, we can obtain diagnoses.

Measurement deviations are represented as qualitative $\pm$ values (above, below nominal), and are predicted as *fault signatures* using the TCG (Mosterman & Biswas 1999). A fault signature represents the qualitative value of zeroth- through kth-order derivative changes on a measurement due to a fault occurrence. Because only magnitude and slope can be reliably measured, we condense the signatures to the magnitude change symbol and the first nonzero derivative change, e.g., `00-+-` becomes `0-`, and `+-+-+` becomes `+-`. We can do this because higher-order changes will eventually manifest as first-order changes, and only the first change on a measurement provides discriminatory information (Manders *et al.* 2000). Therefore, we represent a fault signature for measurement m as an element of the set $\Sigma_m \triangleq \{m^{+-}, m^{-+}, m^{0+}, m^{0-}\}$[1]. The superscript indicates the observed deviation. The first symbol represents the immediate direction of change (a discontinuity) at fault occurrence and the second symbol represents the slope of the change after fault occurrence.

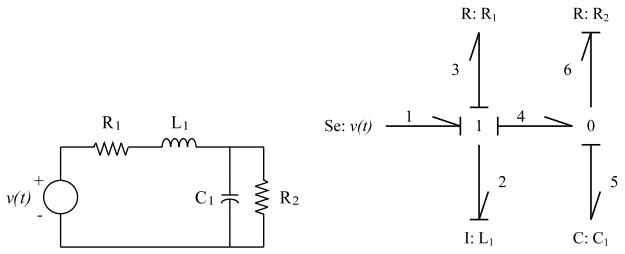
Definition 1 (Fault Signature)**.** *A* fault signature *for a fault f and measurement m is the qualitative effect of the occurrence of f on m, and is denoted by $\sigma_{f,m} \in \Sigma_{f,m}$, where $\Sigma_{f,m} \subseteq \Sigma_m$. We denote the set of all fault signatures for fault f as Σ_f.*

Relative measurement orderings define, with respect to a given fault, a partial order of measurement deviations, and are based on the intuition that some measurements deviate before others due to a fault. These are predicted using the TCG based on common temporal subpaths (Daigle, Koutsoukos, & Biswas 2005).

Definition 2 (Relative Measurement Ordering)**.** *Consider a fault f and measurements m_i and m_j. If f manifests in m_i before m_j then we define a* relative measurement ordering *between m_i and m_j for fault f, denoted as $m_i \prec_f m_j$. We denote the set of all measurement orderings for f as Ω_f.*

Throughout the paper we will illustrate the diagnosis methodology with a circuit example. Fig. 1(a) gives the

[1] In general, $\sigma_{f,m}$ may not be unique if the direction of change cannot be determined by qualitative propagation.



(a) Schematic. (b) Bond graph.

Figure 1: Circuit example.

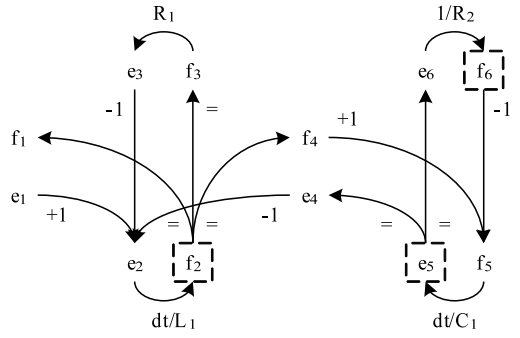


Figure 2: Temporal causal graph for the circuit. Measured variables are boxed.

schematic. We assume that our input voltage, $v(t)$, is constant and positive. The associated bond graph is given in Fig. 1(b). It models the elements of the circuit and the energy exchange between them (Karnopp, Margolis, & Rosenberg 2000). The derived TCG is given in Fig. 2. Relations between system variables are direct (+1) or inverse (-1) proportionality relations, component parameter values (e.g., R_1), or time-derivative effects (dt). The set of faults is assumed to be $F = \{R_1^+, R_1^-, R_2^+, R_2^-, C_1^+, C_1^-, L_1^+, L_1^-\}$, where the superscript indicates the direction of change of the parameter value. We define the measurement set as the current through L_1, the voltage across C_1, and the current through R_2, or $M = \{f_2, e_5, f_6\}$ in the bond graph model.

The fault signatures and relative measurement orderings for the circuit system are given in Table 1. For example, consider L_1^-. A decrease in L_1 will cause an immediate increase in f_2, because of the inverse relation implied in the TCG. Since all subsequent paths from f_2 to any other observed variable in the system contain some edge with a dt specifier (implying an integration), then deviations in these measurements will only be detected after f_2 deviates. Either e_5 or f_6 may deviate next. It cannot be determined which will deviate first because the path from e_5 to f_6 contains no integrals. The measurement deviations will not be abrupt because of the integration in the path from L_1 to the measurement, and the direction of change will be opposite that of f_2 because the -1 specifier in the path from f_2 to e_5 and f_6 indicates an inverse proportionality relationship.

Fault	f_2	e_5	f_6	Measurement Orderings
R_1^+	$0-$	$0-$	$0-$	$f_2 \prec e_5, f_2 \prec f_6$
R_1^-	$0+$	$0+$	$0+$	$f_2 \prec e_5, f_2 \prec f_6$
R_2^+	$0-$	$0+$	$-+$	$e_5 \prec f_2, f_6 \prec f_2, f_6 \prec e_5$
R_2^-	$0+$	$0-$	$+-$	$e_5 \prec f_2, f_6 \prec f_2, f_6 \prec e_5$
C_1^+	$0+$	$-+$	$-+$	$e_5 \prec f_2, f_6 \prec f_2$
C_1^-	$0-$	$+-$	$+-$	$e_5 \prec f_2, f_6 \prec f_2$
L_1^+	$-+$	$0-$	$0-$	$f_2 \prec e_5, f_2 \prec f_6$
L_1^-	$+-$	$0+$	$0+$	$f_2 \prec e_5, f_2 \prec f_6$

Table 1: Fault Signatures and Relative Measurement Orderings for the Circuit

Event-based Fault Modeling

We combine the notion of fault signatures and relative measurement orderings into an event-based framework. Essentially, for a specific fault, the combination of all fault signatures and relative measurement orderings yields all the possible ways a fault can manifest. We denote one of these possibilities as a *fault trace*.

Definition 3 (Fault Trace)**.** *A* fault trace *for a fault f, denoted by λ_f, is a string of length $\leq |M|$ that includes, for every $m \in M$ that will deviate due to f, a fault signature $\sigma_{f,m}$, such that the order of fault signatures satisfies Ω_f.*

Consider C_1^+. $\lambda_{C_1^+} = e_5^{-+} f_6^{-+} f_2^{0+}$ is a valid fault trace, but $\lambda_{C_1^+} = f_2^{0+} e_5^{-+} f_6^{-+}$ is not because the measurement deviation sequence does not satisfy $\Omega_{C_1^+}$. We group the set of all fault traces into a *fault language*, which can be represented concisely by a *labeled transition system* (LTS).

Definition 4 (Fault Language)**.** *The* fault language *of a fault f, denoted by L_f, is the set of all fault traces for f.*

Definition 5 (Labeled Transition System)**.** *A* labeled transition system *is a tuple $\mathcal{L} = (Q, q_o, \Sigma, \delta)$ such that: Q is a set of states, $q_o \in Q$ is an initial state, Σ is a set of labels, and $\delta \subseteq Q \times \Sigma \times Q$ is a transition relation.*

To systematically construct the LTS representation of a fault language, called a *fault model*, we can represent each fault signature and each relative measurement ordering as an LTS, and then compose all the information. Each fault signature $\sigma_{f,m}$ can be represented as an LTS, shown to the left of Fig. 3. It consists of only the single event corresponding to the fault signature[2]. Also, each relative measurement ordering, $m_i \prec_f m_j$, with associated signatures σ_{f,m_i} and σ_{f,m_j}, can be represented as an LTS, shown to the right of Fig. 3. It consists of the two associated signatures in the determined ordering.

Lemma 1. *The LTS representation of a fault language L_f for fault f, denoted by $\mathcal{L}_f$, is the synchronous product of the individual LTS for all $\sigma_{f,m} \in \Sigma_f$ and all $m_i \prec_f m_j \in \Omega_f$, where the alphabets for each LTS are taken to be the events contained in the LTS.*

[2]If $\sigma_{m,f}$ is not unique, multiple edges for each possibility are needed going from the first state of the LTS to the final state.

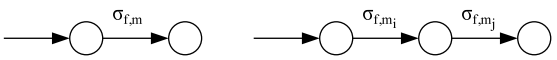


Figure 3: Fault signature LTS representation (left) and relative measurement ordering LTS representation (right).

Proof. Since the synchronous product must obey all individual ordering constraints and includes all measurement deviation events for the fault, it produces all valid measurement deviation sequences and no others. □

Lemma 2 (Distinguishability)**.** *A fault f_i is* distinguishable *from a fault f_j, denoted by $f_i \nsim f_j$, if $(\forall\, \lambda_{f_i} \in L_{f_i}, \lambda_{f_j} \in L_{f_j})\ (\neg\exists\, \lambda)\ \lambda_{f_i}\lambda = \lambda_{f_j}$.*

Proof. Two faults are distinguishable if it is not possible for them to manifest in the measurements in the same way. Since a fault language represents all possible measurement deviation sequences for a particular fault, if one fault exhibits a trace that is a substring of another fault, then the faults cannot be distinguished. Otherwise, they cannot manifest in the same way and are distinguishable. □

Depending on how they actually manifest in the system however, two faults which are indistinguishable may be discriminated if fault f_i occurs and manifests in a way that it is not possible for fault f_j to manifest, i.e, $\lambda_{f_i} \notin L_{f_j}$. Distinguishability is, therefore, a conservative notion. To design diagnosers, we look for the notion of *diagnosability*, based on the notion of distinguishability.

Lemma 3 (Diagnosability)**.** *A system is* single fault diagnosable *if $(\forall f_i, f_j \in F) f_i \neq f_j \implies f_i \nsim f_j$.*

Proof. A system is diagnosable if each possible fault trace is consistent with a unique fault. If two faults are distinguishable, then they cannot manifest in the same way. Therefore, if all pairs of faults are distinguishable, then a given fault trace cannot be consistent with the more than one fault. Therefore, the fault trace corresponds to a unique fault, so the system is diagnosable. □

Diagnoser Design

The notion of diagnosability is used in building correct diagnosers. To guarantee unique diagnosis of every fault, a system must be diagnosable. We now describe a method to systematically create such a diagnoser, but first, we define formally a *diagnosis* and a *diagnoser*.

Definition 6 (Diagnosis)**.** *A* diagnosis *$d \subseteq F$ is a set of faults consistent with the observations.*

Definition 7 (Diagnoser)**.** *A* diagnoser *is a tuple $\mathcal{D} = (Q, q_o, \Sigma, \delta, D, Y)$ such that: Q is a set of states, $q_o \in Q$ is an initial state, Σ is a set of labels, $\delta \subseteq Q \times \Sigma \times Q$ is a transition relation, $D \subseteq C$ is a set of diagnoses, and $Y : Q \to D$ is a diagnosis map.*

A diagnoser is a LTS extended by a set of diagnoses and a diagnosis map. Similar to the LTS of a fault, the labels correspond to measurement deviations. Associated with the states are diagnoses, i.e., the set of possible faults for the measurement deviations seen thus far.

Algorithm 1 $\mathcal{D} \leftarrow$ `CreateDiagnoser`$(\mathcal{D}_1, \mathcal{D}_2)$

```
Q ← ∅, δ ← ∅, D ← ∅, Σ ← Σ1 ∪ Σ2
q_o ← (q_o1, q_o2), Y(q_o) ← ∅, Q_pend ← {q_o}
while Q_pend ≠ ∅ do
  (q1, q2) ← pop(Q_pend)
  for all σ_m ∈ Σ do
    if m ∉ H((q1, q2)) then
      if δ1(q1, σ_m) and δ2(q2, σ_m) then
        q' ← (δ1(q1, σ_m), δ2(q2, σ_m))
        h ← Y(δ1(q1, σ_m)) ∪ Y(δ2(q2, σ_m))
      else if δ1(q1, σ_m) then
        q' ← (δ1(q1, σ_m), q2)
        h ← Y(δ1(q1, σ_m))
      else if δ2(q2, σ_m) then
        q' ← (q1, δ2(q2, σ_m))
        h ← Y(δ2(q2, σ_m))
      else
        q' ← ∅
        h ← ∅
      if q' ≠ ∅ then
        if Y((q1, q2)) = ∅ then
          d ← h
        else
          d ← Y((q1, q2)) ∩ h
        if d ≠ ∅ then
          Q ← Q ∪ {q'}
          H(q') ← H((q1, q2)) ∪ {m}
          δ((q1, q2), σ_m) ← q'
          D ← D ∪ {d}
          Y(q') ← d
          if q' ∉ Q_pend then
            push(Q_pend, q')
```

The diagnoser construction procedure is shown as Algorithm 1. It is described as combining two diagnosers, but can be easily be modified to combine k diagnosers simultaneously. Diagnosers are constructed by incrementally composing subdiagnosers, i.e., a diagnoser for a set of faults F_i is composed with a diagnoser for a set of faults F_j to create a new diagnoser for $F_i \cup F_j$. Initially, we begin with diagnosers for singleton fault sets. These are constructed using the individual fault models. For a single fault f, we augment $\mathcal{L}_f$ to form $\mathcal{D}_f$ by constructing the diagnosis map as mapping every state except the initial state to $\{f\}$. The initial state is mapped to the empty diagnosis, $\varnothing$, because until a measurement deviation is observed, we assume the system is operating nominally. The diagnosers corresponding to the individual faults of the circuit are shown in Fig. 4.

The construction algorithm operates by tracing paths in the two given diagnosers. If the same event label is available in both current states, then we advance in both machines, i.e., $(q_1, q_2) \xrightarrow{\sigma} (\delta(q_1, \sigma), \delta(q_2, \sigma))$. Otherwise, we advance in only one, e.g., if σ can only be taken from q_1, then , $(q_1, q_2) \xrightarrow{\sigma} (\delta(q_1, \sigma), q_2)$. However, if the measurement associated with σ has already deviated along the cur-

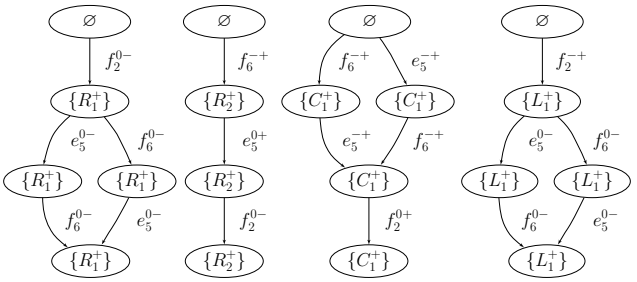


Figure 4: Diagnosers for the individual faults of the circuit. The diagnosers for decreases in the parameter values the same except for a reversal in the signs.

rent path (tracked using H), $\delta((q_1, q_2), \sigma)$ is set to $\varnothing$, because measurement deviations are only detected once per measurement. This also occurs if the computed diagnosis for the new state, d, is empty, because this means the current sequence of measurement deviations is inconsistent with the single fault assumption.

The diagnosis for the new state is formed by composing the current diagnosis with the hypothesis set. The hypothesis set, h, is the set of faults consistent with the current event. It is formed as the union of the diagnoses of the diagnoser states advanced to via σ. The new diagnosis for the composed diagnoser state is constructed as the intersection of the current diagnosis and the hypothesis set. For example, if $\{f_i, f_j\}$ is the current diagnosis and the hypothesis set is $\{f_i\}$ then the new diagnosis is $\{f_i\}$, which means that only f_i is consistent with the current event sequence.

The final composed diagnoser for the circuit is illustrated in Fig. 5. For example, consider the fault trace $f_6^{-+} e_5^{0+} f_2^{0-}$. For f_6^{-+} occurring as the first measurement deviation, only C_1^+ or R_2^+ could have occurred, given the known fault signatures and relative measurement orderings. Therefore, the new diagnosis is $\{C_1^+, R_2^+\}$. For e_5^{0+} occurring next, of our current faults, only R_2^+ is consistent, therefore our new diagnosis is the intersection of $\{C_1^+, R_2^+\}$ and $\{R_2^+\}$, which is $\{R_2^+\}$. At this point we obtain a unique fault. The only possible measurement deviation from here is f_2^{0-} which must be consistent still with $\{R_2^+\}$.

Theorem 1. *The diagnoser constructed by Algorithm 1 for fault sets F_1 and F_2 represents all valid single fault traces for the faults in F_1 and F_2 and associates correct diagnoses with the states.*

Proof. By definition, the diagnoser for a single fault f is correct because it represents L_f, so represents all possible fault traces of f, and every state (except the initial state) of $\mathcal{L}_f$ is consistent with f occurring. Assume that diagnosers $\mathcal{D}_1$ and $\mathcal{D}_2$ are correct. Then they represent all possible fault traces for fault sets F_1 and F_2, respectively. At the initial state, if an event σ happens which can only happen in one of the diagnosers, $\mathcal{D}_i$, then the diagnosis is $Y_i(\delta(q_{oi}, \sigma))$, because it must be consistent with faults in F_i that are consistent with σ. If σ can occur in both diagnosers, then the diagnosis is $Y_1(\delta_1(q_{o1}, \sigma)) \cup Y_2(\delta_2(q_{o2}, \sigma))$ because either a fault in F_1 occurred or a fault in F_2 occurred, and the diagnosis must be consistent with any fault in $F_1 \cup F_2$ consistent with σ. Assume that for a given $(q_1, q_2) \neq q_o$ the diagnoses are correct for the event sequences leading up to (q_1, q_2). Then if an event σ happens which can only happen in one of the diagnosers, $\mathcal{D}_i$, then the diagnosis is $Y((q_1, q_2)) \cap Y_i(\delta_i(q_i, \sigma))$, because it must consistent with the previous diagnosis faults in F_i consistent with σ. If σ can occur in both diagnosers, then the diagnosis is $Y((q_1, q_2)) \cap (Y_1(\delta_1(q_1, \sigma)) \cup Y_2(\delta_2(q_2, \sigma)))$ because it must be consistent with the previous diagnosis and faults in $F_1 \cup F_2$ consistent with σ. Therefore, for any state q, $\delta(q, \sigma)$ has a correct diagnosis. So, for any two diagnosers, the resulting diagnoser is correct. □

The diagnoser for the circuit example, shown in Fig. 5, illustrates certain properties of our approach. Since all the leaves have diagnoses with a unique fault, then the system is diagnosable. Any possible sequence of measurement deviations corresponding to a single fault occurring are captured in the diagnoser, and lead to unique diagnoses, therefore the system is diagnosable. We can also see that a unique diagnosis is obtained after only two of the three measurements deviate, therefore one measurement is redundant for single fault diagnosis of the selected faults.

Online Diagnoser Implementation

This event-based diagnosis framework leads to three different implementations of the onlnie diagnosis approach that trade off space and time complexity.

Implementation as a global LTS Time complexity is in favor of the precomputed global diagnoser (Fig. 5). It needs only to wait for measurement deviations to occur, transition to the next state, and output the current diagnosis associated with the state. Using appropriate data structures, these operations can be achieved in constant time.

The complete diagnoser has, in the worst case, $O(|M|!)$ possible fault traces, and thus $O(|M|!)$ states. Therefore this approach will not be space-efficient, in general. If many temporal orderings exist, then the number of possible fault traces reduces significantly, and the global diagnoser approach may be feasible.

Implementation as an LTS for each fault In this approach, we only precompute the individual fault diagnosers (Fig. 4). Each fault has $O(|M|!)$ possible fault traces, but if there are many temporal orderings, this may also be reduced for many faults.

For online diagnosis, each diagnoser is traced simultaneously. When a diagnoser becomes blocked, i.e., there is no available event to to take from the current state, then it is no longer tracked, because it is no longer consistent with the observed measurement deviations. The candidate set is formed by taking the union of the faults in the current states of each active diagnoser, i.e., those faults that are still consistent with the observed measurement deviations. This operation is simply $O(|F|)$ in time.

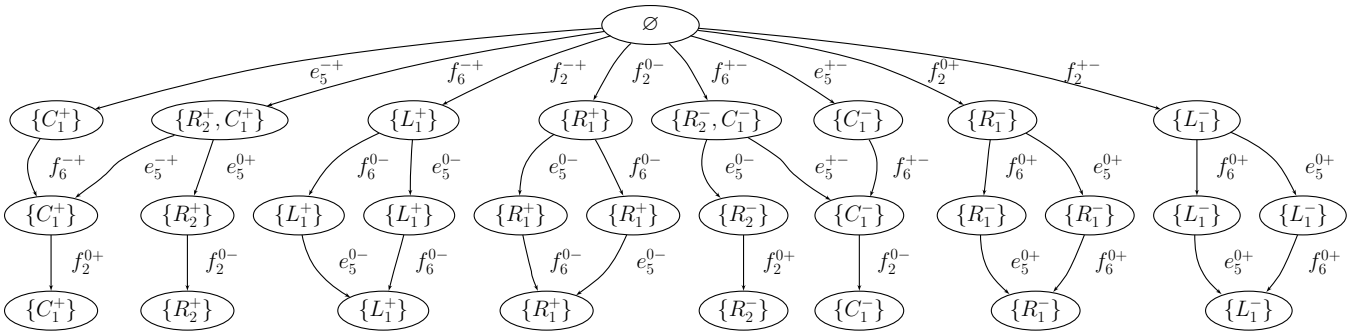


Figure 5: Single fault diagnoser for the circuit.

Implementation without explicit event fault models If the faults are very temporally constrained, then any of the two above approaches should be both space-efficient and time-efficient. If few orderings are available, then the diagnosers approach size $O(|M|!)$, therefore these approaches may not be feasible given the space requirements of the system. For diagnosis without using LTS-based diagnosers, we store only the fault signatures and relative measurement orderings for each fault (Table 1), requiring $O(|F||M|^2)$ space.

Given a current diagnosis of d_{i-1} and an event σ_i occurring, we can check which faults are consistent with σ_i. The hypothesis set h_i consists of those faults. If $i = 1$, then the new diagnosis d_i is simply h_i. Otherwise, the new diagnosis must be consistent with d_{i-1} and with the new information, i.e., $d_i = d_{i-1} \cap h_i$. Therefore, given d_{i-1}, the new diagnosis can be computed simply as the subset of faults in d_{i-1} consistent with σ_i. This corresponds to only constructing the *path* of the global diagnoser relating to the particular fault trace we are observing.

For online diagnosis, we form the hypothesis set corresponding to the current measurement deviation by looking through the fault signatures and measurement orderings, thus taking $O(|F||M|^2)$ time. We then compute the new diagnosis, which is a function of the size of the current diagnosis and the current hypothesis set. In the worst case the hypothesis set consists of all faults, so it is $|F|$ in size. A diagnosis can be as large as $|F|$ also. The intersection of the diagnosis and hypothesis set then takes at worst $O(|F|)$ time. In practice, this time complexity is reduced because as measurements deviate, less faults are being considered.

Discussion and Comparison

Though diagnosis using ordered event sequences is performed similarly in all approaches, the main contrast between existing methods and our approach is the abstraction used to generate the DES models. Most of the traditional DES approaches (Sampath *et al.* 1996; 1995; Zad, Kwong, & Wonham 2003; Jiang & Kumar 2004; Rozé & Cordier 2002; Baroni *et al.* 1999; Benveniste *et al.* 2003; Chen & Provan 1997; Zad, Kwong, & Wonham 1999) assume models created by human experts, and others assume subsystem models created by experts that are then composed to form the global model. The DES model and all its faulty behavior is assumed to be given. To derive a discrete event model for a continuous system, the continuous dynamics must be abstracted in some way. In quantization approaches, e.g., (Lunze 2000), the state space is quantized. This results in several problems. First, the model is, except in trivial cases, inherently nondeterministic, which degrades the performance and increases the computational requirements of diagnosis algorithms. Second, the resulting model is very large. The finer-grained the quantization and the greater the range of possible inputs, the larger and more complex the model will be. To use the quantization approach, faults have to be quantized as well, according to their magnitude and other characteristics. In addition, if faults are possible at any state of the system (as is usually the case), then the DES model becomes larger still, and the number of states explodes.

In our approach, however, faults are represented as parameter changes in the nominal model of the system. As a result, the system model represents both nominal and faulty behavior in a very concise way. From this model (the bond graph model), we can systematically derive the diagnosis model (Mosterman & Biswas 1999), i.e., the TCG, generate fault signatures and measurement orderings, and extract from this information a DES model of the system with respect to faulty behavior. This greatly reduces the burden of the modeling task, as well as providing a systematic framework for deriving the faulty behavior. Approaches such as (Puig *et al.* 2005a) or chronicles do not provide a way to systematically obtain this information from a system model except for very specific applications (Dousson 1996). Additionally, our approach is not dependent on fault magnitude because we are only concerned with the qualitative form of the measurement deviations.

Our approach can be viewed as a *qualitative* abstraction of the observed behavior from the nominal behavior. We model only the faulty behavior relevant to diagnosis, so there are three qualitative states for each measurement: *above nominal*, *at nominal*, and *below nominal*. Measurement deviations directly indicate the presence of a fault. The only state of the diagnoser modeling nominal behavior is the initial state, in which no measurement deviations have been observed. Our nominal behavior is defined through an observer (Manders *et al.* 2000), which is the best way to track a con-

tinuous system with noise. Therefore, faults can be detected very quickly, unlike in quantization approaches, where the fault detection time will depend on the level of quantization, whereas in our approach, fault detection time is a function of noise only. The tasks of tracking nominal behavior and fault isolation are separated so that the diagnoser is concerned only with faulty behavior.

In a way, our abstraction can be viewed as qualitative deviation models (Struss 2004). However, the TCG represents this information in a concise manner, and reasoning with the TCG is very efficient. Further, the TCG is based directly on the system behavior defined by the continuous state equations, so captures complex dynamics and interactions. Generating the predictions in the form of qualitative deviations from the TCG is automatic. We also include discontinuity detection for increased discriminatory ability which is not taken into account by traditional qualitative deviation models.

Other continuous diagnosis approaches that use temporal information (Kościelny 1995; Kościelny & Zakroczymski 2000; Puig *et al.* 2005a; 2005b; Bayoudh, Travеé-Massuyès, & Olive 2006) are based on analytical redundancy methods. These methods are difficult to apply to nonlinear systems and multiplicative faults. In addition, they do not consider the sign of the residual or whether a discontinuity was present in the signal to help isolate faults. The use of time bounds in some of these approaches and in chronicle approaches (Bibas *et al.* 1996; Cordier & Dousson 2000) is also difficult to use for continuous systems. It requires significant analysis to obtain the time bounds, which are often conservative, and assume bounds on fault magnitude, which cannot be made in general.

Because we are working in an event-based framework, the notions of fault traces, fault languages, distinguishability, and diagnosability bear a resemblance to those defined in the DES framework. The notion of a fault trace is similar to the notion of a fault signature defined for DES in (Cordier, Travé-Massuyès, & Pucel 2006), where it is defined as a string (of finite or infinite length) that contains a fault event. Diagnosability can then be defined in terms of ensuring no two faults have the same signature (in our case, the same trace). This is the situation in any modeling framework, because in general, faults can only be distinguished if they manifest in different ways, in whatever way that is represented by the model. Our approach separates out the fault effects and analyzes them separately. The individual fault diagnosers generated by our approach are also similar to chronicles and the individual diagnoser of (Jiang & Kumar 2004). The global diagnoser bears resemblance to (Sampath *et al.* 1996; Zad, Kwong, & Wonham 2003).

Another advantage of operating within an event-based framework is that the model created by our qualitative abstraction approach can be used with any of the other DES diagnosis approaches. However, since in our approach, we essentially abstract out events corresponding to nominal behavior, we obtain a direct mapping of finite event sequences to faults. Consequently, a simpler diagnosis approach than those defined for general DES models can be employed. Because of this, the diagnoser is simpler, and also, does not need to be computed at design time, which improves considerably space-efficiency, because the particular path corresponding to the given measurement deviations can be constructed online efficiently.

Conclusions

We have presented an event-based approach to diagnosis of single abrupt faults in continuous systems. We use a qualitative abstraction from nominal behavior. The approach results in systematic generation of event-based fault models and diagnosers, based on qualitative fault signatures and temporal orderings of measurement deviations. Current and future work is addressing multiple fault diagnosis and extending our hybrid systems diagnosis algorithms (Narasimhan & Biswas 2007) under this framework.

Acknowledgments

This work was supported in part by NSF grant CNS-0615214, NASA USRA grant 08020-013, and NASA NRA grant NNX07AD12A.

References

Baroni, P.; Lamperti, G.; Pogliano, P.; and Zanella, M. 1999. Diagnosis of large active systems. *Artificial Intelligence* 110(1):135–183.

Bayoudh, M.; Travеé-Massuyès, L.; and Olive, X. 2006. Hybrid systems diagnosability by abstracting faulty continuous dynamics. In *Proceedings of the 17th International Principles of Diagnosis Workshop*, 9–15.

Benveniste, A.; Fabre, E.; Haar, S.; and Jard, C. 2003. Diagnosis of asynchronous discrete-event systems: A net unfolding approach. *IEEE Transactions on Automatic Control* 48(5):714–727.

Bibas, S.; Cordier, M.-O.; Dague, P.; Dousson, C.; Lvy, F.; and Roz, L. 1996. Alarm driven supervision for telecommunication network: I – off-line scenarios generation. *Annales des Telecommunications* 51(910):493–500.

Biswas, G.; Simon, G.; Mahadevan, N.; Narasimhan, S.; Ramirez, J.; and Karsai, G. 2003. A robust method for hybrid diagnosis of complex systems. In *Proceedings of the 5th Symposium on Fault Detection, Supervision and Safety for Technical Processes*, 1125–1131.

Chandra, V.; Huang, Z.; and Kumar, R. 2003. Automated control synthesis for an assembly line using discrete event system control theory. *IEEE Trans. on Systems, Man and Cybernetics, Part C* 33(2):284–289.

Chen, Y.-L., and Provan, G. 1997. Modeling and diagnosis of timed discrete event systems – a factory automation example. In *Proc. of the American Control Conf.*, 31–36.

Cordier, M.-O., and Dousson, C. 2000. Alarm driven monitoring based on chronicles. In *Proceedings of the 4th Symposium on Fault Detection Supervision and Safety for Technical Processes (Safeprocess 2000)*, 286–291.

Cordier, M.-O.; Travé-Massuyès, L.; and Pucel, X. 2006. Comparing diagnosability in continuous and discrete-event

systems. In *Proceedings of the 17th International Workshop on Principles of Diagnosis (DX-06)*, 55–60.

Daigle, M.; Koutsoukos, X.; and Biswas, G. 2005. Relative measurement orderings in diagnosis of distributed physical systems. In *43rd Annual Allerton Conference on Communication, Control, and Computing*, 1707–1716.

Daigle, M.; Koutsoukos, X.; and Biswas, G. 2006. Distributed diagnosis of coupled mobile robots. In *Proceedings 2006 IEEE International Conference on Robotics and Automation*, 3787–3794.

Daigle, M. J.; Koutsoukos, X. D.; and Biswas, G. 2007. Distributed diagnosis in formations of mobile robots. *IEEE Transactions on Robotics* 23(2):353–369.

Dousson, C. 1996. Alarm driven supervision for telecommunication network: Ii – on-line chronicle recognition. *Annales des Telecommunications* 51(910):501–508.

Jiang, S., and Kumar, R. 2004. Failure diagnosis of discrete-event systems with linear-time temporal logic specifications. *IEEE Transactions on Automatic Control* 49(6):934–945.

Karnopp, D. C.; Margolis, D. L.; and Rosenberg, R. C. 2000. *Systems Dynamics: Modeling and Simulation of Mechatronic Systems*. New York: John Wiley & Sons, Inc.

Kościelny, J. M., and Zakroczymski, K. 2000. Fault isolation method based on time sequences of symptom appearance. In *Proceedings of IFAC SafeProcess 2000*.

Kościelny, J. M. 1995. Fault isolation in industrial processes by the dynamic table of states method. *Automatica* 31(5):747–753.

Koutsoukos, X.; Antsaklis, P.; Stiver, J.; and Lemmon, M. 2000. Supervisory control of hybrid systems. *Proceedings of IEEE* 88(7):1026–1049.

Kurien, J.; Koutsoukos, X.; and Zhao, F. 2002. Distributed diagnosis of networked embedded systems. In *Proceedings of the 13th International Workshop on Principles of Diagnosis (DX-02)*, 179–188.

Lunze, J. 2000. Diagnosis of quantized systems based on a timed discrete-event model. *IEEE Transactions on Systems, Man, and Cybernetics, Part A* 30(3):322–335.

Manders, E.-J.; Narasimhan, S.; Biswas, G.; and Mosterman, P. 2000. A combined qualitative/quantitative approach for fault isolation in continuous dynamic systems. In *SafeProcess 2000*, volume 1, 1074–1079.

Mosterman, P., and Biswas, G. 1999. Diagnosis of continuous valued systems in transient operating regions. *IEEE Transactions on Systems, Man and Cybernetics, Part A* 29(6):554–565.

Narasimhan, S., and Biswas, G. 2007. Model-based diagnosis of hybrid systems. *IEEE Transactions on Systems, Man and Cybernetics, Part A* 37(3):348–361.

Puig, V.; Quevedo, J.; Escobet, T.; and Pulido, B. 2005a. On the integration of fault detection and isolation in model-based fault diagnosis. In *Proceedings of the 16th International Workshop on Principles of Diagnosis (DX-05)*, 227–232.

Puig, V.; Schmid, F.; Quevedo, J.; and Pulido, B. 2005b. A new fault diagnosis algorithm that improves the integration of fault detection and isolation. In *Proceedings of the 44th IEEE Conference on Decision and Control*, 3809–3814.

Rozé, L., and Cordier, M.-O. 2002. Diagnosing discrete-event systems: Extending the "diagnoser approach" to deal with telecommunication networks. *Discrete Event Dynamic Systems: Theory and Applications* 12:43–81.

Sampath, M.; Sengupta, R.; Lafortune, S.; Sinnamohideen, K.; and Teneketzis, D. 1995. Diagnosability of discrete-event systems. *IEEE Transactions on Automatic Control* 40(9):1555–1575.

Sampath, M.; Sengupta, R.; Lafortune, S.; Sinnamohideen, K.; and Teneketzis, D. 1996. Failure diagnosis using discrete-event models. *IEEE Transactions on Control Systems Technology* 4(2):105–124.

Struss, P. 2004. Deviation models revisited. In *Working Papers of the 18th International Workshop on Qualitative Reasoning*.

Tripakis, S. 2002. Fault diagnosis for timed automata. In *Formal Techniques in Real Time and Fault Tolerant Systems (FTRTFT02)*, volume 2469 of *Lecture Notes in Computer Science*, 205–221. Springer.

Zad, S. H.; Kwong, R. H.; and Wonham, W. M. 1999. Fault diagnosis in timed discrete-event systems. In *Proc. of the 38th Conference on Decision and Control*, 1756–1761.

Zad, S. H.; Kwong, R.; and Wonham, W. 2003. Fault diagnosis in discrete-event systems: Framework and model reduction. *IEEE Transactions on Automatic Control* 48(7):1199–1212.