Designing Distributed Diagnosers for Complex Continuous Systems

Indranil Roychoudhury, *Student Member, IEEE,* Gautam Biswas, *Senior Member, IEEE,* and Xenofon Koutsoukos, *Senior Member, IEEE*

***Abstract*—Wear and tear from sustained operations cause systems to degrade and develop faults. Online fault diagnosis schemes are necessary to ensure safe operation and avoid catastrophic situations, but centralized diagnosis approaches have large memory and communication requirements, scale poorly, and create single points of failure. To overcome these problems, we propose an online, distributed, model-based diagnosis scheme for isolating abrupt faults in large continuous systems. This paper presents two algorithms for designing the local diagnosers and analyzes their time and space complexity. The first algorithm assumes the subsystem structure is known and constructs a local diagnoser for each subsystem. The second algorithm creates a partition structure and local diagnosers simultaneously. We demonstrate the effectiveness of our approach by applying it to the Advanced Water Recovery System developed at the NASA Johnson Space Center.**

***Note to Practitioners*— Fault detection, isolation, and recovery approaches are important for maintaining performance and safety in large safety-critical systems, such as the Advanced Life Support (ALS) System for future long-duration NASA manned missions that we present in this paper. These systems consist of a number of complex, interacting, spatially-distributed subsystems. Centralized model-based diagnosis approaches are expensive in memory and communication requirements, and they create single points of failure. Previous distributed diagnosis approaches apply to discrete event system models, but these approaches become computationally intractable when applied to complex continuous systems. This paper develops a systematic model-based approach to distributing the diagnosis task by designing multiple diagnosers that operate independently and generate globally correct diagnoses. We present a complete approach that includes a topological modeling scheme for constructing the dynamic system models, algorithms for constructing the distributed diagnosers, and a systematic methodology for deriving efficient subsystem diagnosis using these diagnosers. We then demonstrate the applicability of this approach to the ALS system.**

***Index Terms*—Model-based diagnosis, distributed diagnosis, continuous systems**

I. Introduction

Modern day engineered systems are a product of careful design, manufacturing, testing and validation before deployment. This reduces the likelihood of system failures, but degradation and faults in system components still occur due to wear and tear from sustained operations. Early detection and isolation of faults is the key to maintaining system performance, ensuring system safety, and increasing system life. Traditionally, the fault diagnosis task has been performed during maintenance operations, using test results and alarm signals to isolate faults in system components. For present-day, safety-and-mission-critical systems, it is imperative to monitor system behavior and performance during operation, so that system control and operation can adapt to changes and avoid catastrophic failures.

Most model-based diagnosis schemes for continuous systems are centralized with one monolithic diagnoser that is based on a global system model and all the available system measurements [1], [2]. Centralized model-based diagnosis schemes have several drawbacks. They are *expensive* in memory and computational requirements. Reliable transmission of measurements to a centralized computer may incur high *costs* for shielding and protection of the cables to maintain signal quality, especially in harsh environments. These approaches *scale poorly* for continuous systems as changes in the system configuration and components may cause significant changes in the system's dynamic behavior, requiring the diagnoser to be redesigned. A centralized approach also creates a *single point of failure*. A glitch or failure in the supporting computational units may disable the entire diagnosis system.

The drawbacks of centralized diagnosis schemes motivate the need for distributed diagnosis approaches, where the diagnosis task is broken down into subtasks and executed on separate processors. The distributed diagnosis approach fits well with present day embedded systems architectures, where each subsystem has associated local processors, memory, and sensors for monitoring and control of that subsystem (e.g., electronic control units in aircrafts).

In this paper, we develop a distributed, model-based fault diagnosis scheme for continuous systems, where the local diagnosers generate globally correct diagnosis results, with no coordination, and with minimal exchange of information amongst themselves. Our bond graph-based approach provides a unified framework for modeling physical processes, sensors, and actuators for nonlinear electrical, thermal, mechanical, and hydraulic systems [3]. The diagnostic methodology builds on our previous work, Transcend, a centralized, observer-based qualitative approach to diagnosis [4], [5]. We propose two algorithms to design the distributed diagnosers. The first algorithm uses predefined subsystem structure to generate, for each subsystem, a local diagnoser that produces globally correct diagnosis results with minimal exchange of information with the other local diagnosers. The second algorithm

Manuscript received April 16, 2007; revised September 28, 2007. This work was supported in part by the National Science Foundation under Grants CNS-0452067 and CNS-0615214; and NASA NRA NNX07AD12A.

The authors are with the Institute for Software Integrated Systems and the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, 37235, USA (email: indranil.roychoudhury@vanderbilt.edu; gautam.biswas@vanderbilt.edu; xenofon.koutsoukos@vanderbilt.edu).

constructs the system partition structure and local diagnosers simultaneously. The set of diagnosers do not exchange any information between themselves to produce globally correct diagnosis results. We apply both algorithms to a complex, real-world system, the Advanced Water Recovery System developed at the NASA Johnson Space Center [6]. The experimental results demonstrate the computational efficiency and reduction in communication overhead achieved by our distributed diagnosis approach.

Our approach assumes faults are persistent, abrupt, and non-catastrophic. The abrupt fault assumption is a good mathematical approximation for many practical faults [2]. We make the single fault assumption since simultaneous multiple fault occurrences are unlikely. Extensions to TRANSCEND for multiple fault and incipient fault diagnoses are discussed in [7] and [8], respectively. These extensions are not used in this paper.

The rest of this paper is organized as follows. Section II presents related work. Section III presents an overview of the TRANSCEND approach. Formulation of the distributed diagnosis problem is presented in Section IV, and algorithms for designing distributed diagnosers are described in Section V. In Section VI, we demonstrate the effectiveness of our design approach through an experimental case study of the Advanced Water Recovery System. Section VII concludes the paper.

II. RELATED WORK

Model-based diagnosis approaches can be broadly classified into *centralized*, *decentralized*, and *distributed* schemes (e.g., [2], [9]–[11]). Centralized schemes (e.g., [2]), construct a single diagnoser from a global system model. Decentralized schemes, such as [12], use a global system model but distribute the diagnosis computations among several local diagnosers. The local diagnosis decisions based on a subset of observations are communicated to other diagnosers, or to a central coordinator, which use the global model to generate globally consistent solutions. Distributed diagnosis approaches use subsystem models and assume the global model is unknown [13]–[15]. Local diagnosers for each subsystem communicate their diagnosis results to each other to arrive at the global solution.

Most decentralized and distributed diagnosis algorithms have been developed in the discrete-event framework [12]–[17]. In [12], the authors discuss three coordinated decentralized protocols for diagnosis that extend the centralized diagnosis method developed in [1]. Each local diagnoser is built from the global system model and uses only a subset of observable events. Coordination is necessary in the first and second protocols to generate the correct diagnosis result, but the third protocol generates correct results without a coordinator. All three protocols, under certain assumptions, produce the same results as a centralized diagnoser.

The approaches presented in [16] and [17] avoid coordination between local diagnosers by representing the system as a network of communicating finite state machines. First, the observable events for each subsystem are used to generate the individual subsystem diagnoses. Then, the subsystem diagnoses are merged to generate the global diagnosis result. The offline approach presented in [16] assumes all observable events are received in the same order that they were transmitted. The online approach described in [17] achieves efficiency by avoiding merge operations for independent subsystems. Its incremental algorithm does not assume the ordering of observations is preserved.

In [13], the authors describe an approach where each local diagnoser generates a set of local diagnoses, and then communicates with its neighbors to reduce the number of hypotheses. The graph of constraints between the fault hypotheses and the observations is partitioned to minimize communication between local diagnosers. A similar approach is presented in [14], where the partitioning is based on physical connections.

Our approach, designed for diagnosing faults in large continuous systems, differs from [12]–[17]. Abrupt parametric faults, i.e., a step change in a plant parameter value, produce transients in the system dynamics. Capturing these fault-generated transient behaviors in a discrete-event model by quantizing the measurement or state-space can result in state explosion [18]. We adopt a different approach, where we use the continuous model to derive fault effects as qualitative magnitude and higher-order effects on individual measurements. This produces a compact model for online diagnosis.

We use the global system model to design local diagnosers offline. At runtime, the local diagnosers operate independently to generate local diagnosis results that are globally correct. Our approach does not require a coordinator, and there is minimal or no exchange of information among the diagnosers. This is similar to the third protocol in [12], and a failure in a local diagnoser does not affect the diagnosis capability of the other diagnosers. Therefore, our approach operates like other online distributed diagnosis schemes (e.g., [17]).

III. THE TRANSCEND DIAGNOSIS APPROACH

TRANSCEND [4], [5] is an observer-based fault diagnosis approach that combines quantitative *fault detection* with qualitative *fault isolation* schemes.

A. Modeling for Diagnosis

Our system models represented using bond graphs capture both nominal and faulty dynamic system behavior. Bond graphs are a domain-independent, energy-based, topological modeling scheme for physical processes [3]. The nodes of a bond graph are energy storage elements (capacities, C, and inertias, I); energy dissipation elements (resistors, R); energy transformation elements (gyrators, GY, and transformers, TF); and, input-output elements (sources of effort, Se, and sources of flow, Sf). *Bonds* represent the energy exchange pathways between the bond graph elements. Two variables, *effort*, e_i, and *flow*, f_i, are associated with each bond i, and the product $e_i \times f_i$ defines the rate of energy transfer through the bond i. Two idealized elements, 0- (or parallel) and 1- (or series) junctions, connect bond graph elements and satisfy the principles of conservation of energy and continuity of power. Nonlinear systems are modeled by specifying model parameters as arbitrary functions of other system variables and external signals.

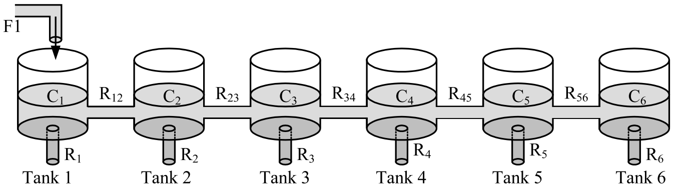


Fig. 1. The six-tank system.

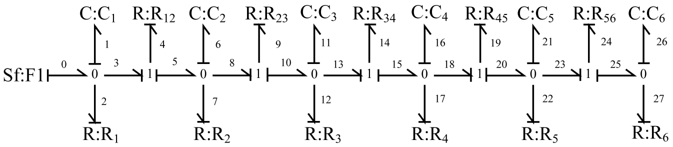


Fig. 2. Bond graph model of the six-tank system.

Fig. 2 shows the bond graph model of an example six-tank fluid system connected by pipes, with a source of flow into the first tank, and drain pipes at the bottom of each tank (Fig. 1). Tanks are modeled as capacitors, and pipes are modeled as resistors [3]. Pipe R_i drains tank C_i and pipe R_{ij} connects tanks C_i and C_j. In the hydraulic domain, the effort, e, denotes pressure, and the flow, f, denotes the fluid flow rate. We use this system to illustrate the different concepts presented in this paper.

1) Diagnosis Model: The *Temporal Causal Graph* (TCG) systematically derived from the bond graph model captures the causal and temporal relations between system variables and forms the basis for an efficient qualitative fault isolation scheme [4], [5]. A TCG is a signal flow graph with the effort and flow variables represented as nodes, and the direction and type of interaction between the variables represented as edges. Fig. 3 shows the TCG for the six-tank system. Edge labels are derived from component constituent relations or relations imposed by junction constraints [4]. For example, for a capacitor in integral causality, the flow-to-effort relation is $f \xrightarrow{\frac{1}{C}dt} e$. The dt specifier implies a temporal edge, i.e., a change in the flow, f, affects the derivative of the effort, e. Resistive elements impose an algebraic relation between effort and flow; and junctions impose direct $(+1)$, inverse (-1), and equality $(=)$ relations between variables.

The TCGs used for diagnosis are extended signal flow graphs (see [4]) which are commonly used by engineers for analyzing system behavior. Therefore, our diagnosis algorithms apply to any annotated signal flow model of dynamic system behavior, independent of how it is derived.

B. Fault Signatures

A *fault* is a persistent change in a component parameter value that causes deviations in a system's nominal behavior. *Abrupt faults* are characterized by parameter value changes that occur at rates much faster than the nominal dynamics of the system. In the six-tank system, the set of possible abrupt faults is $F = \{C_1^-, \ldots, C_6^-, R_1^+, \ldots, R_6^+, R_{12}^+, \ldots, R_{56}^+\}$. A $+$ $(-)$ superscript implies that the fault is an abrupt increase (decrease) in the corresponding parameter value. For example, a block in the tank 1 drain pipe is represented as R_1^+.

The transients produced by abrupt faults can only have discontinuities at the time point of failure. For all other times, the system behavior is continuous and continuously differentiable, and the transient response to a fault can be approximated by its Taylor series expansion:

$$\begin{aligned} y(t) &= y(t_f) + y'(t_f)\frac{(t-t_f)}{1!} \\ &+ y''(t_f)\frac{(t-t_f)^2}{2!} + \ldots + y^{(k)}(t_f)\frac{(t-t_f)^k}{k!} + \ldots, \end{aligned}$$

where t_f is the time point of fault occurrence, and $t > t_f$.

If $|y^{(k+1)}|$ is bounded and t is close to t_f, the Taylor series is a good approximation of the true signal $y(t)$. The time-varying residual signal, $r(t) = y(t) - \hat{y}(t_f)$, where $\hat{y}(t_f)$ is the predicted measurement value at time point t_f, is computed as

$$\begin{aligned} r(t) &= y(t_f) - \hat{y}(t_f) + y'(t_f)\frac{(t-t_f)}{1!} \\ &\quad + y''(t_f)\frac{(t-t_f)^2}{2!} + \ldots + y^{(k)}(t_f)\frac{(t-t_f)^k}{k!} + \ldots, \end{aligned}$$

i.e., the difference, $y(t_f) - \hat{y}(t_f)$, and k derivative values $(y'(t_f), y''(t_f), \ldots, y^{(k)}(t_f))$. After a fault occurs, the nominal system model cannot be used to calculate the numeric values of the derivatives. Instead, we use the TCG model to express the fault residual as qualitative magnitude and derivative changes [4], [5]. This becomes the basis for establishing a signature for a fault transient [5].

Definition 1 (Qualitative Fault Signature): Given a fault f, and measurement m, a *qualitative fault signature*, $FS(f,m)$, of order k, is an ordered $(k+1)$-tuple consisting of the predicted magnitude and 1^{st} through k^{th} order time-derivative effects of a residual signal of measurement m, at the point of failure of fault f, expressed as qualitative values: below normal $(-)$, normal (0), and above normal $(+)$. Typically k is chosen to be the order of the system.

In the remainder of this paper, we abbreviate *qualitative fault signature* and just call it a *fault signature*. Table I shows some fault signatures of the six-tank system. The signature, $(+-+-+-+)$ of fault C_1^- for measurement e_1, the pressure at the bottom of tank 1, implies that an abrupt decrease in capacity of the tank 1 will cause a discontinuous increase in the pressure in the tank at the time point of failure, and then a gradual decrease in the pressure.

After fault detection, online fault isolation compares the magnitude and slope of measurement residual signals to derived fault signatures. Computing higher order derivatives from noisy measurement signals is unreliable [19]. For this measurement scheme, we have shown that all of the discriminatory evidence for fault isolation is provided by the first change in residual magnitude from the point of failure detection [5]. This reduces the possible fault signatures for a measurement to the set of symbols, $\Sigma = \{(+,-),\ (-,+),\ (0,+),\ (0,-)\}$. The first two signatures correspond to a discontinuous change in a signal while the last two signatures imply that at the point of failure, no discontinuous jump in the measurement residual will be observed. $(+,+)$ and $(-,-)$ are not considered because they imply positive feedback loops, and hence, unstable systems.

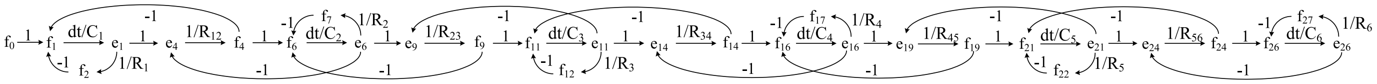


Fig. 3. Temporal causal graph for the six-tank system.

TABLE I

Fault signatures from tanks 1 and 2 for the six-tank system.

Fault	e_1	e_6
C_1^-	$+-+-+-+$	$0+-+-+-$
R_2^+	$0+-+-+-$	$0+-+-+-$
C_2^-	$0+-+-+-$	$+-+-+-+$

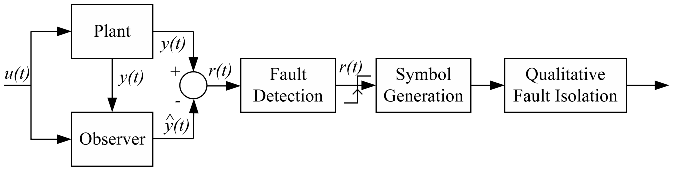


Fig. 4. Block diagram of the Transcend fault diagnosis approach.

Given the set of possible faults, $F = \{f_1, \ldots, f_l\}$, and the set of measurements, $M = \{m_1, \ldots, m_n\}$, the *fault signature matrix*, $FSM_{(F,\ M)} = [FS(f_i, m_j)]_{l \times n}$, is a $l \times n$ matrix with rows corresponding to faults and columns corresponding to measurements, and $FS(f_i, m_j)$, the fault signature of fault f_i for measurement m_j, as its elements. A *fault signature tuple*, $\langle FS(f_i, \widehat{M}) \rangle$, defined for fault f_i and a measurement set $\widehat{M} = \{m_1, m_2, \ldots, m_k\} \subseteq M$, can be extracted from row i of the $FSM_{(F,\ M)}$ by selecting only those elements that are in the columns corresponding to the measurements in $\widehat{M}$. Formally, $\langle FS(f_i, \widehat{M}) \rangle = \langle FS(f_i, m_1), FS(f_i, m_2), \ldots, FS(f_i, m_k) \rangle$.

C. The Diagnosis Approach

We define a system $S = (F, M, FSM_{(F,\ M)})$ to be *diagnosable* if every fault $f_i \in F$ can be uniquely isolated using the measurements in M. Formally, *diagnosability* is defined as follows.

Definition 2 (Diagnosability): Given the set of available measurements, M, and the set of faults, F, a system is *diagnosable* if all single faults in F can be uniquely isolated using M, i.e., $\forall (f_h, f_g \in F, f_h \neq f_g)$, $\langle FS(f_h, M) \rangle \neq \langle FS(f_g, M) \rangle$.

For the six-tank system, with $F = \{C_1^-, C_2^-, R_2^+\}$ and $M = \{e_1, e_6\}$, and the FSM given in Table I, we see that that measurement e_1 can discriminate between faults C_1^- and R_2^+, but not C_2^- and R_2^+. e_6 discriminates between $C_2^-$and R_2^+, so e_1 and e_6 together can uniquely isolate all single faults in F, i.e., the system with faults F is diagnosable using the measurements in M. If $M = \{e_6\}$, then faults C_1^- and R_2^+ cannot be uniquely isolated. The discriminatory power of signatures are the basis for measurement selection algorithms that construct the minimum measurement set to establish complete diagnosability [20].

Given M, we define the qualitative measurement residual as $\Sigma^{|M|}$, the $|M|$-dimensional cartesian product of elements in $\Sigma = \{(+,-),\ (-,+),\ (0,+),\ (0,-)\}$, the set of possible symbols representing the magnitude and lowest-order non-zero derivative of individual measurement residuals. Formally, a diagnoser, $D_{(F,\ M)}$, is defined as $D_{(F,\ M)} = (F, M, H)$, where $H : \Sigma^{|M|} \rightarrow 2^F$ is a mapping from the qualitative measurement residuals in M to the fault hypotheses set. In Transcend, the mapping H is implemented as follows. Using the symbolic qualitative measurement residual deviations, the diagnoser uses a backward propagation algorithm [4] on the TCG to identify the fault hypotheses that match the observed deviation. The diagnoser then monitors the observed qualitative measurement residuals, and compares them to the predicted fault signatures for the fault hypotheses. A mismatch between a residual and a fault signature results in the corresponding hypothesis being dropped, thereby refining the hypothesis set. This process continues till a single fault hypothesis is established, or all residual deviations have been observed.

Fig. 4 illustrates the run time components of the Transcend fault diagnosis approach. The *observer*, implemented as an Extended Kalman filter [21], takes as input the control signals and sensor measurements, and estimates system states as well as outputs. *Fault detection* employs the Z-test, to ensure the observed deviation is statistically significant [22]. A significant deviation triggers the *symbol generation* and *qualitative fault isolation* processes. The symbol generation module takes as input the observed residuals and converts their magnitude and slope into qualitative $+$, $-$, and 0 values, which are used by the Transcend diagnoser for isolating faults.

IV. Formulating the Design Problem for Distributed Diagnosis

Given a system that is *diagnosable*, our objective is to decompose the overall diagnosis task into smaller subtasks performed by local diagnosers with the following properties: *(i)* all single faults of interest in the system can be diagnosed, *(ii)* the local diagnosis results are globally correct, and *(iii)* the number of measurements communicated between the diagnosers to satisfy the above two conditions is minimal. Properties *(i)* and *(ii)* eliminate the need for a centralized coordinator.

For large systems, the system bond graph can be derived from their subsystem bond graph models, with the energy interactions between subsystems captured by connecting bonds and junctions. For diagnosis purposes, a subsystem is defined as $S_i = (F_i, M_i, FSM_{(F_i,\ M_i)})$, where F_i is the set of faults, M_i is the set of measurements, and $FSM_{(F_i,\ M_i)}$ is the fault signature matrix corresponding to F_i and M_i. The F_i's and M_i's together form partitions of the set of faults, F, and all measurements, M, respectively. A subsystem, S_i, is *globally diagnosable* if every single fault, $f \in F_i$, can be *uniquely isolated with respect to the global fault set F* using the measurements, $\widetilde{M_i} \subseteq M$. We use "global diagnosability" in the context of fault isolability. We can have faults in a subsystem that are "locally" diagnosable from other faults in the subsystem, but which may not be "globally" diagnosable from faults outside the subsystem. Formally, *global diagnosability*, which extends the notion of *diagnosability* in Definition 2, is defined as follows:

Definition 3 (Global Diagnosability): Given the set of all faults, F, $F_i \subseteq F$, is *globally diagnosable* by $\widetilde{M_i} \subseteq M$ if $\widetilde{M_i}$ can uniquely isolate every single fault, $f \in F_i$, from all other faults in F, i.e., $\forall (f_h \in F_i, f_g \in F, f_h \neq f_g)$, $\langle FS(f_h, \widetilde{M_i}) \rangle \neq \langle FS(f_g, \widetilde{M_i}) \rangle$.

For the six-tank system in Fig. 1, assume $F = \{C_1^-, C_2^-, R_2^+\}$ and $M = \{e_1, e_6\}$. For a subsystem S_1 with $F_1 = \{C_1^-, R_2^+\}$, and $M_1 = \{e_1\}$, S_1 is not globally diagnosable as the fault signature tuples $\langle FS(R_2^+, M_1) \rangle$ and $\langle FS(C_2^-, M_1) \rangle$ are equal. However, S_1 is globally diagnosable with $\widetilde{M_1} = \{e_1, e_6\}$ since the fault signature tuples $\langle FS(f_i, \widetilde{M_1} \rangle$ for every fault, $f_i \in F_1$, are unique, globally.

Each *local diagnoser*, $D_{(F_i,\, \widetilde{M_i})} = (F_i, \widetilde{M_i}, H_i)$, must satisfy the global diagnosability condition, i.e., all faults in F_i must be globally diagnosable by measurements in $\widetilde{M_i}$. The local diagnosers are each implemented using the TRANSCEND scheme with a distributed, decentralized, extended Kalman filter-based observer (e.g., [21]), a fault detection module, and a symbol generation module. The local diagnosers run independently, and when a measurement deviates, the qualitative fault isolation scheme is triggered for all local diagnosers, which use that measurement for their diagnosis.

We now describe how these local diagnosers generate a global diagnosis result without a coordinator. Assume we have κ local diagnosers $D_{(F_i,\, \widetilde{M_i})}$, $i = 1, 2, \ldots, \kappa$, such that the fault sets, F_i, form a partition of the set of faults F. For the centralized diagnosis scheme, a diagnosis is reached when the fault hypothesis set is reduced to a singleton set. In the distributed diagnosis scheme, since the fault sets F_i form a partition of F, we expect only the local diagnoser responsible for diagnosing the true fault to establish a single fault diagnosis, and the others to return empty diagnoses. In practice, we do not have to wait for all the diagnosers to have reached their final diagnosis results. A global diagnosis result is obtained 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.

Each local diagnoser reports its single or null hypothesis result independently, and the system diagnosis result is determined once conditions 1 or 2 are satisfied. The local diagnosers do not communicate with one another to establish their diagnosis results.

We assume that the system under consideration is *diagnosable*, and develop two different problems for designing distributed diagnosers:

1) In the first problem, we assume the system partition is known and construct local diagnosers for each subsystem that exchange minimal information to globally diagnose each subsystem.
2) In the second problem, we create the system partition structure and local diagnosers simultaneously, in a way that no measurements are shared between the subsystems.

The first problem applies to designing diagnostic schemes for distributed systems with known partition structures. The second problem is more open-ended, and the system partition structure and corresponding diagnosers are derived simultaneously at design time to ensure efficient distributed diagnosis.

In a distributed diagnosis scheme for systems with relatively slow dynamics, such as chemical processes, individual diagnosers implemented for each component can operate independently. The large time constants associated with the global interactions make the subsystem behaviors relatively independent, and the individual diagnosers converge to correct isolation results before the fault effects propagate across subsystem boundaries. Such an approach also works in well-instrumented systems where sensors are placed in close proximity to possible fault sources in individual units, but the cost of employing a large number of sensors may be prohibitive. For system with fast dynamics, such as electromechanical and aerospace systems, fault effects propagate across component boundaries relatively fast, and ignoring component interactions will result in incorrect diagnosis. We need the extra analysis incorporated into our two algorithms to design distributed diagnosers for such systems.

In situations when the system is not globally diagnosable for a set of measurements, we can define the notion of "aggregate faults". An aggregate fault includes all single faults that have the same fault signatures for the available measurements, and hence, are not distinguishable from one other. Our diagnosis methodology can be applied to the reduced fault set with the indistinguishable faults represented as aggregate faults.

Formally, the two problems can be defined as follows:

Problem 1 (Partitioned System Diagnoser Design): Given κ subsystems, $S_i = (F_i, M_i, FSM_{(F_i,\, M_i)})$, $1 \leq i \leq \kappa$, construct, for each subsystem, a measurement set $\widetilde{M_i} \subseteq M$ such that *(i)* $\widetilde{M_i} - M_i$ is minimal, and *(ii)* all single faults in F_i are globally diagnosable by measurements in $\widetilde{M_i}$. Given F_i and $\widetilde{M_i}$, we construct a local diagnoser, $D_{(F_i,\, \widetilde{M_i})}$, for each subsystem. By ensuring that $\widetilde{M_i} - M_i$ is minimal, the local diagnosers share minimal information with one another.

Problem 2 (Unpartitioned System Diagnoser Design): Given a system $S = (F, M, FSM_{(F,\, M)})$, partition F and M into fault and measurement subsets, F_i and $\widetilde{M_i}$, respectively, such that all single faults in F_i are globally diagnosable using measurements only in $\widetilde{M_i}$. From each F_i and $\widetilde{M_i}$ subset pairs, we then construct local diagnosers $D_{(F_i,\, \widetilde{M_i})}$ that do not share any measurements.

These two problems are variations of the *measurement selection* problem [20], with applications in control engineering [23], structural dynamics [24], and robotics [25], among others. The measurement selection problem is an instance of the set covering problem [26], which is known to be NP-complete. Our goal, while designing the local diagnosers, is

Algorithm 1 Designing Diagnosers for a Partitioned System

```
Input: $\kappa$ local subsystems, $S_i = (F_i, M_i, FSM_{(F_i,\ M_i)})$
for each $S_i$ do
  identify $remFaults_i \in F_i$ that cannot be uniquely isolated using $M_i$.
end for
for each $remFaults_i$ do
  $\delta = 1$; $\widetilde{M_i} = M_i$
  while $remFaults_i \neq \varnothing$ do
    identify measurement set $\widehat{M_i}$ from measurements of subsystems $S_i$ at a distance $d \leq \delta$ that isolates maximal $r \in remFaults_i$ and $\widetilde{M_i} - \widehat{M_i}$ is minimal.
    $\widetilde{M_i} = \widetilde{M_i} \cup \widehat{M_i}$
    $remFaults_i = remFaults_i - r$
    if $remFaults_i \neq \varnothing$ then
      $\delta = \delta + 1$
    end if
  end while
  construct $D_{(F_i,\ \widetilde{M_i})}$
end for
```

to select fault-measurement sets that together make the system completely diagnosable, with an emphasis on minimizing the sharing of measurements across sets. For Problem 1, measurement selection is applied to each subsystem with the constraint that the local diagnosis results must be globally correct. Problem 2 represents a "double" measurement selection problem because of the simultaneous partitioning of the fault and measurement sets to ensure that the local diagnosers generate globally correct diagnosis results with no information exchange. To avoid the exponential complexity, we use heuristics that exploit our knowledge of system dynamics to derive less expensive solutions for both problems.

V. Designing the Distributed Diagnosers

We present the two algorithms for generating the distributed diagnosers for continuous systems.

A. Designing Diagnosers for a Partitioned System

Problem 1 designs a local diagnoser for each subsystem $S_i = (F_i, M_i, FSM_{(F_i,\ M_i)})$ using the local measurements, M_i and additional measurements, if required. The goal is to minimize the number of additional measurements, while ensuring that each subsystem is globally diagnosable. For each subsystem S_i, we identify the faults that are not globally diagnosable given M_i, and then, search for a minimal number of additional measurements that will make these faults globally diagnosable.

The search is simplified by defining a notion of proximity among subsystems and using this information to prioritize the selection of additional measurements for a local diagnoser. We represent the system, S, as a graph of connected subsystems. Each subsystem, S_i, forms a node of the graph, and an undirected edge (S_g, S_h) implies direct energy or information exchange between S_g and S_h. The proximity d is defined as the minimum path length from S_g to S_h. The search for additional measurements starts from closer subsystems.

The procedure for designing diagnosers for a partitioned system is presented in Algorithm 1. For each subsystem S_i, we assign to $remFaults_i$ the faults in F_i that cannot be uniquely isolated using measurements in M_i. When $remFaults_i$ is not empty, we start by assigning $\widetilde{M_i}$ equal to M_i, and generating a working measurement set $\widetilde{M_i}^{d \leq 1}$ by pooling in measurements from all subsystems, S_l, at a distance $d \leq 1$ from subsystem S_i, i.e., $\widetilde{M_i}^{d \leq 1} = \bigcup_l M_l$. Using the measurement selection algorithm in [20] we select additional measurements from $\widetilde{M_i}^{d \leq 1} - M_i$ to reduce the number of faults in $remFaults_i$. When different measurement combinations provide the same reductions, we pick the measurement set $\widehat{M_i}$ that adds minimal number of external measurements to M_i while making the maximum number of faults in $remFaults_i$ globally diagnosable. The set $\widetilde{M_i}$ is expanded, and $remFaults_i$ is reduced to a smaller set. If $remFaults_i$ is non-empty, d is incremented, and the procedure is repeated till $remFaults_i$ is empty. At this point, we have the local diagnoser $D_{(F_i,\ \widetilde{M_i})}$. The search algorithm is complete as it will always find the measurements required to diagnose all faults in $remFaults_i$.

We apply this algorithm to the six-tank system example of Fig. 1 with $F = \{C_1^-, \ldots, C_6^-, R_{12}^+, \ldots, R_{56}^+\}$ and $M = \{e_1, f_2, e_6, f_7, e_{11}, f_{12}, e_{16}, f_{17}, e_{21}, f_{22}, e_{26}, f_{27}\}$. The fault signature matrix for the fault and measurement sets appear in Table II. Each tank and the pipe connecting it to the tank on its right defines a subsystem. The six subsystems include the fault sets $F_1 = \{C_1^-, R_{12}^+\}$, $F_2 = \{C_2^-, R_{23}^+\}$, $F_3 = \{C_3^-, R_{34}^+\}$, $F_4 = \{C_4^-, R_{45}^+\}$, $F_5 = \{C_5^-, R_{56}^+\}$, and $F_6 = \{C_6^-\}$, and the measurement sets $M_1 = \{e_1, f_2\}$, $M_2 = \{e_6, f_7\}$, $M_3 = \{e_{11}, f_{12}\}$, $M_4 = \{e_{16}, f_{17}\}$, $M_5 = \{e_{21}, f_{22}\}$, and $M_6 = \{e_{26}, f_{27}\}$.

Algorithm 1 generates the following local diagnosers: $(\{C_1^-, R_{12}^+\}, \{e_1, f_2, \mathbf{e_6}\}, H_1)$, $(\{C_2^-, R_{23}^+\}, \{e_6, f_7, \mathbf{e_{11}}\}, H_2)$, $(\{C_3^-, R_{34}^+\}, \{e_{11}, f_{12}, \mathbf{e_{16}}\}, H_3)$, $(\{C_4^-, R_{45}^+\}, \{e_{16}, f_{17}, \mathbf{e_{21}}\}, H_4)$, $(\{C_5^-, R_{56}^+\}, \{e_{21}, f_{22}, \mathbf{e_{26}}\}, H_6)$, $(\{C_6^-\}, \{e_{26}, f_{27}\}, H_7))$. The external measurements required for global diagnosability appear in bold. A capacitance fault for the i^{th} tank is diagnosable by the effort variable of that tank, but to achieve global diagnosis of the interconnecting pipe faults, the algorithm adds the pressure variable e_{i+1} of the adjoining tank to the measurement set of tank i.

The distributed diagnosis scheme improves the centralized diagnosis approach. Given the system $S = (F, M, FSM_{(F,\ M)})$, we define the size of a centralized diagnoser, $D_{(F,\ M)}$, as the size of its FSM, i.e., $|D_{(F,\ M)}| = |F| \times |M|$. On the other hand, with κ local diagnosers, $D_{(F_i,\ \widetilde{M_i})}$, the total FSM size is $\sum_i |D_{(F_i,\ \widetilde{M_i})}| = \sum_i (|F_i| \times |\widetilde{M_i}|)$. Hence, the total space requirement for all local diagnosers generated using Algorithm 1 will never exceed that of a centralized diagnoser, i.e., $\sum_i |D_{(F_i,\ \widetilde{M_i})}| \leq |D_{(F,\ M)}|$. Only a few measurements are communicated between local diagnosers, so there is considerable savings with the distributed diagnosers.

The computational complexity for deriving the diagnosers for subsystem S_i depends on the number of faults $|F_i|$. The algorithm to find $remFaults_i$ is $O(|F_i|^2)$. To diagnose every element of $remFaults_i$ (which in the worst case, can be of size $O(|F_i|)$, we assume m is the maximum number of measurements in subsystems at a distance of $d = 1$. In the worst case, the algorithm will have to generate all possible combinations of these measurements, i.e., $O(m^{\lfloor \frac{m}{2} \rfloor})$, and the algorithm to

TABLE II
FAULT SIGNATURES FOR THE SIX-TANK SYSTEM EXAMPLE.

Fault	e_1	f_2	e_6	f_7	e_{11}	f_{12}	e_{16}	f_{17}	e_{21}	f_{22}	e_{26}	f_{27}
C_1^-	$+-$	$+-$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$
C_2^-	$0+$	$0+$	$+-$	$+-$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$
C_3^-	$0+$	$0+$	$0+$	$0+$	$+-$	$+-$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$
C_4^-	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$+-$	$+-$	$0+$	$0+$	$0+$	$0+$
C_5^-	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$+-$	$+-$	$0+$	$0+$
C_6^-	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$+-$	$+-$
R_{12}^+	$0+$	$0+$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$
R_{23}^+	$0+$	$0+$	$0+$	$0+$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$
R_{34}^+	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0-$	$0-$	$0-$	$0-$	$0-$	$0-$
R_{45}^+	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0-$	$0-$	$0-$	$0-$
R_{56}^+	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0+$	$0-$	$0-$

Algorithm 2 Designing Diagnosers for an Unpartitioned System

```
Input: Global system $S = (F, M, FSM_{(F,\ M)})$
generate root node $R = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), ..., (\widetilde{M_{|M|}}, F_{|M|})]$ s.t. $|\widetilde{M_i}| = 1$
level $l = 1$
while true do
    check for goal node, $G = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), ..., (\widetilde{M_n}, F_n)]$, at level $l$, s.t. $\cup_i F_i = F$
    if goal node $G$ is found then
        for each $F_i \in G$ such that $F_i \neq \varnothing$ do
            construct $D_{(F_i,\ \widetilde{M_i})}$
        end for
        return
    else
        identify node $N$ s.t. $FC_N = h$
        expand node $N$ to generate level $l + 1$ of search tree
    end if
end while
```

identify the measurement combination that isolates maximal faults in *remFaults*$_i$ while adding the least number of external measurements has complexity $O(m^{\lfloor m \rfloor}|F_i|^2 + |F_i|^2)$. Usually $|remFaults_i| << |F_i|$, and using the measurement selection method in [20] reduces the complexity of this operation to a much smaller value. In the worst case, for all of the κ subsystems, the complexity of the algorithm is $O(\kappa|M|^{\lfloor|M|\rfloor}|F_i|^2 + \kappa|F_i|^2)$, but the average run-time performance of this algorithm is much better. In continuous systems we seldom need to look beyond the immediate neighbors of each subsystem for measurements that diagnose all faults in that subsystem. The tractability of the approach is illustrated in our case study on the ALS system.

B. Designing Diagnosers for an Unpartitioned System

Problem 2 assumes no prior knowledge of subsystem structure for the system S. The goal is to partition the system into subsystems, and construct local diagnosers for each subsystem that satisfy global diagnosability, and do not have to share measurements to achieve global diagnosability. Algorithm 2 solves this problem by generating the maximum number of local diagnosers that do not share measurements, with an added constraint that the measurement subsets are balanced across the diagnosers.

Let $P_I(M)$ denote a partition for the set of measurements, M, in a system, and assume F_i is the set of faults that are globally diagnosable using every $\widetilde{M_i} \in P_I(M)$. Note that F_i can be empty. If $\bigcup_i F_i = F$, for every non-empty F_i, we can construct a set of local diagnosers, $D_{(F_i,\ \widetilde{M_i})} = (F_i, \widetilde{M_i}, H_i)$, that make the system globally diagnosable. The solution to Problem 2 is developed as a tree search algorithm. Each node N of the tree is defined as $N = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), \ldots, (\widetilde{M_n}, F_n)]$ such that $\widetilde{M_i} \in P_N(M)$ and F_i is globally diagnosable with $\widetilde{M_i}$. Our goal is to construct the largest number of local diagnosers which together can globally diagnose all faults in F. Hence our goal node is a node N that partitions M into the largest number of subsets, i.e., $|P_N(M)|$ is maximal, and $\bigcup_i F_i = F$.

The *root node*, R, of the tree is $R = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), ..., (\widetilde{M_{|M|}}, F_{|M|})]$, where each $\widetilde{M_i}$ is represented by a single measurement, i.e., $|\widetilde{M_i}| = 1$. For each $\widetilde{M_i}$, we derive the corresponding F_i such that $\widetilde{M_i}$ produces a global diagnosis for F_i. For a *goal node*, $G = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), ..., (\widetilde{M_n}, F_n)]$, the fault sets F_i cover the set of all faults F, i.e., $\bigcup_i F_i = F$.

The search algorithm generates nodes at level $l + 1$ of the tree by creating all possible pairs of measurement sets from the parent nodes at level l, and computing the corresponding fault-sets for the new measurement sets. For example, if for node $N_1 = [(\widetilde{M_1}, F_1), (\widetilde{M_2}, F_2), (\widetilde{M_3}, F_3)]$, the following nodes will be formed as children of this node: $N_2 = [(\widetilde{M_1} \cup \widetilde{M_2}, F_{12}), (\widetilde{M_3}, F_3)]$, $N_4 = [(\widetilde{M_1} \cup \widetilde{M_3}, F_{13}), (\widetilde{M_2}, F_2)]$, and $N_5 = [(\widetilde{M_2} \cup \widetilde{M_3}, F_{23}), (\widetilde{M_1}, F_1)]$. Note that F_{ij}, the set of faults that are globally diagnosed by measurements in $\widetilde{M_i} \cup \widetilde{M_j}$, can include more faults than $F_i \cup F_j$. This is because the two sets of measurements may uniquely diagnose more faults than the sum of the faults that each can diagnose.

For every new level added to the tree, the algorithm checks if any of the new nodes is a goal node. If there are none, the merge process is repeated at the next level of search till a *goal node* is found. Exhaustive expansion of all nodes at each level would result in an algorithm whose search space and search time are doubly exponential. To reduce computational complexity, our algorithm imposes a greedy heuristic to choose a single node for expansion. If $[N]_l$ represents the set of all nodes at a level l in the search tree, we define our heuristic function $h_l = \max_{\forall N \in [N]_l}(FC_N)$, where $FC_N = |\bigcup_i F_i|$ denotes the total number of faults that are diagnosable in node N by the measurements in $P_N(M)$. Intuitively, at any level, the greedy approach prefers nodes whose local diagnosers can together diagnose the maximum number of faults, i.e., the node with the largest FC_N value is chosen for expansion. The process is repeated till a *goal node* is found.

For a goal node, $G = [(\widetilde{M}_1,F_1),(\widetilde{M}_2,F_2),\ldots,(\widetilde{M}_n,F_n)]$, we construct local diagnosers, $D_{(F_i,\ \widetilde{M}_i)}$, for every fault measurement subset pair, if F_i is not empty. If a fault is uniquely diagnosable by more than one $\widetilde{M}_i$, we assign the fault to the local diagnoser that uses the smallest $\widetilde{M}_i$. This results in balanced diagnosers. It should be noted that for tightly coupled systems, it is possible that the the only solution found by Algorithm 2 is $G = [(M,F)]$, i.e., the system cannot be partitioned.

Algorithm 2 applied to the six-tank system produces seven local diagnosers: $(\{C_1^-\}, \{e_1\}, H_1)$, $(\{C_2^-, R_{12}^+\}, \{e_6\}, H_2)$, $(\{C_3^-, R_{23}^+\}, \{e_{11}, f_7\}, H_3)$, $(\{C_4^-, R_{34}^+\}, \{e_{16}, f_{12}\}, H_4)$, $(\{C_5^-, R_{45}^+\}, \{e_{21}, f_{17}\}, H_5)$, $(\{C_6^-\}, \{f_{27}\}, H_6)$, $(\{R_{56}^+\}, \{e_{26}, f_{22}\}, H_7)$. When one compares the number of node expansions required to generate the solutions, an exhaustive search used 183,074 node expansions, and Algorithm 2 derived its solution with 203 node expansions. We have run a number of other experiments with n-tank systems ($6 \le n \le 15$), and in almost all cases, the heuristic algorithm expanded 1% of the nodes that would be generated by the exhaustive algorithm. This demonstrates that the heuristic algorithm is efficient and generates acceptable solutions.

Like Algorithm 1, $\sum_i |D_{(F_i,\ \widetilde{M}_i)}| = \sum_i(|F_i| \times |\widetilde{M}_i|)$, the size of the local diagnosers is smaller than $|D_{(F,\ M)}|$. Hence, there is considerable space complexity improvement using distributed diagnosers designed by Algorithm 2.

To analyze the time complexity of Algorithm 2, assume $|F| = l$ and $|M| = n$. The root node has n local diagnosers. For each measurement set $\widetilde{M}_i$, we identify the set of faults F_i diagnosable by the measurements in $\widetilde{M}_i$. The faults in F_i have unique fault signatures for the measurements in $\widetilde{M}_i$ and they are computed by traversing the columns of the fault signature matrix, $FSM_{(F,\ M)}$, that correspond to the measurements in $\widetilde{M}_i$. This operation can be computed in $O(l^2n)$ time. To expand the node N, we merge all pairs of $\widetilde{M}_i \in P_N(M)$ to obtain the measurement sets of the children nodes. Therefore, we have $\binom{n}{2}$ nodes in the next level and each node will have $(n-1)$ measurement sets, $\widetilde{M}_i$. Identifying the fault sets, F_i, for each node at this level is also $O(l^2n)$. Since, we are expanding only one node, we will have only $\binom{n-1}{2}$ children. The number of nodes generated is $\binom{n}{2}+\binom{n-1}{2}+\binom{n-3}{2}+\ldots+\binom{2}{2} = O(n^3)$ as there are at most n levels. Hence the complexity of Algorithm 2 is $O(l^2n^4)$, which is polynomial in the number of faults and measurements.

VI. An Experimental Case Study: The Advanced Water Recovery System

We apply our distributed diagnosis approach to a large real-world system, the Advanced Water Recovery System (AWRS), designed and built at the NASA Johnson Space Center (JSC) as part of Advanced Life Support (ALS) Systems for long duration manned missions [6]. The AWRS, shown in Fig. 5, is a closed loop system that converts wastewater to potable water in microgravity conditions.

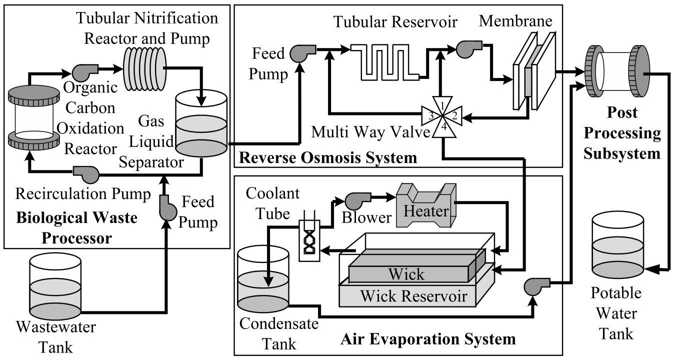


Fig. 5. Schematic of the Advanced Water Recovery System.

A. The Advanced Water Recovery System

The conversion of wastewater, stored in the *Wastewater Tank*, is a multi-step process that starts with a *Biological Waste Processor* (BWP), which removes organic matter and ammonia from the wastewater, followed by a *Reverse Osmosis Subsystem* (RO), which removes inorganic and particulate matter using a *high pressure membrane filtration* system. The concentrated brine that collects in the RO is passed into the *Air Evaporation Subsystem* (AES), which recovers the remaining water using a cyclic *evaporation* and *condensation* process. Finally, the *Post Processing Subsystem* (PPS) uses ultraviolet light treatment to remove trace impurities and infectants from the RO and AES effluents, and the potable water produced is collected in the *Potable Water Tank*.

1) Biological Water Processor: The bond graph model of the BWP is shown in Fig. 6. A *feed pump*, modeled as a constant flow pump using the single flow source, Sf_{fp}, feeds wastewater into the BWP. The *Organic Carbon Oxidation Reactor* (OCOR), which oxidizes the organic carbon, is modeled as a tank, C_{ocor}. The effluent from OCOR enters the *Nitrification Reactor* (NR) through the R_{ocor} pipe. The NR has four parallel tubes (NR_i, $1 \le i \le 4$) with nitrifying microorganisms packed into each tube, and a boost pump that maintains the flow. The resistance R_{NR_i} of NR_i is modeled to increase as wastewater flows through the pipe, simulating the

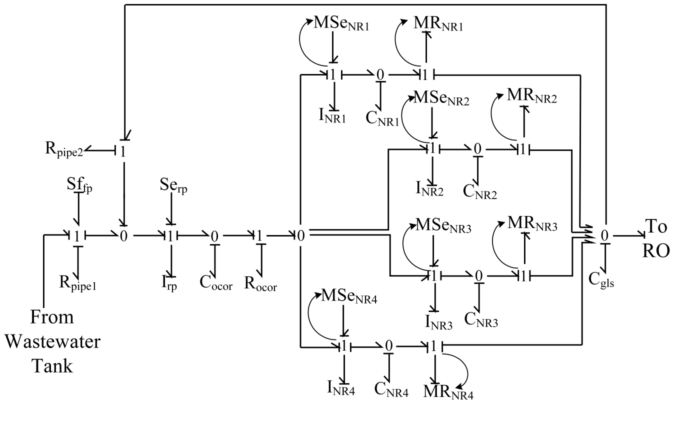


Fig. 6. Bond graph model of the Biological Water Processor.

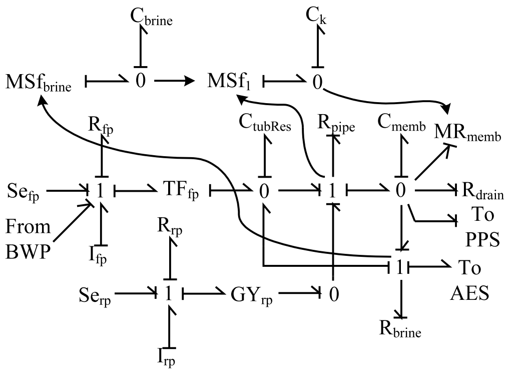


Fig. 7. Bond graph model of the Reverse Osmosis Subsystem.

growth of microorganims as they feed on the organic waste[1]. The effluent of the NR is sent to an ambient pressure *gas-liquid separator* (GLS), modeled as C_{gls}, where the majority of the water effluent is recycled back to the OCOR by the *recirculation pump*, and a smaller stream, equal to the initial feed during steady state operations, is transferred to the RO subsystem for further processing. The recirculation pump is modeled as a simple boost pump with two bond graph elements: an effort source, Se_{rp}, and the pump rotor intertia, I_{rp}. R_{pipe1} and R_{pipe2} model the pipes between the feed pump and the OCOR, and the GLS and the recirculation pump, respectively.

2) Reverse Osmosis Subsystem: Fig. 7 shows the bond graph model for the RO subsystem. The *feed pump* that moves effluent from the BWP into the RO is modeled as a source of effort, Se_{fp}, with rotor inertia, I_{fp}, and resistance, R_{fp}, to model frictional losses. The transformer, TF_{fp}, models the conversion of rotational speed to fluid flow. A coiled pipe, modeled as C_{tubRes}, acts as a tubular reservoir to help reduce fluctuations in liquid flow through the system. The connecting pipe is modeled as a resistance R_{pipe}. The RO subsystem operates in multiple modes, determined by the 4-way multi-position valve, but in this work, we restrict the RO to the primary mode of operation where the water circulates in a longer loop. The *recirculation pump* has parameters Se_{rp}, R_{rp}, I_{rp}, and GY_{rp}. The *membrane* assembly is modeled as a fixed chamber with capacitor, C_{memb}, and a variable resistance, R_{memb}, that models the resistance to flow through the membrane. Dirt accumulates as waste water flows through the membrane causing R_{memb} to increase, and the outflow of clean water to decrease with time. Hence, the resistance, R_{memb}, is modulated by the *conductivity* (K) of the water flowing in the system. The water that does not pass through the membrane has a greater concentration of impurities, and is recirculated through the pipe, R_{brine}.

3) Air Evaporation Subsystem: The bond graph, shown in Fig. 8, models the AES. It includes the *wick*, a porous element modeled as C_{wick}, which dips into the brine that is collected in a tank. Warm air blown over the wick evaporates some of the water. C_{steam} represents the quantity of vapor generated due to the evaporation. The moisture laden air is then passed through a chilled water heat exchanger, and clean condensate is collected in the condensate tank, $C_{condensate}$. The *condensate pump*, modeled as a simple source of flow, $Sf_{condFlow}$, pumps water to the PPS in a continuous stream. A blower (modeled as Se_{blower}) is used in the airflow loop to maintain the flowrate, and a heater (Se_{heater}) heats up the air cooled in the exchanger to ensure that its capacity to absorb moisture remains high. The transformers, TF_{blower} and TF_{heater}, model the efficiency of the blower and the heater, respectively. The energy exchanges and temperature content at different parts of the air in the AES is modeled as capacitors C_{air_i} $(1 \leq i \leq 3)$. $R_{airFlow}$ models the resistance to the flow of air in the AES heat exchange loop.

4) Post Processing Subsystem: The PPS disinfects the effluent from the RO and the AES components through a five step treatment procedure to generate potable water. Since the PPS does not have interesting flow dynamics, we do not include it in our diagnosis model.

The multi-domain bond graph models represent the mechanical and hydraulic domains in the BWP, RO, and AES. The RO bond graph also models the fluid conductivity domain, to simulate the changing concentration of impurities and their effects on the flow process. The AES bond graph models the exchange of heat between the water absorbed by the wick, the air, and the coolant liquid in the thermal domain.

The AWRS is a large, complex, physical system with interacting subsystems, each containing a large number of components. These interactions cause fault effects to propagate across subsystem boundaries, eventually affecting all parts of the system. Hence, a centralized approach, when applied to this system, will have high memory and computation requirements. On the other hand, the well-defined subsystem structure of the AWRS lends itself well to our distributed diagnosis approach.

B. Diagnoser Design Experiments

The AWRS testbed is well instrumented. Table III shows the list of measurements and faults that we chose for these

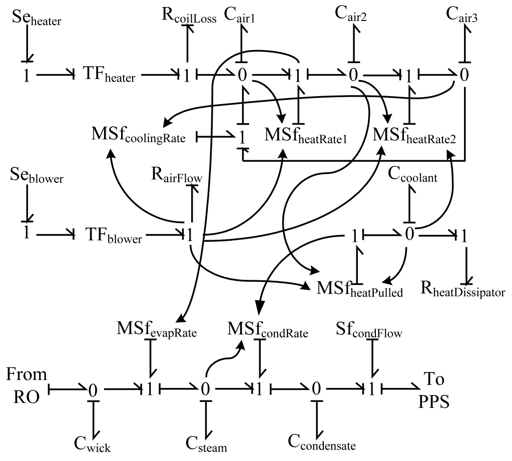


Fig. 8. Bond graph model of the Air Evaporation System.

[1] Note that in the bond graphs, components modulated as a function of system variables have a prefix M added to their names, e.g., MR_{NR_i} denotes that R_{NR_i} is modulated by the flow of water through it. An arrow pointing from the system variable to a modulated component represents this modulation graphically.

TABLE III
MEASUREMENTS AND FAULTS CHOSEN FOR THE EXPERIMENTS.

Subsystem	Measurement	Description	Fault	Description
BWP	$BWP.P_{rp}$	Recirculation pump output pressure	$BWP.C^{-}_{gls}$	Buildup of sediments in the GLS tank
	$BWP.P_{gls}$	Pressure in the GLS tank	$BWP.R^{+}_{ocor}$	Blockage in the R_{ocor} pipe
	$BWP.P_{ocor}$	Pressure at the output of the OCOR	$BWP.R^{+}_{pipe1}$	Blockage in the R_{pipe1} pipe
	$BWP.P_{fp}$	Pressure at the output of the feed pump	$BWP.R^{+}_{pipe2}$	Blockage in the R_{pipe2} pipe
	$BWP.F_{NR_1}$	Flow through nitrifier tube 1	$BWP.R^{+}_{NR_1}$	Blockage in nitrifier tube 1
	$BWP.F_{NR_2}$	Flow through nitrifier tube 2	$BWP.R^{+}_{NR_2}$	Blockage in nitrifier tube 2
	$BWP.F_{NR_3}$	Flow through nitrifier tube 3	$BWP.R^{+}_{NR_3}$	Blockage in nitrifier tube 3
	$BWP.F_{NR_4}$	Flow through nitrifier tube 4	$BWP.R^{+}_{NR_4}$	Blockage in nitrifier tube 4
RO	$RO.P_{memb}$	Pressure in the membrane	$RO.R^{+}_{brine}$	Blockage in the pipe carrying brine
	$RO.F_{permeate}$	Permeate flow rate	$RO.C^{-}_{memb}$	Buildup of sediment in the membrane
	$RO.P_{rp}$	Pressure at the output of the feed pump	$RO.R^{+}_{memb}$	Blockage of flow through the membrane
	$RO.P_{back}$	Backflow pressure	$RO.R^{+}_{pipe}$	Blockage in pipe carrying water to the membrane
			$RO.TF^{-}_{fp}$	Decrease in efficiency of the feed pump
			$RO.GY^{-}_{rp}$	Decrease in efficiency of the recirculation pump
AES	$AES.V_{air}$	Velocity of air flowing through the wick	$AES.TF^{-}_{blower}$	Decrease in efficiency of the blower
	$AES.P_{wick}$	Pressure in the wick	$AES.TF^{-}_{heater}$	Decrease in efficiency of the heater
	$AES.P_{steam}$	Pressure of the steam generated	$AES.C^{-}_{wick}$	Buildup of sediment in the wick
	$AES.P_{condensate}$	Pressure in the condensate tank	$AES.C^{-}_{steam}$	Decrease in the capacity to produce steam
	$AES.T_{coolant}$	Temperature of the coolant	$AES.C^{-}_{condensate}$	Buildup of sediment in the condensate tank
			$AES.R^{+}_{airFlow}$	Reduction of airflow

TABLE IV
RESULTS FOR EXPERIMENTS 1-A, 1-B, AND 1-C.

Faults	Maximal number (17) of Measurements Considered	Minimal number (14) of Measurements Considered	Intermediate number (16) of Measurements Considered
$BWP.C^{-}_{gls}$, $BWP.R^{+}_{ocor}$	$BWP.P_{rp}$, $BWP.P_{gls}$		$BWP.P_{rp}$
$BWP.R^{+}_{pipe1}$, $BWP.R^{+}_{pipe2}$	$BWP.P_{ocor}$, $BWP.P_{fp}$	$BWP.P_{ocor}$, $BWP.P_{fp}$	$BWP.P_{ocor}$, $BWP.P_{fp}$
$BWP.R^{+}_{NR_1}$, $BWP.R^{+}_{NR_2}$	$BWP.F_{NR_1}$, $BWP.F_{NR_2}$	$BWP.F_{NR_1}$, $BWP.F_{NR_2}$	$BWP.F_{NR_1}$, $BWP.F_{NR_2}$
$BWP.R^{+}_{NR_3}$, $BWP.R^{+}_{NR_4}$	$BWP.F_{NR_3}$, $BWP.F_{NR_4}$	$BWP.F_{NR_3}$, $BWP.F_{NR_4}$	$BWP.F_{NR_3}$, $BWP.F_{NR_4}$
$RO.R^{+}_{brine}$, $RO.TF^{-}_{fp}$	$RO.P_{memb}$	$RO.P_{memb}$, $RO.P_{rp}$	$RO.P_{memb}$, $RO.F_{permeate}$
$RO.C^{-}_{memb}$, $RO.R^{+}_{memb}$	$RO.F_{permeate}$, $\mathbf{BWP}.\mathbf{P}_{\mathbf{rp}}$	$RO.P_{back}$, $\mathbf{BWP}.\mathbf{P}_{\mathbf{ocor}}$	$RO.P_{rp}$, $RO.P_{back}$
$RO.R^{+}_{pipe}$, $RO.GY^{-}_{rp}$	$RO.P_{rp}$, $RO.P_{back}$	$\mathbf{AES}.\mathbf{P}_{\mathbf{wick}}$	$\mathbf{BWP}.\mathbf{P}_{\mathbf{rp}}$
$AES.TF^{-}_{blower}$, $AES.TF^{-}_{heater}$	$AES.V_{air}$	$AES.V_{air}$	$AES.V_{air}$
$AES.C^{-}_{wick}$, $AES.C^{-}_{steam}$	$AES.P_{wick}$, $AES.P_{steam}$	$AES.P_{wick}$, $AES.P_{steam}$	$AES.P_{wick}$, $AES.P_{steam}$
$AES.C^{-}_{condensate}$, $AES.R^{+}_{airFlow}$	$AES.P_{condensate}$, $AES.T_{coolant}$	$AES.P_{condensate}$, $AES.T_{coolant}$	$AES.P_{condensate}$, $AES.T_{coolant}$

experiments. In the following, we first derive diagnosers for the three AWRS subsystems using three measurements sets. Then diagnoser-design experiments are run assuming the subsystem structure is unknown.

We use the bond graph model described above to systematically derive the TCG for the AWRS. The distributed diagnosers are derived from this model using a Python implementation of the design algorithms.

1) Designing Diagnosers for a Partitioned System : We assume the AWRS to be partitioned into the BWP, RO, and AES subsystems. We run three experiments, for the same fault set (see Table IV), but with different measurement sets. The prefixes *BWP*, *RO*, and *AES*, in Table IV, indicate that the measurement or fault is associated with the BWP, RO, and AES subsystem, respectively.

Experiment 1-A is run with measurements shown in Table IV, column 2. The BWP and AES measurements are sufficient to generate global diagnosis results for these subsystems. However, the RO subsystem diagnoser needs the pressure at the BWP recirculation pump, $\mathbf{BWP}.\mathbf{P}_{\mathbf{rp}}$, to uniquely isolate all of its faults.

Experiment 1-B uses a measurement set generated by the measurement selection algorithm [20]. These 14 measurements listed in Table IV, column 3, are the minimum number of measurements required to isolate all faults. The diagnosers for the BWP and the AES are the same as in Experiment 1-A. However, the RO diagnoser now needs two external measurements, $\mathbf{BWP}.\mathbf{P}_{\mathbf{ocor}}$, and $\mathbf{AES}.\mathbf{P}_{\mathbf{wick}}$, to uniquely isolate all of

TABLE V
RESULTS FOR EXPERIMENT 2-A (17 MEASUREMENTS).

Faults	Measurements
$BWP.C^{-}_{gls}$, $BWP.R^{+}_{ocor}$	$BWP.P_{gls}$, $BWP.P_{ocor}$
$BWP.R^{+}_{pipe1}$, $BWP.R^{+}_{pipe2}$	$BWP.P_{fp}$
$BWP.R^{+}_{NR_1}$, $BWP.R^{+}_{NR_2}$	$BWP.F_{NR_1}$, $BWP.F_{NR_2}$
$BWP.R^{+}_{NR_3}$, $BWP.R^{+}_{NR_4}$	$BWP.F_{NR_3}$, $BWP.F_{NR_4}$
$RO.R^{+}_{brine}$, $RO.TF^{-}_{fp}$	$BWP.P_{rp}$, $RO.P_{memb}$
$RO.C^{-}_{memb}$, $RO.R^{+}_{memb}$	$RO.F_{permeate}$
$RO.R^{+}_{pipe}$, $RO.GY^{-}_{rp}$	$RO.P_{rp}$, $RO.P_{back}$
$AES.C^{-}_{wick}$	$AES.P_{wick}$
$AES.C^{-}_{steam}$, $AES.TF^{-}_{blower}$, $AES.TF^{-}_{heater}$	$AES.V_{air}$, $AES.P_{steam}$
$AES.C^{-}_{condensate}$	$AES.P_{condensate}$
$AES.R^{+}_{airFlow}$	$AES.T_{coolant}$

TABLE VI
RESULTS FOR EXPERIMENT 2-B (14 MEASUREMENTS).

Faults	Measurements
$BWP.R^{+}_{NR_1}$, $BWP.R^{+}_{NR_2}$ $BWP.C^{-}_{gls}$, $BWP.R^{+}_{ocor}$ $BWP.R^{+}_{pipe1}$, $BWP.R^{+}_{pipe2}$	$BWP.F_{NR_1}$, $BWP.F_{NR_2}$ $BWP.P_{ocor}$, $BWP.P_{fp}$
$BWP.R^{+}_{NR_3}$, $BWP.R^{+}_{NR_4}$ $RO.C^{-}_{memb}$, $RO.R^{+}_{memb}$ $RO.R^{+}_{pipe}$, $RO.GY^{-}_{rp}$ $RO.R^{+}_{brine}$, $RO.TF^{-}_{fp}$ $AES.C^{-}_{wick}$, $AES.TF^{-}_{blower}$ $AES.TF^{-}_{heater}$	$BWP.F_{NR_3}$, $BWP.F_{NR_4}$ $RO.P_{memb}$,$RO.P_{back}$ $RO.P_{rp}$, $AES.P_{wick}$ $AES.P_{steam}$, $AES.V_{air}$
$AES.C^{-}_{condensate}$, $AES.R^{+}_{airFlow}$ $AES.C^{-}_{steam}$	$AES.P_{condensate}$, $AES.T_{coolant}$

TABLE VII
RESULTS FOR EXPERIMENT 2-C (16 MEASUREMENTS).

Faults	Measurements
$BWP.R^{+}_{NR_1}$, $BWP.R^{+}_{NR_2}$ $BWP.R^{+}_{NR_3}$, $BWP.R^{+}_{NR_4}$ $BWP.C^{-}_{gls}$, $BWP.R^{+}_{ocor}$ $BWP.R^{+}_{pipe1}$, $BWP.R^{+}_{pipe2}$ $RO.R^{+}_{brine}$, $RO.R^{+}_{pipe}$ $RO.GY^{-}_{rp}$, $RO.TF^{-}_{fp}$ $AES.C^{-}_{condensate}$	$BWP.P_{rp}$, $BWP.P_{ocor}$ $BWP.P_{fp}$, $BWP.F_{NR_1}$ $BWP.F_{NR_2}$, $BWP.F_{NR_3}$ $BWP.F_{NR_4}$, $RO.P_{rp}$ $RO.F_{back}$, $AES.P_{condensate}$
$RO.C^{-}_{memb}$	$RO.P_{memb}$
$RO.P^{+}_{memb}$	$RO.F_{permeate}$
$AES.C^{-}_{steam}$, $AES.R^{+}_{airFlow}$ $AES.C^{-}_{wick}$, $AES.TF^{-}_{blower}$ $AES.TF^{-}_{heater}$	$AES.P_{wick}$, $AES.P_{steam}$ $AES.T_{coolant}$, $AES.V_{air}$

TABLE VIII
SOME FAULT SIGNATURES FOR THE AWRS DIAGNOSIS EXPERIMENT.

Fault	$BWP.$ P_{ocor}	$RO.$ P_{memb}	$RO.$ P_{back}	$RO.$ P_{rp}	$AES.$ P_{wick}
$BWP.C^{-}_{gls}$	0−	0+	0−	0+	0+
$RO.C^{-}_{memb}$	0−	+−	+−	0−	0+
$RO.R^{+}_{memb}$	0−	0+	0+	0−	0+
$RO.R^{+}_{brine}$	0+	0+	0+	0−	0+
$RO.R^{+}_{pipe}$	0−	0−	0−	0−	0−
$RO.TF^{-}_{fp}$	0+	0−	0+	0−	0+
$AES.TF^{-}_{blower}$	0+	0−	0−	0−	0−
$AES.TF^{-}_{heater}$	0+	0−	0−	0−	0−

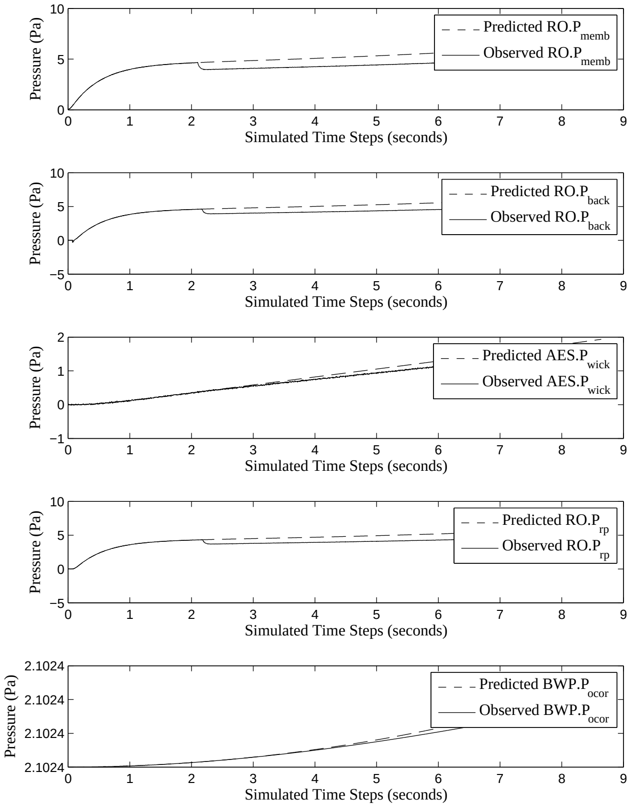


Fig. 9. Experimental observations.

its single faults.

Experiment 1-C uses 16 measurements (column 4 of Table IV). Like Experiment 1-A, only $\mathbf{BWP.P_{rp}}$ needs to be communicated to the RO for complete diagnosability. This shows that the extra measurement in Experiment 1-A provides no additional diagnostic information.

The derived local diagnoser structures match our intuition. Comparing the results of the experiments with 14 measurements to that with 16 measurements, it is clear that additional measurements provide more redundancy of information, and make the diagnosers more independent. The trade-off between the cost of additional sensors versus greater communication overhead and independence of the local diagnosers is evident.

2) Designing Diagnosers for an Unpartitioned System : For the case where we did not assume any subsystem information, we again ran three experiments for the measurement sets and faults listed in Table III.

Experiment 2-A to 2-C produced 11, 3 and 4 local diagnosers, respectively (see Tables V-VII).

It is clear that additional measurements increases redundancy, which Algorithm 2 exploits to create smaller diagnosers. Tables V and VI results show that the balance heuristic works well. The Table VII result is different, because the algorithm derived one large, one medium, and two very small diagnosers. A different set of 16 measurements would very likely have produced a more balanced result.

Comparing the results of the experiments with 14 measurements, the partition structure created by Algorithm 2 is found to be similar to that generated by Algorithm 1, even though Algorithm 2 rearranges the faults and measurements between the diagnosers to ensure that less measurements are needed for each diagnoser. For the experiments with additional measurements, Algorithm 2 tends to use the redundant information to create a larger number of smaller diagnosers, to improve the overall computational efficiency.

C. Distributed Fault Isolation

We illustrate the online operation with one set of distributed diagnosers. We show how the local diagnosers generated in Experiment 1-B isolate a block in the pipe ($RO.R_{pipe}$) that connects the tubular reservoir to the membrane in the RO subsystem. The three local diagnosers are implemented as described in Section IV.

For this demonstration, we use a Matlab® Simulink® simulation model of the AWRS that was systematically derived from the bond graph models described in Section VI-A [27].

TABLE IX
DIAGNOSIS RESULTS FOR 20% ABRUPT FAULT $RO.R^+_{pipe}$ AT $t_f = 21000$ SECONDS.

Step	Symbols	$D_{(F_1,\ \widetilde{M}_1)}$ (BWP) Candidate set	$D_{(F_2,\ \widetilde{M}_2)}$ (RO) Candidate set	$D_{(F_3,\ \widetilde{M}_3)}$ (AES) Candidate set
0	$RO.P_{memb} : (0-)$	No measurement deviation detected	$RO.R^+_{pipe}$, $RO.TF^-_{fp}$	No measurement deviation detected
1	$RO.P_{back} : (0-)$	No measurement deviation detected	$RO.R^+_{pipe}$	No measurement deviation detected
2	$RO.P_{rp} : (0-)$	No measurement deviation detected	$RO.R^+_{pipe}$	No measurement deviation detected
3	$AES.P_{wick} : (0-)$	No measurement deviation detected	$RO.R^+_{pipe}$	$AES.TF^-_{blower}$, $AES.TF^-_{heater}$
4	$BWP.P_{ocor} : (0-)$	$BWP.C^-_{gls}$	$RO.R^+_{pipe}$	$\varnothing$

The fault, modeled as a 20% abrupt increase in the $RO.R^+_{pipe}$ pipe resistance, is introduced at time $t = 21,000$ seconds. The simulation is run for 86,400 simulation seconds. Measurement noise is Gaussian with a noise power level set at 2% of the average signal power for each measurement. The measurements are sampled at 1 Hz. Table VIII gives some of the relevant fault signatures for this experiment.

The diagnosis steps are shown in Table. IX. A block causes decreased flow through the pipe initially. As a result, $RO.P_{memb}$, the pressure in the membrane, decreases, but not discontinuously $(0-)$. The deviation in $RO.P_{memb}$ is first detected by the RO diagnoser. The candidate set, at this time, includes $RO.P^+_{pipe}$, and a decrease in the RO feed pump efficiency, $RO.TF^+_{fp}$, the only faults whose fault signatures are consistent with the observed $(0-)$ change. Subsequently, measurement $RO.P_{back}$, i.e., the pressure in the RO loop also deviates as $(0-)$. The fault signature of $RO.TF^-_{fp}$ for this measurement is not consistent with this change and hence this fault is dropped from the candidate list. At this point, $RO.R^+_{pipe}$ is the only fault candidate, but all measurements of $D_{(F_2,\ \widetilde{M}_2)}$ have not deviated, therefore, we cannot be sure that we have the final diagnosis result. The measurement deviation, $(0-)$, in $RO.P_{rp}$ is consistent with the candidate. The fourth measurement deviation observed, is a drop in the pressure in the wick reservoir, i.e., $AES.P^+_{wick}$. The observed deviation $(0-)$ continues to be consistent with the $RO.R^+_{pipe}$ fault candidate. Since this measurement is also accessible to AES, it triggers the fault diagnoser $D_{(F_3,\ \widetilde{M}_3)}$ and generates the fault candidate set of size 2. Finally, when $BWP.P_{ocor}$ is observed to deviate, diagnoser $D_{(F_1,\ \widetilde{M}_1)}$ is initiated with a single fault in the candidate set. This deviation is inconsistent with the candidates of $D_{(F_3,\ \widetilde{M}_3)}$, and hence they are dropped. Hence $D_{(F_3,\ \widetilde{M}_3)}$ generates a *null diagnosis*. Since *all* measurements of $D_{(F_2,\ \widetilde{M}_2)}$ have deviated, and it has one fault candidate remaining, the system supervisor declares $RO.R^+_{pipe}$ as the true fault, and this corresponds to the correct global diagnosis. The plots for the measurement deviations are shown in Fig. 9.

VII. SUMMARY AND CONCLUSIONS

In this paper, we have presented a novel model-based distributed diagnosis approach, where local diagnosers generate globally correct local diagnosis results, with minimal exchange of information, and no coordination. Since no coordination is required, the computational complexity of the overall diagnosis task is significantly reduced. Moreover, minimal exchange of information also guarantees reduction in communication overhead. We proposed two approaches to design distributed diagnosers. In the first approach, we assumed knowledge of subsystem structure, especially the measurements and faults that belong to each subsystem, and based on this information, we designed a local diagnoser for each subsystem such that it required minimal number of additional *external* measurements to diagnose *all* the faults assigned to that subsystem. In the second approach, we assumed no prior partitioning information. Instead, we generated the maximal number of distributed diagnosers, such that, each local diagnoser could operate independently without sharing measurements.

In future work, we will adopt recent extensions to the TRANSCEND algorithm to allow for diagnosis of multiple faults [7] and incipient fault diagnosis [8]. Other extensions include fault identification, and use of this information in distributed fault adaptive control schemes, as well as analysis with uncertain models. Finally, [28] presents a DES approach for diagnosis of continuous systems, derived from the TRANSCEND diagnostic framework. This approach automatically constructs a labeled transition system that describes the fault model, and also generates a computationally efficient event-based diagnoser. As part of future work, we would like to investigate how the algorithms described in this paper can be extended to develop distributed DES approaches for diagnosing continuous systems.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their constructive comments in improving the technical quality and presentation of this paper.

REFERENCES

[1] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete-event systems," *IEEE Transanctions on Automatic Control*, vol. 40, no. 9, pp. 1555–1575, 1995.

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

[3] D. C. Karnopp, D. L. Margolis, and R. C. Rosenberg, *Systems Dynamics: Modeling and Simulation of Mechatronic Systems*, 3rd ed. New York, NY: John Wiley & Sons, Inc., 2000.

[4] P. J. Mosterman and G. Biswas, "Diagnosis of continuous valued systems in transient operating regions," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 29, no. 6, pp. 554–565, 1999.

[5] E.-J. Manders, S. Narasimhan, G. Biswas, and P. J. Mosterman, "A combined qualitative/quantitative approach for fault isolation in continuous dynamic systems," in *Proceedings of the 4th Symposium on Fault Detection, Supervision, and Safety for Technical Processes*, Budapest, Hungary, 2000, pp. 1074–1079.

[6] K. D. Pickering, K. Wines, G. M. Pariani, L. A. Franks, J. Yeh, B. W. Finger, M. L. Campbell, C. E. Verostko, C. Carrier, J. C. Gandhi, and L. M. Vega, "Early results of an integrated water recovery system test," in *Proceedings of the* 29*th* *International Conference on Environmental Systems*, Orlando, FL, 2001.

[7] M. Daigle, X. Koutsoukos, and G. Biswas, "Multiple fault diagnosis in complex physical systems," in *Proceedings of the* 17*th* *International Workshop on Principles of Diagnosis*, Spain, 2006, pp. 69–76.

[8] I. Roychoudhury, G. Biswas, and X. Koutsoukos, "A Bayesian approach to efficient diagnosis of incipient faults," in *Proceedings of the* 17*th* *International Workshop on Principles of Diagnosis*, Spain, 2006, pp. 243–250.

[9] R. Davis and W. Hamscher, "Model-based reasoning: troubleshooting," *Exploring Artificial Intelligence*, pp. 297–346, 1988.

[10] X. Zhang, M. M. Polycarpou, and T. Parisini, "A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 4, pp. 576–593, Apr. 2002.

[11] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. N. Kavuri, "A review of process fault detection and diagnosis Part I: Quantitative model-based methods," *Computers and Chemical Engineering*, vol. 27, pp. 293–311, 2003.

[12] R. Debouk, S. Lafortune, and D. Teneketzis, "Coordinated decentralized protocols for failure diagnosis of discrete event systems," *Discrete Event Dynamic System: Theory and Applications*, vol. 10, no. 1-2, pp. 33–86, 2000.

[13] J. Kurien, X. Koutsoukos, and F. Zhao, "Distributed diagnosis of networked embedded systems," in *Proceedings of the* 13*th* *International Workshop on Principles of Diagnosis*, Semmering, Austria, 2002, pp. 179–188.

[14] R. Su and W. M. Wonham, "Global and local consistencies in distributed fault diagnosis for discrete-event systems," *IEEE Transactions on Automatic Control*, vol. 50, no. 12, pp. 1923–1935, 2005.

[15] E. Fabre, A. Benveniste, S. Haar, and C. Jard, "Distributed monitoring of concurrent and asynchronous systems," *Journal of Discrete Event Systems*, vol. 15, no. 1, pp. 33–84, 2005.

[16] P. Baroni, G. Lamperti, P. Pogliano, and M. Zanella, "Diagnosis of large active systems," *Artificial Intelligence*, vol. 110, no. 1, pp. 135–183, 1999.

[17] Y. Pencole and M.-O. Cordier, "A formal framework for the decentralised diagnosis of large scale discrete event systems and its application to telecommunication networks," *Artificial Intelligence*, vol. 164, no. 1-2, pp. 121–170, 2005.

[18] J. Lunze, "Diagnosis of quantized systems based on a timed discrete-event model," *IEEE Transactions on Systems, Man and Cybernetics, Part A*, vol. 30, no. 3, pp. 322–336, 2000.

[19] M. J. Chantler, S. Daus, T. Vikatos, and G. M. Coghill, "The use of qualitative dynamic models and dependency recording for diagnosis," in *Proceedings of the* 9*th* *International Workshop on the Principles of Diagnosis*, Cape Cod, MA USA, 1998, pp. 59–68.

[20] S. Narasimhan, P. J. Mosterman, and G. Biswas, "A systematic analysis of measurement selection algorithms for fault isolation in dynamic systems," in *Proceedings of the* 9*th* *International Workshop on Principles of Diagnosis*, Cape Cod, MA USA, 1998, pp. 94–101.

[21] A. G. Mutambara, *Decentralized Estimation and Control for Multisensor Systems*. Boca Raton: CRC Press, 1998.

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

[23] T. K. Yu and J. H. Seinfeld, "Observality and optimal measurement locations in linear distributed parameter systems," *International Journal of Control*, vol. 18, pp. 785–799, 1973.

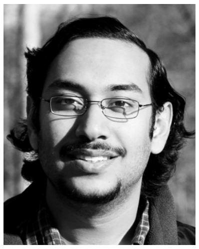
[24] J. E. T. Penny, M. L. Friswell, and S. D. Garvey, "Automatic choice of measurement locations for dynamic testing," *AIAA Journal*, vol. 32, no. 2, pp. 407–414, 1994.

[25] S. Y. Chen and Y. F. Li, "Automatic sensor placement for model-based robot vision," *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 34, no. 1, pp. 393–408, Feb 2004.

[26] V. T. Paschos, "A survey of approximately optimal solutions to some covering and packing problems," *ACM Computing Surveys*, vol. 29, no. 2, pp. 171–209, 1997.

[27] I. Roychoudhury, M. Daigle, G. Biswas, X. Koutsoukos, and P. J. Mosterman, "A method for efficient simulation of hybrid bond graphs," in *Proceedings of the International Conference of Bond Graph Modeling*, San Diego, CA, 2007, pp. 177–184.

[28] M. Daigle, X. Koutsoukos, and G. Biswas, "A discrete event approach to diagnosis of continuous systems," in *Proceedings of the* 18*th* *International Workshop on Principles of Diagnosis*, Nashville, TN, May 2007, pp. 259–266.



Indranil Roychoudhury (S'08) received the B.E. (Hons.) degree in electrical and electronics engineering from Birla Institute of Technology and Science, Pilani, Rajasthan, India, in 2004, and the M.S. degree in computer science from Vanderbilt University, Nashville, TN, in 2006, where he is currently a Ph.D. Candidate in computer science.

Since September 2004, he has been a Graduate Research Assistant with the Institute for Software Integrated Systems, Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN. During the summers of 2006 and 2007, he was an intern with Mission Critical Technologies, Inc., at the NASA Ames Research Center. 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.

He is a Professor of Computer Science and Computer Engineering in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, and a Senior Research Scientist at the Institute for Software Integrated Systems (ISIS), Vanderbilt University. He conducts research in 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 and 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.

Dr. Biswas is an Associate Editor of the IEEE Transactions on Systems, Man, and Cybernetics — Part A: Systems and Humans. He has served on the Program Committees of a number of conferences. He is a senior member of the IEEE Computer Society, ACM, AAAI, and the Sigma Xi Research Society. His research has been supported by funding from NASA, NSF, DARPA, and ONR.

Xenofon 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, 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 the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, where he is currently an Assistant Professor and a Senior Research Scientist in the Institute for Software Integrated Systems. He has authored or coauthored more than 80 technical publications and is the holder of three U.S. patents. His research interests include hybrid systems, real-time embedded systems, and sensor networks. He currently serves as Associate Editor for the ACM Transactions on Sensor Networks and for Modelling Simulation Practice and Theory.

Dr. Koutsoukos is a senior member of IEEE and a member of ACM. He was the recipient of the National Science Foundation CAREER Award in 2004.